Introduction aux réseaux

Noël Giraud
Institut de Physique nucléaire de Lyon

Table des matières

Table des matières 6
Table des illustrations 7
Introduction 9
Chapitre 1 Qu'est-ce qu'un réseau ? 11
  Historique 11
    Le temps des « mainframes » 11
    La tentative de l'OSI 11
    L'apparition des réseaux locaux 12
    La naissance de l'Internet 12
    L'explosion du Web 12
  Qu'est-ce qu'un réseau ? 12
  Les organismes de normalisation 13
    ANSI 13
    IEEE 13
    IAB 13
    IETF 13
    ISO 13
    ITU 13
    W3C 13
Chapitre 2 Liaison série 15
  Codage des données 15
    Code ASCII 15
    Le code EBCDIC 15
    Les codes ISO 16
    Unicode 16
    Autres types de données 16
  Transmission : 17
    Interface RS232/V24 17
    Transmission asynchrone 18
    Interface RS422/V11 18
  Les trames : HDLC, LAPB 19
Chapitre 3 Les réseaux locaux 21
  Ethernet 21
    Version originale 21
    CSMA/CD 21
    Adresses Ethernet 22
    Trame Ethernet 22
    L'Ethernet « mince » 22
    Ethernet sur paires torsadées 23
    Répéteurs, ponts et commutateurs 23
    Evolution : Ethernet 100, Gigabit Ethernet 23
    Ethernet sur fibre optique 23
  Autres protocoles de réseau local 24
    Token Ring 24
    FDDI 24
  La famille 802 24
Chapitre 4 Les modems 25
  Principes élémentaires 25
  Un ancêtre : le Minitel/V23 25
  Le modem moderne : V90 25
  RNIS 26
  L'ADSL 26
  Le câble 26
  Les commandes Hayes 27
Chapitre 5 Architecture de réseau : le modèle OSI 29
  Historique 29
  Le modèle en couches 29

Couche physique : 30
Couche liaison 30
Couche réseau 30
Couche transport 30
Couche session 30
Couche présentation 30
Couche application 31
Exemples historiques 31
X25 31
X400 31
X500 31
L'echec 31
Chapitre 6 Architecture de réseau : TCP/IP 33
Historique 33
Les rfcs 33
Le modèle TCP/IP 33
Acheminement des données : IP 34
La notion de port et de socket 35
Transfert de datagrammes : udp 36
Transfert de flux : tcp 36
Supervision du réseau : icmp 37
Ping 38
Traceroute 38
Ip sur Ethernet 38
ARP 38
Ip sur liaison série 39
ppp 39
Chapitre 7 Le routage 41
Position du problème 41
Structure des adresses IP 41
Architecture interne/externe 42
L'explosion de Internet dans les années 90 43
Epuisement des adresses 43
Explosion des tables de routages 43
Les solutions 43
Panorama des protocoles 44
Classification des protocoles 44
Protocoles externes 45
Protocoles Internes 45
Utilisation d'adresses locales 46
NAT 47
Chapitre 8 Le service de nom 49
Le modèle hiérarchique 49
Fonctionnement 50
Dialogue avec les serveurs 50
Résolutions inverses 51
Configuration élémentaire 51
Chapitre 9 Connexion à distance 53
telnet et rlogin 53
ssh 54
Chapitre 10 Transfert de fichiers 57
ftp 57
scp 59
Chapitre 11 Le courrier électronique 61
Modèle du courrier électronique 61
Les adresses 62
Format d'un courrier : RFC 822 63
Transfert de courrier : smtp 64
Transport de données : Mime 64
Accès au courrier 65

POP ... 65
IMAP ... 65
La confidentialité ... 65
Les logiciels de courrier ... 66
Pine ... 66
Netscape ... 66
Outlook ... 66
Eudora ... 66
Chapitre 12 Les « news » ... 67
Organisation des forums ... 67
Architecture des news ... 67
Logiciels de news ... 68
Netscape ... 68
Outlook Express ... 68
tin ... 68
Archivage des news ... 68
Chapitre 13 Le « web » ... 69
Format d'une page : html ... 69
Désignation des objets : les urls ... 69
Transfert des informations : http ... 70
Les cookies ... 70
Pages dynamiques : JavaScript et Java ... 70
Les lecteurs ... 70
Netscape ... 71
Internet explorer ... 71
Lynx ... 71
Les indexeurs ... 71
Altavista ... 71
google ... 71
Chapitre 14 « X Window » ... 73
Le modèle du MIT ... 73
Le serveur X window ... 74
Les clients ... 74
XDMCP et CDE ... 75
Serveurs X ... 75
XFree86 ... 75
Exceed ... 75
Chapitre 15 Synchronisation des horloges ... 77
Le protocole : ntp ... 77
Configuration simple ... 77
Chapitre 16 Partage des comptes : NIS ... 79
La participation de Sun ... 79
Applications réparties ... 79
Représentation neutre de données : XDR ... 79
Appel d'applications à distance : RPC ... 79
Les « maps » NIS ... 80
Chapitre 17 Partage des disques : NFS ... 83
Principes ... 83
Mise en œuvre ... 84
Chapitre 18 Partage d'imprimantes : lpd ... 85
Chapitre 19 Configuration dynamique des machines ... 87
Chargement de logiciel : TFTP ... 87
Chargement de configuration : BOOTP ... 88
Attribution dynamique d'adresses : DHCP ... 88
Chapitre 20 Administration réseau ... 89
Chapitre 21 Le monde « Windows » ... 93
Hiérarchie : W95/W98/W98SE,W98ME , WNT,W2000 ... 93
Accès aux ressources réseau ... 93
Windows et Unix : SAMBA ... 93
Connection d'une machine Windows ... 93

En réseau local 93
A distance 94
Chapitre 22 Le monde « MacIntosh » 95
AppleTalk et Unix : 95
Connection d'une machine MacIntosh 95
En réseau local 95
A distance 95
Appendice : Principaux RFCs 97
Références 99
Index 104

Table des illustrations

Figure 1 : Code ASCII 7 bits (CCITT n°5) 15
Figure 2 : Interface RS232/V24 18
Figure 3 : Trame LAPB 19
Figure 4 : Trame Ethernet 22
Figure 5 : Les 7 couches du modèle OSI 29
Figure 6 : Architecture TCP/IP 34
Figure 7 : Format d'un paquet IP 35
Figure 8 - Le « three way handshake » 37
Figure 9 - Client-serveur 53
Figure 10 - Circuit de données ftp 58
Figure 11 - Modèle pour le courrier électronique 62
Figure 12 - Format d'un message 63
Figure 13 - Le modèle X Window 73
Figure 14 - Principes de SNMP 90

Introduction

Ce petit fascicule sert de support de cours à une introduction au réseaux informatiques. Il survole de nombreux points de ce domaine sans la prétention de les approfondir.

Après une introduction historique dans le chapitre 1 ; une première partie (chapitres 2 à 6) présente les notions de base : codage et transmission de données, modems, réseaux locaux, modèle OSI et TCP/IP.

Sont ensuite abordés deux services indispensables au fonctionnement de l'Internet : le nommage et le routage (chapitres 7 et 8).

Les applications élémentaires : connexion à distance, transfert de fichiers, courrier, « news », « Web » sont ensuite abordés (chapitres 9 à 13).

Les services que l'on utilise abondamment en réseau local (même si certains sont aussi utilisés dans d'autres circonstances) sont étudiés dans les chapitres 14 à 19 : X Window, service de temps, comptes réseau, partage de disques et d'imprimantes, configuration dynamique des machines.

Le chapitre 20 est une introduction à l'administration des réseaux.

Les deux derniers chapitres sont plus spécialement consacrés à l'insertion des machines du monde Windows et MacIntosh en réseau.

Chapitre 1 Qu'est-ce qu'un réseau ?

Historique

La nécessité de transmettre des données remonte à un temps très ancien. Des moyens de plus en plus sophistiqués se sont succédés : signaux optiques ou sonores divers, puis signaux électriques et ondes électromagnétiques. L'apparition des ordinateurs, avec leur capacité de traitement de l'information, a vu se développer une nouvelle discipline : les réseaux informatiques.

Le temps des « mainframes »

Dans les années soixante, le paysage informatique se compose de machines rares car très coûteuses. La nécessité de rentabiliser l'utilisation de ces ordinateurs, ainsi que le besoin de collecter un nombre croissant d'informations entraîne le développement des premières forme de réseau informatique : d'abord les « terminaux lourds », constitués en général d'une imprimante et d'un lecteur de cartes perforées reliés à l'ordinateur central par une paire de modems et une liaison téléphonique ordinaire ou permanente (liaison spécialisée).

Les ordinateurs de l 'époque constituent une véritable tour de Babel. Tous utilisent non seulement des architectures différentes, des systèmes d'exploitation différents ; mais aussi des codages de données différents. Toute communication entre ordinateurs de constructeurs différents (voire même entre deux gammes d'un même constructeur) est quasiment impossible.

Le marché est alors dominé de très loin par un seul constructeur : IBM.

Les réseaux d'ordinateurs utilisent des protocoles liées à chaque constructeur : SNA pour IBM, Decnet pour Digital Equipment (Dec, racheté il y a 3 ans par un constructeur de PCs texans), …

La tentative de l'OSI

Cet état de fait rendait la concurrence entre constructeurs quasi impossible : un organisme équipé à l'aide du matériel d'une marque était condamné, s'il ne voulait pas renoncer à tout son investissement précédent, à rester dans le giron de ce même fabricant.

Au début des années 80, l'organisme de normalisation international, l'ISO, entreprend d'établir une série de normes pour réaliser des réseaux informatiques permettant d'interconnecter des ordinateurs de toutes marques : l'Open System Interconection : OSI.

Le première démarche de l'OSI, à savoir la création d'un modèle de référence en 7 couches est un succès bien connu. La normalisation des première couches du modèle est aussi un succès avec X25 et voit l'apparition des premiers réseaux nationaux permettant l'interconnexion d'ordinateurs de fabricants différents (Transpac, …).

Ces succès entraînent des choix très important de la part des administrations, grosses consommatrices de produits informatiques, avec en particulier le choix du gouvernement américain imposant les normes OSI à tous les contractants de l'administration (GOSIP : Government OSI Profiles).

Les tentatives pour la normalisation des couches supérieures vont par contre s'embourber dans un cadre administratif trop rigide et coupé de la réalité.

Au début des années 90, l'échec total des normes d'administration de réseau (CMIP/CMIS) entraîne un abandon de nouveau développement du modèle OSI.

L’apparition des réseaux locaux

Au début des années 70, les « mini-ordinateurs » ont rendu l’informatique plus abordables et ce type de machine commence a se répandre dans les services des entreprises et les laboratoires universitaires. Trois grands de l’informatique : Digital, Intel et Xerox unissent leurs efforts pour développer un protocole de communication rapide entre des ordinateurs peu éloignés : Ethernet. Repris comme norme par l’IEEE sous la dénomination de 802.3, ce protocole va permettre quelques années plus tard la création de réseaux locaux . L’existence de ce support rapide (10 mégabits/s) va permettre le développement d’applications nouvelles : partage des fichiers d’administration des machines (NIS) et partage des disques (NFS) développés par Sun, affichage graphique à distance (X-Windows) développé au MIT. Après plusieurs évolutions, ce protocole est maintenant largement utilisé comme architecture pour constituer les réseaux d’entreprise, de campus, etc…

La naissance de l’Internet

A la fin des années 70, sous l’impulsion du département de la défense américain, le réseau Arpanet expérimente une série de protocoles destinés à l’interconnexion d’ordinateurs sur tout le territoire des Etats-Unis. Cela conduit à la série des protocoles TCP/IP, décrits dans une série de documents nommés RFCs (Request For Comment), qui va servir de base à l’Internet. En 1986, la NSF (National Science Fondation) fonde les bases d’un réseau utilisant ces protocoles et reliant les laboratoires américains, auxquels se joignent bientôt des universités de nombreux pays.

Cette initiative sert de base à la structuration de réseaux soit disciplinaires (HEPNET pour la physique des hautes énergie par exemple), soit nationaux (Renater), soit continentaux (Ripe : Réseaux IP Européens).

L’explosion du Web

En 1989, des chercheurs du Cern (dont Tim Berners-Lee) cherchent un moyen universel pour constituer des documents qui se réfèrent entre eux. Ils inventent l’URL (Uniform Resource Locator) pour désigner ces références, HTML (HyperText Markup Langage) pour décrire ces documents et HTTP (HyperText Transport Protocol) pour les véhiculer. L’ensemble constitue les bases du Wold Wide Web.

Parallèlement, l’explosion du nombre d’ordinateurs personnels, la baisse du coup des modems et des lignes créent un terrain favorable au développement d’entreprises cherchant à tirer un profit financier de ce marché naissant.

10 ans plus tard, l’accès au « Web » est banalisé et se trouve communément aussi bien chez le particulier que dans les écoles. Il est devenu la base de travail de nombreuses équipes et le support de vente ou de promotion de nombreuses entreprises.

Qu’est-ce qu’un réseau ?

Il s’agit de l’ensemble des techniques permettant d’interconnecter des ordinateurs entre eux, ou de les relier à un ensemble plus vaste, l’Internet.

Il peut s’agir :

- de relier un ordinateur isolé au reste du réseau par un modem et une ligne téléphonique (ou un autre moyen) en passant éventuellement par un fournisseur d’accès internet (FAI).
- d’équiper un laboratoire en postes de travail informatique.
- d’installer sur ce réseau de laboratoire des services communs : disques partagés, imprimantes, etc…
- de relier l’ensemble des machines d’un laboratoire au reste de l’Internet par l’intermédiaire d’un réseau, par exemple Renater.

- …

Les services offerts par ce réseau peuvent être de nombreux types :

- connexion à des ordinateurs distants (serveurs d'un centre de calcul par exemple)
- transfert de fichiers entre différents ordinateurs
- accès à des disques partagés
- accès à des imprimantes partagées
- accès au courrier électronique
- accès au service de « news »
- accès au « Web »
- …

Les organismes de normalisation

Les différents protocoles utilisés dans le monde des réseaux sont standardisés par une certain nombre d'organismes de différents statuts :

ANSI

American National Standards Institute : Organisme américain constitué de représentants des groupes industriels et coordonnant le travail de normalisation. Il représente les Etats Unis à l'ISO.

IEEE

Institute of Electrical and Electronics Engineers : Association d'ingénieurs, américains au début mais maintenant de tous horizons connus pour ses travaux de normalisation dans le domaine des réseaux locaux (famille des protocoles IEEE802).

IAB

Internet Architecture Board : Le bureau exécutif de l'IETF :

IETF

Internet Engineering Task Force : Organisme constitué de volontaires pour promouvoir le développement de l'Internet. Ce sont eux qui éditent les RFCs.

ISO

International Organization for Standardization : Organisme international chargé de coordonner les travaux de normalisation dans le monde. Ils normalisent tout, de la taille des boulons au réseaux informatique. L'OSI est une de leur émanation.

ITU

International Telecommunication Union : Organisation internationale chargé de coordonner les travaux en téléphonie et en radio dans le monde. A la fin de l'aventure de l'OSI, ils ont intégré le CCITT (Comité Consultatif International des Télégraphes et Téléphones) qui était chargé de rédiger les normes OSI. Ils sont associés au normes de la série V (interfaces, modems) et X (OSI).

W3C

World Wide Web Consortium : Organisme de création récente (1994) qui rédige les normes pour l’évolution du Web : HTML , XML, …

Chapitre 2 Liaison série

Codage des données

Toute transmission de données sous forme numérique suppose au préalable une convention pour la représentation de ces données : une table de codage. Historiquement, plusieurs codes ont été utilisé, en particulier dans les années 70, chaque constructeur utilisait son propre code. Les codages actuellement utilisés sont :

Code ASCII

Codes signifiant American Standard Code for Information Interchange, puis repris comme code international sous le nom de code CCITT n° 5. Les caractères sont codés sur 7 bits, ce qui donne 128 codes possibles, dont 33 pour des codes de « contrôle » (saut de page, saut de ligne, « beep », contrôle de flux, etc…) et 95 pour les caractères. On notera que le nombre de codes disponibles ne permet pas le codage des caractères diacritiques. Une extension connue sous le nom « ASCII 8 bits » rajoute 96 caractères (et 32 codes de contrôle) pour représenter quelques caractères nationnaux.

	0	1	2	3	4	5	6	7
0	NUL	DLE		0	@	P	‘	p
1	SOH	XON	!	1	A	Q	a	q
2	STX	DC2	¨	2	B	R	b	r
3	ETX	XOFF	#	3	C	S	c	s
4	EOT	DC4	$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	‘	7	G	W	g	w
8	BS	CAN	(	8	H	X	h	x
9	HT	EM	)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[	k	{
C	FF	FS	,	<	L	\	l	|
D	CR	GS	-	=	M	]	m	}
E	SO	RS	.	>	N	^	n	~
F	SI	US	/	?	O	_	o	DEL

Figure 1 : Code ASCII 7 bits (CCITT n°5)

Le code EBCDIC

Le dernier survivant des codes « propriétaires ». C'est le « Extended Binary Coded Decimal Interchange Code » d'IBM qui est encore utilisé sur les systèmes centraux et les mini-ordinateurs de ce constructeur, ainsi que dans une partie de sont protocole de réseau : SNA. Les caractères sont codés sur 8 bits, donc avec 256 possibilités, mais de nombreux codes sont inutilisés et l'accès au caractères diacritiques de fait par un système compliqué de changement de jeu de caractères.

Les codes ISO

L'organisation de normalisation ISO a proposé une série de codes adaptés aux langages nationnaux. Il s'agit de la série des codes ISO-8859. Le jeu 8859-1 reprend les codes « ASCII 8 bits » et il existe ainsi 15 jeux 8859-1 à 8859-15 permettant l'expression des caractères de la plupart des langues, à l'exception notable des langues asiatiques. Le dernier en date, 8859-15 a été créé spécialement pour introduire le signe de l'Euro.

Unicode

Pour pouvoir représenter l'ensemble des caractères nationaux avec un seul jeu de caractères, 8 bits ne sont plus suffisants. On utilise donc 16 bits pour représenter jusqu'à 65536 caractères découpés en 256 « pages » de 256 caractères chacune. La première page reprends le code ISO 8859-1. Ce code a été adopté pour encoder les chaînes de caractères du langage Java et des bibliothèques de fonctions permettant de l'utiliser existent maintenant dans les langages de programmation courants.

Autres types de données

Les images

Les images sont représentées au départ comme un tableau bi-dimentionnel de point indiquant les valeurs de couleurs. Ces tableaux sont ensuite compressés par divers algorithmes. Les plus utilisés sont :

- GIF Graphics Interchange Format. Les images sont réduites à une table de 256 couleurs et compressées à l'aide d'un algorithme non destructif (Lempel-Ziev)
- JPEG : Join Photographic Exchange Group. Ce format utilise une série d 'algorithmes complexes prenant en compte la physiologie de la vision humaine. Un taux de compression élevé est possible, mais cette compression est destructive.

Les sons

Les son présentent plus de difficulté dans leur codage et dans leur transmission du fait de la necessité de maintenir un flot continu de données, sans « blancs ». De très nombreux formats existent, avec de multiples variantes pour chaque : AIFF, WAW, etc… Un format particulièrement efficace rencontre un très grand succès pour la diffusion, plus ou moins légale, d'œuvres musicales : le MP3.

Les vidéos

La transmission de séquences vidéo représente la difficulté maximale, avec la contrainte d'un volume important de données devant être acheminées avec des contraintes de temps très fortes pour maintenir une animation fluide. Plusieurs formats existent, dont le format MPEG, normalisé. On peut aussi citer le format QuickTime d'Apple, dont la version 4 sert de base à la dernière version de MPEG.

Transmission de fichiers binaires

Le passage de fichiers binaires (dont les images, son,… mais aussi des fichiers applicatifs tels que des documents Word,…) peuvent subir des altérations irréversibles s'ils transitent sur des moyens de transmission adaptés au codes ASCII, tels que le courrier électronique. Plusieurs moyens existent pour transformer ces fichiers dans un format adapté (ramené au jeu de caractère minimal ASCII et avec des lignes courtes) :

- uuencode : le standard historique de Unix
- binhex : le standard du monde Apple
- base 64 : le standard actuel de l'Internet

Transmission :

Une fois les données encodées dans une représentation binaire, il reste à déterminer un protocole permettant de les transmettre sur un support physique. Les réseaux informatiques utilisent une vaste variété de supports de transmission :

- signaux électriques sur des paires métalliques ou des câbles coaxiaux.
- signaux électromagnétiques sur différentes plages de longueur d'onde, des ondes radio aux micro-ondes.
- signaux lumineux dans des fibres optiques.

Parmi celles-ci, la transmission des données sur des fils métalliques en envoyant les bits les uns après les autres a été une des premières utilisées et est encore largement employée. Elle est connues sous le nom de transmission série. Deux variantes sont utilisée en fonction du débit à transmettre et de la distance à franchir.

Les débits sont mesurés en bits par seconde (bit/s), à ne pas confondre avec les bauds, unité de rapidité de modulation (un modem moderne transmet 33,6Kbits/s avec une rapidité de modulation de 2400 bauds !).

Interface RS232/V24

L'interface de liaison série la plus simple est l'interface RS232 (américaine) ou son équivalent dans les standards internationaux, V24. L'état de la liaison (bit à 0 ou à 1) est signalé par la tension électrique, l'état « 1 » est signalé par une tension comprise entre 5 et 15 Volts. Tous les signaux sont repérés par rapport à une même référence de tension : la terre de signalisation (signal ground). L'interface comprend deux signaux pour la transmission et la réception de données, ainsi que plusieurs signaux de supervision :

- des signaux d'horloge destinés au cadencement des données.
- des signaux destinés au contrôle de flux : RTS (Request To Send, demande pour émettre), CTS (Clear To Send, prêt à émettre).
- des signaux d 'état : DTR (Data Terminal Ready, terminal de données prêt), DSR (Data Set Ready, poste de données prêt)
- signaux d'appel : CD (Carrier Detect, détection de porteuse), RI (Ring Indicator, indicaterur d'appel)

Physiquement, cette interface se présente sous la forme d'une prise à 25 broches : DB25 (comportant de nombreux autre signaux dont la plupart sont inutilisés), mais plusieurs variantes existent : prises 9 broches : DB9 utilisées sur les « PCs » ou prises « mini-DIN » utilisés sur les anciens MacIntosh.

DB25	DB9	CCITT V24	Français	Anglais
1		101	Terre de protection	Protective ground
2	2	103	Emission de données	Transmitted data
3	3	104	Reception de données	Received data
4	7	105	Demande pour émettre	Request to send (RTS)
5	8	106	Prèt à émettre	Clear to send (CTS)
6	6	107	Poste de données prèt	Data set ready (DSR)
7	5	102	Terre de signalisation	Signal ground
8	1	109	Détection de porteuse	Carrier detect (CD)
15		114	Horloge émission interne	Transmit clock
17		115	Horloge externe	Receive clock
20	4	108	Terminal de données prèt	Data terminal ready (DTR)
22	9	125	Indicateur d'appel	Ring indicator (RI)

Figure 2 : Interface RS232/V24

L'interface RS232 est destinée à la transmission de données à des débits faibles et sur de courtes distances. La norme initiale prévoit des débites limités à 19200 bits/s, mais elle est couramment utilisées jusqu'à la limite des transmetteurs série des « PCs » : 115000 bits/s. La distance utilisable varie de quelques dizaines de mètres à 9600 bits/s à un mètre aux débits les plus élevés.

Transmission asynchrone

Cette interface est souvent utilisée pour transmettre des données dans un mode dit « asynchrone ». Dans ce mode, les données sont envoyées caractère par caractère sans signaux d'horloge pour synchroniser les équipements : un bit (« start ») est utilisé pour indiquer le début de la transmission. Les bits de données suivent, puis un éventuel bit de parité (permettant de détecter une éventuelle erreur de transmission) et enfin un bit (« stop ») marquant la fin du caractère. Ce type de transmission est caractérisé par :

- le nombre de bit de données (7 ou 8)
- le nombre de bits « stop » (1 ou 2)
- le type de contrôle de parité (even/pair, odd/impair, none/sans)

Historiquement, de nombreux réseaux ont étés basés sur ce type de transmission en utilisant un ensemble d'équipements adaptés :

- terminaux asynchrone de type VT100
- autocommutateurs de données
- multiplexeurs statistiques
- logiciels de transmission de données tels que Kermit

Interface RS422/V11

L'interface RS232 n'est pas utilisable à des débits élevés, principalement à cause de son utilisation d'une référence de tension commune à tous les signaux, entraînant une diaphonie élevée et donc des erreurs de transmissions. L'interface RS422 (au Etats-Unis) ou V11 (norme internationale) pallie cet inconvénient en utilisant pour chaque circuit de données la différence de tension entre deux fils :

- Va-Vb < -0,3 volt : bit à 1
- Va-Vb > 0,3 volt : bit à 0

Deux circuits transmettent les horloges permettant de synchroniser les signaux. Enfin, les signaux de contrôle, ainsi que des signaux indicateurs, sont transmis de la même manière sur deux autres circuits.

Cette interface se présente physiquement sous la forme d'une prise à 15 broches (DB15). Elle permet la transmission de données sur une dizaine de mètres à un débit de 10Mbits/s, ou sur un kilomètre à un débit de 100Kbits/s.

Les trames : HDLC, LAPB

Le mode asynchrone (transmission caractère par caractère) est bien adapté à la connexion d'un terminal tel qu'on les utilise pour accéder à un ordinateur distant en mode « caractère », par exemple la connexion à un « shell » sous Unix. Ce mode est toutefois très limité en débit (de l'ordre de 9600bits /s), mais aussi en efficacité. Les liaison séries à débit élevé transmettent leurs données sous forme de segments structurés (trames ou datagrammes). Cette structure doit permettre :

- de transmettre des commandes liés au fonctionnement de la liaison (établissement, fermeture, gestion des incidents,…).
- de multiplexer plusieurs communications sur la même liaison.
- de contrôler le flux de données de manière à éviter un engorgement.
- de contrôler de l'intégrité des donnée.

Plusieurs protocoles ont étés successivement proposés pour standardiser la transmission de tels segments de données entre deux équipements de communication. Ils sont tous dérivés d'un protocole IBM : SDLC (Synchronous Data Link Control). La version largement utilisée actuellement est nommée LAPB (Link Access Protocol version B) dérivée de HDLC (High level Data Link Control).

Ces protocoles utilisent des trames délimitées par des séries de « fanions » (suite de bits : «01111110» émis en permanence sur la liaison). Les données utiles sont accompagnées de données de commande (2 octets) et d'une somme de contrôle (CRC : Cyclic Redondancy Check) destiné à détecter d 'éventuelles erreurs de transmission.

flag	addr	cmd	données	crc	flag

Flag : fannion
Addr : selecteur d'adresse
Cmd : champ de commande
Crc : somme de contrôle

Figure 3 : Trame LAPB

Chapitre 3 Les réseaux locaux

A la fin des années 70, avec le développement des ordinateurs départementaux dans les campus et les laboratoires de recherche des entreprises, le besoin d'une méthode de communication rapide entre ces équipements est apparu. Trois constructeurs : Digital Equipment (DEC), Intel et Xerox unissent leurs efforts pour développer un tel protocole. Le résultat a été le protocole de réseau local Ethernet. Ce protocole a été repris comme norme par l'IEEE sous le nom IEEE 802, puis par l'ISO sous le nom ISO 8802 (il y a en fait une légère différence entre ethernet et les protocoles 802).

Ethernet

Ethernet est une méthode de transmission de données à haut débit (10Mbits/s dans sa version originale) et sur des distances compatibles avec la taille d'un campus universitaire (1km). Depuis sa version initiale en 1978, il a connu de nombreux développements, d'abord sur son support (« gros cable », puis cable 50 ohms, puis paires torsadées), puis sur son débit (ethernet 100Mbits/s , puis Gigabit Ethernet).

Version originale

Dans sa version historique, Ethernet repose sur un câble coaxial d'un demi pouce de diamètre (1,27cm). L'accès au câble se fait par un boîtier (« tranceiver ») fixé au câble par un sertissage qui pénètre jusqu'à l'âme par une aiguille (« prise vampire »). Le boîtier est relié à la station par un câble (« drop cable ») utilisant des prises à 15 broches (« prises DIX »).

Les données binaires sont traduites sur le câble par des transitions entre les états +0,85 Volts et –0,85 Volts à une vitesse de 10 Mbits/s (codage Manchester).

Cette version est aussi connue sous le nom 10Base5. Chaque brin est limité à 500 mètres de long, avec au maximum 100 connexions.

Les brins peuvent être interconnectés au moyen de répéteurs, simples circuits électroniques qui propagent le signal sur le brin adjacent. L'ensemble du réseau est limité à 5 brins.

Les deux extrémités du câble sont terminées par des « bouchon » d'adaptation d'impédance de 50 ohms destinés à éviter les réflexions.

L'accès au câble est régi par un protocole nommé CSMA/CD : Carrier Sence Multiple Acces/ Collision Detection) qui permet à plusieurs centaines de stations de cohabiter sur un support unique.

CSMA/CD

Avant d'émettre, une station « écoute » le câble. Si elle détecte une émission, elle attend la fin de celle-ci. Sinon, elle émet (CSMA). Dans le cas où, à cause des délais de propagation, deux stations émettent simultanément, la première qui détecte le signal de l'autre émet sur le cable un signal de brouillage signalant la collision. Les deux stations attendent alors un délai aléatoire avant de tenter de ré-emettre. En cas de collisions successives, le délais avant ré-emission est augmenté exponentiellement.

Ce mode de fonctionnement implique des contraintes sur la topologie du réseau :

- longueur maximale pour éviter des temps de propagation trop longs
- longueur minimale pour les trames émises
- nombreuses temporisations : temps de brouillage, délai inter-trame, etc…

Adresses Ethernet

Chaque station Ethernet est identifiée par une adresse unique attribuée au circuit électronique qui contrôle l'accès au câble. Ces adresses comportent 6 octets :

- les 3 premiers identifient le constructeur du circuit, par exemple « 0xAA0004 » correspond à Dec
- les 3 suivants sont distribuées de manière unique par chaque constructeur.

A noter qu'une série d'adresses est destinée à identifier des destinataires multiples, ce sont les adresses de « multicast ».

Enfin, l'adresse particulière avec tous les bits à 1 (« broadcast ») désigne tous les équipement connectés au cable.

Les adresses Ethernet sont conventionnellement notées sous la forme de 6 nombres hexadécimaux séparés par des « : », par exemple « AA :00 :04 :12 :23 :34 » .

Trame Ethernet

La trame Ethernet comprent successivement :

- un préambule de 8 octets permettant au récepteur de synchroniser son horloge.
- 6 octets avec l'adresse de destination.
- 6 octets avec l'adresse de l'émetteur.
- 2 octets contenant le type de trame (le protocole transporté)
- les données proprement dites, de 46 à 1500 octets.
- une somme de contrôle sur 4 octets.

préambule	dest	source	type	données	crc

Figure 4 : Trame Ethernet

Il existe une différence entre « Ethernet » et la norme 802 :

- en Ethernet, le champ « type » contient le type de protocole transporté, par exemple « 0x0800 » pour IP.
- en 802, ce champs contient la longueur des données. Le type de données, ainsi que des informations supplémentaires analogues au champ de commande de la trame LAPB occupent 8 octets du champs de données qui est maintenant limité à 1492 octets.

Les « types » Ethernet commencent au-delà de 1500, ce qui permet de reconnaître les trames des deux protocoles.

L'Ethernet « mince »

Le câble Ethernet original était coûteux et difficile à mettre en œuvre (rigidité élevée, prises délicates à poser, configuration difficile à modifier). Il a eut comme successeur l'Ethernet sur câble mince, plus connu sous le nom de 10Base2.

On utilise ici du câble 50 ohms standard (RG58) avec la connectique correspondante. Les stations sont alors reliées au câble par de simples raccords en T, facile à brancher. Le réseau est ainsi beaucoup plus simple à reconfigurer.

Chaque brin est limité à 185 mètres et à 30 stations et on peut ici aussi interconnecter jusqu'à 5 brins à l'aide de répéteurs.

Ethernet sur paires torsadées

L'Ethernet sur câble coaxial présente un grave inconvénient : un incident sur un brin se répercute sur l'ensemble du réseau. Dans le cas d'un réseau étendu, cela peut entraîner des pannes très difficiles à diagnostiquer.

D'autre part , les câbles coaxiaux restent coûteux, alors que de nombreux bâtiments sont équipés d'un câblage téléphonique sur paires torsadées. Ces considérations, ainsi que les progrès en traitement du signal, ont entraîné le développement de l'Ethernet sur paires torsadées, aussi connu sous le nom de 10BaseT.

Le réseau change dans ce cas de topologie : au câble coaxial qui constitue l'ossature d'Ethernet se substitue des concentrateurs (« hub ») auxquels les stations sont reliées par des câbles de type téléphonique en paires torsadées à l'aide de connecteurs de type RJ45. Ces concentrateurs sont à leur tour reliés par une connexion ethernet, soit coaxiale, soit sur paires.

Répéteurs, ponts et commutateurs

Comme on l'a vu, les brins Ethernet peuvent être interconnectés par des répéteurs permettant d'étendre le réseau. Cependant, on est limité par le fait que toutes les trames se propagent sur l'ensemble du réseau, ce qui limite sa taille à cause du risque grandissant de collision.

L'utilisation de ponts (« bridges ») à la place de répéteurs permet d'aller au-delà : ces équipement ne propagent pas les trames dont le destinataire se trouve sur le même brin. On obtient ainsi un cloisonnement du réseau où seul le trafic de transit est passé au brins adjacents. On peut alors construire des réseau comportant plusieurs milliers de machines a condition de choisir soigneusement la topologie.

Le réseau constitué à l'aide de ces ponts pouvant comporter des boucles pour des raisons de redondance, il est nécessaires d'avoir un protocole leur permettant de communiquer entre eux pour qu'il se transmettent des informations sur la topologie de leur interconnexion. Ce protocole est connu sous le nom de « spanning-tree ».

Pour aller au-delà, on peut mettre en œuvre des commutateurs Ethernet, physiquement analogues au concentrateurs, mais disposant d'un logiciel leur permettant d'acheminer une trame directement sur l'interface ou est connectée l'adresse de destination. Un tel commutateur à 16 voies peut ainsi acheminer simultanément 8 communications à 10 (100) Mbits/s entre 8 paires de machines connectées.

Evolution : Ethernet 100, Gigabit Ethernet

Les progrès en traitement du signal, la diminution des circuits spécialisés et l'amélioration de la qualité du câblage ont permis d'étendre Ethernet à un débit de 100Mbits/s sur paires torsadées (100BaseT). Ce débit est maintenant celui proposé par la plupart des équipement, y compris les « PCs » de bas prix.

La débit maximal accessible actuellement est de 1Gbit/s sur des fibres optique ou, sur de courtes distances, sur des paires torsadées (Gigabit Ethernet).

Ethernet sur fibre optique

Pour des besoin spécifiques (interconnexion de bâtiments ou liaison à grande distance), la norme 10BaseFL permet de réaliser des connexion Ethernet sur des fibres optiques à des distances allant jusqu'à 2000m.

Autres protocoles de réseau local

Token Ring

En 1985, IBM annonce son choix pour son protocole de réseau local : l'anneau à jeton (token ring). Contrairement à Ethernet, basé sur des considérations statistiques, il s'agit d'un protocole dit « déterministe » permettant en principe de garantir l'accès au réseau.

Ce réseau est constitué d'un anneau reliant les différentes machines entre elles. Les données sont insérées dans des trames qui parcourent l'anneau de station en station jusqu'à atteindre la station destinataire.

Dans sa première version, les données étaient acheminées à 4Mbits/s, puis 16 par la suite.

L'accès à l'anneau est contrôlé par un « jeton » qui circule en permanence quand l'anneau est au repos. Un station voulant émettre prend le jeton et émet sa trame. Celle-ci est alors acheminée sur l'anneau jusqu'à sa destination. A réception de la trame, la station destinataire remet en circulation un jeton qui permettra à une autre station d'utiliser l'anneau.

Bien que la topologie théorique de ce réseau soit un anneau, il a plutôt d'une topologie en étoiles centrées sur des « MAU » (Media Access Unit) permettant l'insertion de nouvelles stations sans rupture de l'anneau.

FDDI

Conçu au milieu des années 80, FDDI (Fiber Distributed Data Interface) fut à cette époque le premier réseau à haut débit. Il est basé sur un double anneau de fibres optiques permettant une forte tolérance aux pannes. Comme Token Ring, il est basé sur la possession d'un jeton pour pouvoir émettre une trame.

Les données sont acheminées à 100Mb/s et le réseau est prévu pour avoir de l'ordre du kilomètre de dimension. Une version utilisant des fibres mono-mode et des lasers à la place des diodes permet de constituer des réseaux de plusieurs dizaines de kilomètres.

Très coûteux et dépassé en performances par les évolutions d'ethernet, ce protocole est maintenant réservé à des utilisations spécifiques (résistance au pannes).

La famille 802

De nombreux protocoles de réseaux locqux sont regroupés dans la famille IEEE 802, dont Ethernet et token-ring :

- 802.1 : spécifie les interfaces
- 802.2 : spécifie le LLC : Logical Link Control, c'est à dire le format et l'utilisation des champs qui se trouvent au début des données.
- 802.3 : CSMA/CD
- 802.4 : le « token-bus », un protocole très utilisé dans les réseaux industriels
- 802.5 : le « token-ring »
- 802.6 : DQDB : Distribured Queue Dual Bus, un protocole basé sur un double bus et destiné à la réalisation de réseaux urbains à haut débit : Metropolitean Area Network (MAN).
- 802.11 : protocole de réseau local utilisant des ondes radio dont une des premières réalisation courante est le réseau « Airport » d'Apple.
- 802.14 : le protocole utilisé pour l'accès par le cable

Chapitre 4 Les modems

Les liaison séries et les réseaux locaux ne permettent que des communications sur des distances courtes et dans le cadre d'un domaine privé. Toute liaison entre divers domaines privés doit passé par un opérateur de transmission tel que France Telecom. Il est alors nécessaire de transformer le signal numérique en un signal analogique acceptable par une liaison téléphonique. C'est le rôle des modems : MOdulateur/DEModulateur. Ces équipements ont connus des développements spectaculaires dans les 15 dernières années, passant de vitesses de l'ordre de 300bits/s à 56Kbits/s sur les liaisons téléphoniques classique, et à des vitesses de l'ordre de 1Mbit/s sur des supports tels que l'ADSL ou le câble.

Principes élémentaires

Pour pouvoir transiter par le réseau téléphonique, un signal doit être mis sous forme d'une série d'ondes, seules capables de se propager sur de longues distances (un signal numérique se déforme très rapidement à cause des vitesses de propagations aux différentes fréquences). Les techniques utilisées sont similaires à celles utilisées en radio :

- modulation d'amplitude
- modulation de phase
- modulation de fréquence

A cela s'ajoutent des techniques spécifiques au support téléphonique tel que l'annulation d'écho.

Un ancêtre : le Minitel/V23

Dans le but de développer un terminal peu coûteux accessible à tous, France Telecom développa au début des années 70 un modem le plus simple possible réalisable avec les techniques de l'époque : le modem V23, répandu à des millions d'exemplaires dans les Minitels.

Le modem fonctionne en utilisant simplement deux fréquences : 2100Hz=bit à 0, 1300Hz=bit à1. On dispose ainsi d'une voie de données à 1200bits/s dans le sens réseau -> abonné. Deux fréquences supplémentaires : 450Hz et 390Hz permettent une voie à 75bits/s dans le sens abonné -> réseau (bien suffisante vue la qualité du clavier).

Le modem moderne : V90

Le premier modem moderne fut le modem V32 (fin des années 80) qui introduisait les techniques qui allaient permettre le développement d'une nouvelle génération :

- la négociation des paramètres : les modems échangent une séries de signaux prédéfinis leur permettant de juger de la qualité de la ligne et d'optimiser leurs paramètres de fonctionnement (cette négociation obéit à une norme : V8)
- le codage en treillis : en utilisant des combinaisons de phase/amplitude prédéfinies (par exemple 32 pour le modem V32, un millier pour le V34bis), on transmet des combinaisons de bits permettant d'optimiser l'utilisation de la ligne.
- l'annulation d'écho qui limite les perturbations que sa propre émission créée pour un modem.

Avec une rapidité de modulation de 2400 bauds et un codage à 32 moments, le modems V32 permettait de transmettre à 9600bits/s sur le réseau téléphonique.

Il eut une série de successeurs qui aboutirent au modem V34bis permettant une transmission à 33600bits/s, soit le maximum que permet le théorème de Shanon avec les caractéristiques du réseau téléphonique.

L'évolution ultime est le modem V90 qui a un fonctionnement dissymétrique : dans le sens réseau -> abonné, il fonctionne en utilisant une transmission numérique pouvant aller jusqu'à 56Kbits/s (ce qui suppose que l'autre extrémité, le fournisseur d'accès, soit équipé de liaisons numériques), et dans le sens abonné->réseau, il utilise le protocole V34bis à 33600bits/s.

Il est à noter que dans la phase de négociation, les modems peuvent choisir une vitesse de transmission inférieure au maximum permis si la qualité de la ligne n'est pas suffisante.

Enfin, ces modems intègrent deux autres techniques :

- V42 permettant la détection et la correction des erreurs de transmission.
- V42bis permettant la compression de données.

RNIS

Le réseau téléphonique (Réseau Numérique à Intégration de Services)étant maintenant entièrement numérique, il devient possible d'utiliser ces nouvelles possibilités pour aller au-delà des performances du modem V90.

Un abonnement numérique standard (dit S0) comporte deux circuits de données à 64Kbits/s, dits canaux B et servant pour chaque sens et un canal de supervision à 16Kbits/s, dit canal D servant à acheminer des données telles que le numéro de l'appelant ou un indicateur d'appel.

Ces abonnements numériques sont parfaitement adaptés aux connexions informatiques :

- vitesse de 64Kbits/s dans les deux sens et dans tous les cas.
- numérotation rapide par le canal D.

Des équipement d'un prix abordable utilisent ce type de connexion en établissant automatiquement l'appel en cas de trafic pour le libérer en cas d'inactivité.

L'ADSL

Le réseau téléphonique n'est pas adapté à la transmission de données à haut débit de par sa structure. Toutefois, sur sa partie terminale, entre le central et l'abonné, les câbles métalliques utilisés peuvent transmettre, à condition d'utiliser toutes les techniques du traitement du signal qui ont été développées pour les modems, des débits très élevés.

C'est ce qui est utilisé dans l'ADSL (Asymetric Digital Subscriber Line) : au niveau du central et chez l'abonné, un filtre sépare les signaux en une partie basse fréquence destinée au téléphone et une partie haute fréquence destinées aux données. Sur cette partie haute fréquence, les techniques précédemment citées permettent de transmettre des données à des vitesses de l'ordre de 1Mbits/s (variable suivant le type d'abonnement et la distance au central). Pour une utilisation chez un particulier, cette vitesse est asymétrique, avec un débit dans le sens abonné->réseau de l'ordre de 64Kbits/s.

Arrivé au central, les données numériques sont alors acheminées par un canal autre que le réseau téléphonique, en utilisant les services numériques à haut débit tels que ATM .

Le câble

Le déploiement des réseau de télévision câblés a entraîné la mise en place d'un câblage de la plupart des habitations urbaine à l'aide de câbles coaxiaux permettant de transmettre de très hauts débits. En utilisant une technique appelée modulation large bande, la bande passante du câble (entre 40 et 550MHz) est découpée en bandes de 6MHz permettant d'acheminer soit une chaîne de télévision, soit une connexion numérique à haut débit.

Le modem câble placé chez l'abonné utilise un de ces canaux pour recevoir et transmettre des données. Il se connecte en général à l'aide d'une interface Ethernet sur un réseau local placé chez l'abonné.

Il faut noter que, le réseau câble étant conçu pour un flot important dans le sens réseau->abonné, la vitesse de connexion est très dissymétrique.

Les commandes Hayes

Il est en général nécessaire de passer un certain nombre de commandes à un modem pour lui transmettre les paramètres de fonctionnement voulus. Historiquement, la quasi totalité des modems utilise un jeu de commandes dérivé de celui utilisé par le modem Hayes Smartmodem TM300 (un vénérable ancêtre travaillant à 300bits/s). Toute ces commandes sont précédées des caractères « AT ». Parmi les commandes usuelles, on peut noter :

A :mise en réponse automatique
Dxxxxxxx :numérote xxxxxxx
P :numérotation par impulsioins (Pulses)
T :numérotation par fréquences vocales (Tone)
, :pause pendant la numérotation
H :racrocher (Hangup)
I :informations sur le modem
Sn ? :affiche la valeur du registre n
Sn=x :modifie le registre n

Quand le modem est passé en mode communication, trois « plus » espacés d'une seconde permettent de repasser en mode commande.

Chapitre 5 Architecture de réseau : le modèle OSI

Historique

En 1983, l'organisation de normalisation des réseaux : l'OSI (Open System Interconection) propose un modèle de référence en 7 couches. Ce modèle analyse les fonctionnalités d'un réseau et les classe en couches fonctionnelles. Ce modèle servira de référence incontournable par la suite. Cependant, après le succès de la normalisation des couches basses, la normalisation des couches applicatives va se heurter à des difficultés qui entraîneront son abandon au début des années 90.

Les documents décrivant les normes OSI sont disponibles (et fort chers d'ailleurs) auprès de l'AFNOR en France.

Le modèle en couches

Le modèle OSI analyse le fonctionnement d'un réseau en 7 couches :

Application
Présentation
Session
Transport
Réseau
Liaison
Physique

Figure 5 : Les 7 couches du modèle OSI

Chaque couche fournit des services pour les couches supérieures en s'appuyant sur les services offerts par les couches inférieures.

Chaque couche fournit des services de :

- connexion/deconnexion
- transfert de données
- commandes
- contrôle de flux
- intégrité
- multiplexage de services

Chaque couche transmet les données de la couche supérieure en les enrobant des données nécessaire à ses propres fonctionnalité par un mécanisme de « poupées russes ». Le segment de données concerné par une couche est appelé un PDU : Protocol Data Unit.

Les fonctionnalités des différentes couches sont les suivantes/

Couche physique :

Elle est chargée de la transmission des données sur un support physique. Les normes de cette couche définissent la manière de transmettre une suite de bits sur un support de données. On y trouve les spécifications des liaison séries, de l'accès au câble Ethernet, …

Couche liaison

Cette couche définit la manière dont des segments de données sont échangés entre deux équipements réseau adjacents. A ce niveau, les segments échangés sont appelés « trames ». On y trouve des protocoles tels que LAPB, la trame Ethernet, …

Couche réseau

Cette couche est chargée de l'acheminement des segments de données (paquet) entre les deux équipements source et destination. Un paquet va ainsi voyager à travers le réseau en empruntant entre chaque équipement des trames différentes.

Deux des problème résolus à ce niveau sont :

- l'adressage : l'assignation d'une adresse unique à chaque machine.
- le routage : la structuration de ces adresses de manière à trouver une voie d'acheminement.

Couche transport

A ce niveau, deux problèmes sont traités :

- le multiplexage des flux de données arrivant sur une machine (la couche réseau permet d'arriver jusqu'à une machine, trouver l'application à qui remettre les données à l'intérieur de la machine est du ressort de la couche transport)
- l'établissement d'un flot de données fiable

Couche session

A partir de cette couche, on entre dans le domaine des application : la couche transport a sélectionné une application chargée de traiter les données.

Les service offerts par cette couche permettent de segmenter un échange en unités avec des points de reprise.

Cette couche était surtout prévue pour intégrer la notion de transaction dans les applications réseau.

Couche présentation

Cette couche était initialement prévue pour effectuer les traduction entre les différents codages de données. Il lui a été adjoint :

- la représentation de données complexe avec les protocoles ASN1 (Abstract Syntax Number 1) et son protocole de transfert : BER (Binary Encoding Rules).
- la compression des données.
- le cryptage des données.

Couche application

A ce niveau, on doit trouver les applications « utilisables » du réseau :

- FTAM : File Transfer, Access and Management
- VT : Virtual Terminal
- X400 : le courrier électronique
- X500 : le service d'annuaire électronique.

Exemples historiques

X25

X25 comprend la normalisation des trois premières couches du modèle. Il utilise :

- une liaison série de type V24 ou V11 pour l'accès à la couche physique
- LAPB pour le transfert des données entre équipements (couche 2)
- un mode paquet avec un adressage permettant de joindre un correspondant du réseau et d'établir avec lui un circuit virtuel de données réalisant les fonctions de la couche 3.

X25 fut adopté par de nombreux pays qui établirent des réseaux public utilisant ce protocole tel Transpac en France. Ces différents réseaux X25 nationaux furent de plus interconnectés avec un système d'adressage (X121) universel.

La plupart des constructeurs d'ordinateurs inclurent dans leur offre une interface X25 de sorte qu'il fut pour la première fois possible de faire communiquer des ordinateurs de fabricants différents en utilisant un réseau et un protocole public.

En ce sens, X25 est un succès et est encore largement employé dans le domaine des terminaux bancaires entre autres.

X400

L'étape suivante fut la normalisation de la messagerie électronique avec le protocole X400.

Basé sur une solide analyse des fonctionnalités d'un service de courrier électronique, la normalisation fut malheureusement trop ambitieuse en voulant régler tous les problèmes simultanément. Il en résultat un protocole non terminé et trop difficile à implémenter.

Des services publics tels que Atlas 400 se mirent en place mais furent balayés par l'émergence des protocoles de l'Internet, simples et efficaces.

X500

Parallèlement au service de messageries, l'OSI essaya de concevoir un système d'annuaire électronique très (trop) ambitieux. Affligé de la lourdeur des 7 couches bien trop lourdes à mettre en œuvre et d'objectifs bien trop complexe, cette norme n'eut jamais de réalisations viable.

Cependant, un sous-ensemble de X500, aux fonctionnalités moins ambitieuses et utilisant un simple transport TCP/IP au lieu de la machinerie OSI se développa au début des années 1990 : LDAP (Lighweight Directory Access Protocol). Il sert maintenant de protocole pour la plupart des services d'annuaire de l'Internet. LDAP est décrit dans le RFC2251

L'echec

Les autre protocoles du modèle OSI, tels que FTAM ou VT, ne furent pas utilisés. La situation devint intenable quand il s'agit de normaliser l'administration des réseaux avec les protocoles CMIP/CMIS (Common Management Information Protocol/ Common Management Information Services). Une fois de plus, le commité OSI réalisa une remarquable analyse du problème avec un modèle qui sert toujours de référence. Malheureusement, la normalisation s'enlisa dans une course à la complexité, chaque norme soulevant de nouveaux problème censés être réglés par une nouvelle norme.

Au début des années 90, devant le succès grandissant de l'Internet, l'OSI arrêta ses développements.

Ces épisodes eurent de nombreuses conséquences : devant l'obligation de faire référence qu modèle OSI pour les marchés officiels, tous les constructeurs ajoutèrent à leur offre une interface conforme au modèle OSI. Ils utilisèrent des voies différentes : IBM a conservé son réseau SNA en l'habillant d'une interface conforme au modèle. Dec décida de jouer le jeu et se lança dans une implémentation complète du modèle en remplacement de son protocole Decnet. Le résultat, Decnet/OSI fut une échec…

Chapitre 6 Architecture de réseau : TCP/IP

Historique

L'origine d'Internet remonte à une étude commandée par le département de la défense américain (DoD) sur un réseau à commutation de paquet vers le milieu des années 70. Cette étude conduit à la réalisation du réseau Arpanet, du nom de l'Advanced Research Projects Agency, ARPA.

Une des caractéristiques demandées était que le réseau soit sans hiérarchie (de manière à pouvoir résister à une attaque thermonucléaire !). Cette contrainte conduisit à un modèle de réseau très simple constitué d'une interconnexion de réseaux. Ce modèle présente la qualité fondamentale de pouvoir croître dans des proportions importantes sans rencontrer de problème d'échelle : c'est la clé du succès de l'Internet qui dans les années 90 va accueillir des dizaines de millions d'ordinateurs personnels.

Les protocoles développés pour cette étude vont être sélectionnés pour constituer ce que l'on appelle les protocoles TCP/IP (mais qui vont bien au-delà de ces deux protocoles).

Les choix définitifs vont être faits entre 1977 et 1979 concernant les fonctions de base du protocole. Ce sont eux qui servent encore actuellement.

Les rfcs

Les documents décrivant les protocoles de l'Internet sont appelés RFCs : Request For Comment. Ce sont des documents portant un numéro, depuis le RFC1. Il y en a actuellement environ 3000. Au départ, un RFC est juste une proposition pour une nouvelle norme, ou l'utilisation d'une nouvelle norme. Un RFC, une fois publié, n'est jamais modifié. Il sert de référence pour la discussion sur le sujet qu'il traite. Un RFC n'est pas soumis à l'approbation d'un comité de lecture, son contenu n'engage que son auteur et le seul examen qu'il subit avant publication concerne la forme et non le contenu.

Après discussion, un RFC peut passer plusieurs phases avant de devenir éventuellement un standard. L'état des différents stades de normalisation est publié lui-même sous forme de RFC. Le RFC2400 donne un état récent (1998) des différents standards reconnus.

Les RFC sont des documents publics dont l'accès est gratuit. Ils sont disponibles sur de nombreux serveurs « ftp anonyme », dont en France celui de l'INRIA : ftp.inria.fr.

Le modèle TCP/IP

L'architecture TCP/IP repose comme pour le modèle OSI sur un schéma en couches, heureusement simplifié à 5 composantes :

- Une couche physique identique dans ses fonctionnalités à la couche physique du modèle OSI.
- Une couche liaison similaire aussi à celle du modèle OSI
- Une couche réseau très simple : IP, Internet Protocol, chargée d'acheminer des segments de données appelés paquet entre deux machines du réseau.
- Deux couches de transport :
    - UDP : User Datagram Protocol, chargé d'acheminer à l'aide d'IP des segments de données entre deux applications avec le minimum de contrôle.
    - TCP : Transport Control Protocol, un protocol complexe fournissant aux applications un flux de données bidirectionnel fiable.
- Une couche application où se placent les divers protocoles permettant l'utilisation du réseau : telnet, ftp, http, …

Parallèlement à cette structure, on trouve, au-dessus de IP, un protocole servant à acheminer des messages de service pour gérer le fonctionnement de IP : ICMP, Internet Control Messages Protocol.

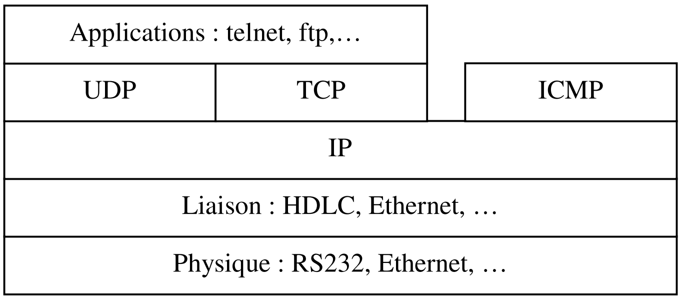

Figure 6 : Architecture TCP/IP

Acheminement des données : IP

La couche IP : Internet Protocol constitue la brique élémentaire sur laquelle est construit tous le réseau. Sa fonction est très simple : acheminer autant que possible des segments de données appelés paquet entre deux machines du réseau. IP ne garantit rien sur la réussite de cet acheminement, des paquets peuvent être :

- perdus
- altérés
- dupliqués
- ré-ordonnés

Il est à la charge des couches supérieures de se prémunir contre ces altérations. L'extrême simplicité de ce protocole est sans doute la clé de son succès : il fait très bien ce pour quoi il a été conçu, laissant aux autres couches le soin d'implémenter des mesures plus robustes si cela est nécessaire

Les moyens qui sont mis en œuvre pour acheminer ces paquet sont l'objet du routage et seront examinés par la suite.

IP est défini dans le RCF791.

Pour acheminer les paquets, IP enrobe les données que lui ont confié les couches supérieures avec les informations dont il a besoin :

- la version de IP utilisée : 4 actuellement.
- le type de protocole transporté (par exemple tcp).
- l'adresse de la source du paquet et celle de son destinataire.
- un numéro identifiant le paquet
- une « durée de vie » (TTL : Time To Live) décrémentée à la traversée de chaque équipement réseau de manière à détecter les paquets qui « bouclent » dans le réseau.
- une somme de contrôle permettant de vérifier l'intégrité de l'en-tête de données (mais non des données elle-même : cela est du ressort des couches supérieures).
- un champ Type Of Service (TOS) indiquant au équipements traversé ce que l'on souhaite privilégier :
    - le temps d'acheminement
    - le débit
    - la fiabilité du transfert
- des champs permettant de transporter des options suplémentaires
- des champs permettant de fragmenter les paquets IP s'il doivent transiter par un support pour lequel il sont trop longs (par exemple un paquet de 2000 octets traversant un réseau Ethernet).

<table>
<tr><td>V</td><td>hl</td><td>TOS</td><td colspan="2">longueur totale</td></tr>
<tr><td colspan="3">id.</td><td>F</td><td>ind. fragment</td></tr>
<tr><td colspan="2">TTL</td><td>prot.</td><td colspan="2">CRC</td></tr>
<tr><td colspan="5">adresse source</td></tr>
<tr><td colspan="5">adresse destination</td></tr>
<tr><td colspan="5">options</td></tr>
<tr><td colspan="5">données</td></tr>
</table>

Figure 7 : Format d'un paquet IP

La notion de port et de socket

Le protocole IP permet d'acheminer des paquets de données entre deux machines. A l'intérieur de chacune de ces machines, il faut de plus identifier les applications. Cela est fait au moyen de la notion de port : il s'agit d'un nombre (sur 16 bits, de 0 à 65535) identifiant une application. Certains ports sont particuliers :

- réservés à une applications (« well known ports »), par exemple 23 pour telnet, 25 pour le courrier électronique, 6000 pour X-Windows.
- privilégiés, c'est à dire d'un numéro inférieur à 1024 : seul le « super utilisateur » (root) peut les utiliser.

L'ensemble des adresses des deux machines et des ports utilisés sur chaque machine identifie de manière unique deux applications en cours de dialogue sur le réseau et est appelé « socket » :

AdresseA :PortA<->AdresseB :PortB

La remise des paquets au port destination est du ressort des couches supérieures (udp et tcp) et se nomme le multiplexage.

Sur une machine Unix, les différents ports sont décrits dans le fichier /etc/services. Les différentes applications sont prises en charge par des « démons » qui sont décrits dans le fichier /etc/inetd.conf.

La commande netstat permet de consulter la liste des « sockets » établis.

Le RFC1700 contient (entre autres) les numéros des ports réservés.

Transfert de datagrammes : udp

Dans la famille TCP/IP, le protocole de transport le plus simple est UDP : User Datagram Protocol. Sa fonction se limite au multiplexage des ports et à un contrôle (facultatif) de l'intégrité des données qu'il transporte.

L'en-tête que UDP rajoute au données qu'il transporte contient dont uniquement les numéros des ports sources et destination, la longueur des données transportées et une éventuelle somme de contrôle.

Les données UDP sont transportées dans des paquet IP dont le numéro de protocole est 17.

En résumé, quand une application remet un segment de données à UDP, celui-ci ajoute son en-tête, passe l'ensemble à IP qui ajoute lui aussi son en-tête et transmet le paquet ainsi constitué à la couche liaison. Le paquet va ainsi traverser le réseau en étant transportés par différentes couches liaison entre chaque équipement pour aboutir finalement à la machine destinataire ou la couche liaison va le remettre à IP, qui va le passer à UDP, qui en fonction du port va le remettre à une application.

UDP est normalisé par le RFC 768.

UDP est utilisé par des applications qui nécessitent simplement l 'échange de messages ou par des applications qui doivent être mise en œuvre avec le minimum de moyen, tel que bootp et tftp (qui doivent pouvoir fonctionner depuis un programme résidant dans la ROM d'une machine).

Transfert de flux : tcp

L'autre protocole de transport est beaucoup plus puissant et complexe : TCP, Transmission Control Protocol, propose un flux de données bidirectionnel et fiable.

La tache de TCP consiste donc à découper les données que lui transmet l'application en segment, à les faire transiter à travers le réseau et à les rassembler pour les remettre à l'application destinataire. Ce processus peut avoir lieu simultanément dans les deux directions.

Pour cela, TCP doit :

- établir une connexion entre les deux application en évitant les « mascarades »
- découper les données en segments et les ré-assembler à l'arrivée.
- vérifier l'intégrité des segment et éventuellement demander le renvoi de segments corrompus
- assurer la retransmission des segments perdus
- éliminer les segments éventuellement dupliqués
- assurer des mécanismes de contrôle de flux pour éviter l'engorgement à différents niveaux
- utiliser au mieux la bande passante disponible dans le réseau

Pour repérer les segments, TCP utilise une numérotation des octets du flux à transmettre. Chaque segment transporte avec lui ce numéro d'ordre appelé numéro de séquence.

TCP s'assure de la remise des données par un mécanisme d'accusé de réception.

Si un accusé de réception n'arrive pas, il déclenche le renvoi des segments manquants au bout d'un intervalle de temps appelé RTO : Retransmission Time Out. Ce RTO est estimé par des algorithmes complexes à partir des mesures du temps d'aller-retour à travers le réseau (RTT : Round TripTime).

L'initialisation de la connexion est faite par un mécanisme appelé le « three way handshake » qui consiste en l'échange de trois paquets avec des numéros de séquence pris au hasard et qui serviront de séquences initiales et de leurs accusés de réception. Ce mécanisme permet en principe d'éviter qu'une machine triche en prenant l'adresse d'une autre : la « mascarade ».

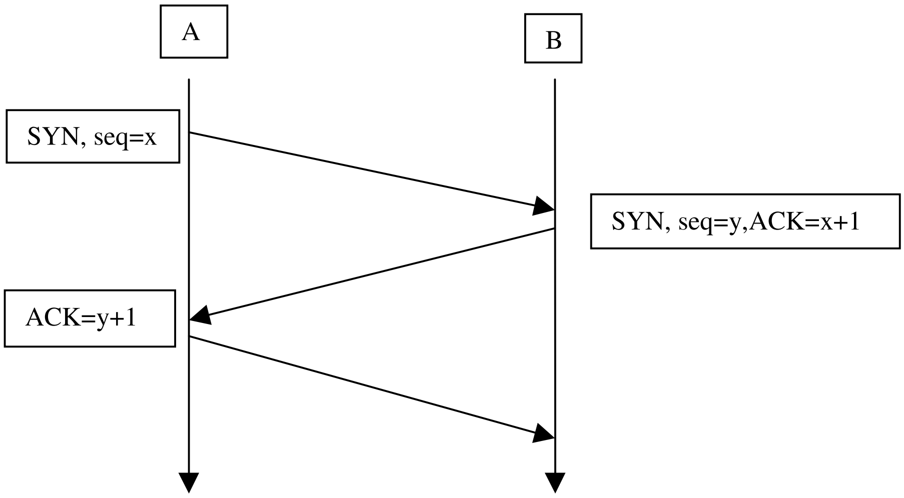

Figure 8 - Le « three way handshake »

Pour augmenter l'efficacité du transport, TCP anticipe l'envoi des données avant accusé de réception des précédentes. La quantité anticipée s'appelle la fenêtre et est réglée par le récepteur.

L'en-tête du paquet TCP est assez complexe et offre de plus la possibilité de rajouter des options.

Les segments TCP sont acheminés dans des paquets IP avec le numéro de protocole 6.

TCP est normalisé par le RCF793.

TCP est utilisé par de nombreuses applications ayant besoin d'un flot de données fiable : telnet, ftp, http, le mail, etc…

Supervision du réseau : icmp

Dès sa sonception, IP a été doté d'un protocole de supervision permettant de faire transiter sur le réseau des informations sur son fonctionnement. Ce protocole est nommé ICMP : Internet Control Message Protocole.

Les messages ICMP circulent dans des paquets IP avec comme numéro de protocole 1.

De nombreux types de messages sont prévus, parmi lesquels :

- La demande d'écho et sa réponse
- L'indication d'une destination inaccessible
- L'indication qu'un paquet a atteint sa durée de vie (c'est à dire qu'il a traversé trop de machine et est probablement en train de boucler)

ICMP est décrit dans le RFC792

ICMP permet le fonctionnement de plusieurs utilitaires :

Ping

Ping est un programme permettant de tester si une machine est accessible ainsi que de la qualité du réseau pour atteindre cette machine.

Il repose simplement sur l'envoi à intervalle régulier de messages ICMP de demande d'écho et l'enregistrement des réponses avec leur heure d'arrivée.

Avec ces données, il fournit alors :

- Le pourcentage de paquets perdus
- Le temps minimal/moyen/maximal mis par les paquets pour faire l'aller-retour.

La longueur de chaque paquet, ainsi que leur fréquence d'émission sont réglables, ce qui en fait un instrument très simple permettant de réaliser de nombreuses mesures sur un réseau.

Traceroute

Traceroute est un programme permettant de déterminer le chemin emprunté par les paquets IP pour atteindre leur destination.

Il repose sur l'émission successive de segments UDP avec des TTL croissants : 1, 2,3,…

Le premier segment voit son TTL tomber à 0 à la première machine traversée et cela déclenche l'émission d'un paquet ICMP : fin de durée de vie. Ce paquet indique, par sa source, la première machine traversée. Le second paquet franchit la première machine, mais à la seconde déclenche à son tour l'envoi d'un message ICMP et ainsi de suite.

On obtient ainsi la liste des machines traversées jusqu'à la machine cible.

Ip sur Ethernet

Les paquets IP doivent pour traverser le réseau être encapsulés entre chaque équipement dans une trame adaptée. Le RFC 894 décrit ce processus pour Ethernet:

La paquet IP est simplement placé dans la partie données du datagramme Ethernet et le type de protocole est mis à 0x0800.

Pour les datagrammes 802.3, la procédure est un peu plus complexe et décrite dans le RFC1042.

Il reste cependant un problème : pour acheminer le datagramme Ethernet vers la bonne machine, il faut connaître son adresse Ethernet, or à priori, seule l'adresse IP est connue. Le protocole ARP (Address Resolution Protocol) permet de trouver l'adresse Ethernet correspondant à une adresse IP sur un réseau local.

ARP

Le processus ARP est décrit dans le RFC826. Il consiste simplement à émettre un « broadcast » sur le réseau avec un datagramme Ethernet de protocole 0x0806 contenant l'adresse IP de la machine cherchée. Cette machine, si elle existe, reçoit le datagramme (les « broadcast » sont lus par toutes les machines du réseau) et répond en indiquant son adresse Ethernet.

La correspondance entre les adresses IP et Ethernet, une fois obtenue par ARP, est conservée dans une table.Sur une machine Unix, cette table peut être consultée ou modifiée par la commande arp.

Ip sur liaison série

De même que pour Ethernet,, il faut définir des règles pour transmettre des datagrammes IP sur une liaison série, telle qu'une connexion chez un Fournisseur d'Accès Internet (FAI). Le premier protocole utilisé dans ce but était SLIP (Serial Line IP) décrit dans le RFC 1055. Il n'est pratiquement plus utilisé et remplacé par PPP (Point to Point Protocol) doté de possibilités beaucoup plus étendues.

ppp

PPP est décrit dans le RFC1661.

Il se décompose en plusieurs fonctions :

- l'identification de l'appelant qui peur être fait soit par le Password Authentication Protocol (PAP), soit par le Challenge Handshake Protocol (CHAP) décrits dans le RFC1334.
- le Link Control Protocol (LCP) chargé d'établir et de gérer la liaison. Sa fonction est similaire à celle de HDLC.
- le Network Control Protocol (NCP) décrit comment transmettre les protocoles de réseau sur la liaison. Divers protocoles sont prévus : IP, Decnet, AppleTalk,… chacun faisant l'objet d'un autre RFC. Pour IP, il s'agit du RFC1332.

D'autre part, sur les liaison utilisant PPP, les en-tête ip et tcp sont très souvent les mêmes, à quelques détails près. Pour économiser la précieuse bande passante, les 40 octets des en-têtes ip et tcp sont compressés à l'aide d'un algorithme décrit dans le RFC1144 (algorithme de Van Jacobson) qui les réduit entre 3 ou 5 octets dans la plupart des cas.

Chapitre 7 Le routage

Position du problème

L'ensemble de TCP/IP repose sur l'acheminement des paquets IP à travers le réseau. Cet acheminement s'appelle le routage.

Chaque machine se voit attribuer une ou plusieurs adresse prise dans un ensemble d'environ 4 milliards. Cet ensemble est structuré en groupements appelés réseaux. Ces ensembles peuvent être de taille variable, de quelques machines à 16 millions.

Deux machines appartenant au même réseau peuvent dialoguer directement en échangeant leurs paquets dans des trames du support qui les relie directement, par exemple Ethernet.

Deux machines appartenant à des réseaux différents doivent passer par un routeur, c'est à dire une machine ayant des adresses dans plusieurs réseaux.

Par interconnexion des différents réseaux, les routeurs acheminent de proche en proche les paquets IP jusqu'à leur remise à la machine destinataire (d'où le nom d'Inter Net(work) donné à cette architecture de réseaux).

Le routage consiste ainsi à choisir à chaque étape l'adresse de la prochaine machine pour atteindre l'adresse cible.

La correspondance entre un réseau et l'adresse de la prochaine machine (routeur) pour l'atteindre s'appelle une route et l'ensemble des routes connues d'une machine s'appelle la table de routage.

Le routage dans une machine se fait donc pour chaque paquet IP en consultant la table de routage pour trouver à quel machine le transmettre.

La constitution de cette table de routage peut être manuelle (c'est la cas de la plupart des machines individuelles) ou automatique. La construction automatique de la table de routage se fait par dialogue entre les routeurs. Ce dialogue ce fait à l'aide d'un protocole de routage.

Sur une machine Unix, la commande route permet de gérer la table de routage :

- route add pour ajouter une route
- route del pour en supprimer une

Pour afficher la table de routage, la commande est netstat –r.

Enfin, il existe une notion de route par défaut : l'adresse à laquelle on envoie touts les paquets pour lesquel on n'a pas de route explicite.

La configuration d'une machine individuelle se limite donc en général à une route pour atteindre le réseau auquel elle appartient (dirigée vers son unique interface, ppp ou Ethernet) et une route par défaut dirigée sur le routeur qui gère son réseau.

Structure des adresses IP

Les adresses IP consistent en un nombre de 4 octets, ce qui offre théoriquement la possibilité de 4 milliard d'adresses. On les note conventionnellement sous la forme de quatre nombre décimaux séparés par des points, par exemple «134.159.69.191 » appelée notation décimale pointée.

Quelques adresses ont un rôle spécial :

- 255.255.255.255 s'adresse à toutes les machines du réseau (broadcast).
- 127.x.x.x désigne la machine elle-même, sans passer par une interface physique (« local loopback ») .

Sur une machine Unix, la commande permettant de configurer ou de vérifier l'adresse IP est ifconfig.

Ces adresses sont regroupées en réseaux (ensembles d'adresses successives) de la manière suivante :

- les réseaux commençant par 0 à 127 constituent 128 réseaux, dits de classe A, dans lesquels les 3 derniers octets peuvent varier, ce qui donne environ 16 millions d'adresses.
- les réseaux commençant par 128.0 à 191.255 constituent environ 16000 réseaux, dits de classe B, dans lesquels les deux dernier octets peuvent varier, ce qui donne environ 65000 adresses.
- les réseaux commençant par 192.0.0.0 à 223.255.255 constituent environ 2 millions de réseaux, dits de classe C, dans lesquels le dernier octet peut varier, ce qui donne 256 adresses.
- les réseaux restants sont classés en :
- des réseaux dits de classe D, réservés aux multicast : un multicast désigne un ensemble de machine au lieu d'une machine individuelle et servent par exemple dans les applications de vidéoconférences .
- des réseaux dits de classe E réservés pour de futures applications.

Une procédure appelée sous-réseau permet de découper un réseau de classe A, B ou C en plus petits réseaux. Par exemple, un réseau de classe B peut être découpé en 32 sous-réseaux de 2048 adresses chacun. Ce découpage est fait à l'aide d'un masque de sous-réseau : le masque 255.255.248.0 réalise le découpage précédent car :

248 = « 11111000 » en binaire

ce qui signifie que dans un sous réseau, les 5 premiers bits sont fixes (il y à donc 32 sous-réseaux possibles) et que les 3 suivant (plus les 8 du dernier octet) varient, ce qui donne 8*256=2048 adresses possibles.

Conventionnellement, dans un réseau ou un sous-réseau,, l'adresse la plus basse (tous les bits qui peuvent varier mis à 0) désigne le réseau lui-même, et l'adresse la plus haute (tous les bits qui peuvent varier mis à 1) est l'adresse de broadcast. Par exemple, si on prend le réseau de classe B 134.158.0.0 et qu'on le découpe en 32 sous-réseaux à l'aide du masque de sous-réseau 255.255.248.0, alors 134.158.136.0 est l'un de ces sous-réseaux (136= « 10001000 » en binaire) et 134.158.143.255 est l'adresse de broadcast sur ce sous-réseau (143= « 10001111 » en binaire.

Architecture interne/externe

La gestion des tables de routage de l'internet est évidemment un problème très complexe, des centaines de milliers de réseaux sont maintenant connectés et il est hors de question que tous les routeurs de la planète tiennent à jours un table de routage complète : l'Internet ne servirait probablement plus qu'à acheminer des informations de routage !

L'attitude choisie à été de « diviser pour régner » : le problème trop complexe est divisé en une série de problème plus simples à régler. Pour cela, on divise l'ensemble des réseaux en sous ensembles nommés systèmes autonomes (AS : Autonomous System). Un système autonome est un ensemble de réseaux administrés par une autorité unique. A l'intérieur du système autonome, l'administrateur choisi une politique de routage qu'il administre à l'aide d'un protocole de routage interne placé sous sa seule responsabilité.

Les systèmes autonomes sont désignés par un nombre de deux octets (il y en a donc 65536), par exemple, l'In2p3 constitue le système autonome numéro 1789.

Vis-à-vis de l'extérieur, les différents systèmes autonomes échangent alors leurs informations de routage à l'aide d'un protocole de routage externe

Cette division du problème permet de ramener le problème du routage de l'Internet au routage entre les quelques milliers de système autonomes qui à leur tour gèrent leurs problèmes de manière interne.

L'explosion de Internet dans les années 90

Au début des années 1990, l'Internet connaît une phase de croissance exponentielle.

Epuisement des adresses

Le nombre de machines connectées augmente de façon exponentielle. L'espace d'adresses disponibles (environ un milliard) permet théoriquement de faire face à la demande, mais la répartition en classe A,B et C se révèle inadaptée : les réseaux de classe C sont en général trop exigus, ce qui entraîne une demande excessive de réseaux de classe B qui s'épuisent rapidement. On arrive à une situation où l'on dispose d'une pléthore de réseaux de classe C avec une pénurie de réseaux de classe B.

Explosion des tables de routages

Parallèlement, on assiste à une croissance explosive des tables de routages : les numéros de réseaux sont attribués à la demande, un réseau situé en Asie est attribué après un réseau localisé en Europe ou en Amérique. Les tables de routage deviennent très morcelées, avec des dizaines de milliers de réseaux attribués sans aucun ordre.

Les solutions

L'IETF adopte alors une série de mesures pour faire face à cette situation :

- Un plan à long terme d'une définition d'une nouvelle version du protocole IP. Après de longues études, un successeur de IP Version 4 est adopté, ce sera IP Version 6, aussi connu sous le nom de IPNG : IP Next Generation.. Cette nouvelle version du protocole IP adopte un nouveau schéma d'adressage avec des adresses sur 16 octets (environ $2*10^{38}$ d'adresses !) et une simplification de l'en-tête IP, reportant certaines informations dans les champs d'option. Le premier RFC spécifiant ce protocole est le RCF2460.

- Un plan à court terme visant à apporter une solution transitoire. Le schéma adopté consiste à revoir l'attribution des adresses de manière géographique en distribuant des groupes de réseaux de classe C et à une modification des protocoles de routages permettant d'acheminer non plus des routes isolées, mais des groupements de routes. Une mesure complémentaire consiste à définir une méthode permettant d'utiliser des adresse locales qui peuvent être dupliquées dans le réseau.

Redéploiement des adresses

La première mesure vise à résoudre l'épuisement des adresses de classe B : les réseaux attribués seront désormais systématiquement des réseaux de classe C. Mais ils seront attribués par groupes consécutifs et il seront distribués de maniére géographique : les opérateurs européens recevront un groupe de réseaux consécutifs qu'il se répartiront en groupes d'adresses contiguës, etc…

Cette première mesure permet d'éviter l'épuisement des réseaux de classe B en puisant dans le parc pléthorique des réseaux de classe C.

Elle permet d'autre part de préparer la mesure suivante.

CIDR

Les numéros de réseaux étant maintenant attribués de manière structuré, il devient possible de simplifier le problème du routage. Cela est réalisé avec CIDR : Classless Inter Domain Routing. L'idée est d'une simplicité extrême : au lieu de transmettre un nombre élevé de route, les routeurs transmettront des groupement de routes pour des réseaux consécutif par un mécanisme similaire au sous-réseau : le sur-réseau ou agglutination de routes. Cette mesure est formalisée dans le RFC1519.

Panorama des protocoles

Une analyse des protocoles de routage est bien au-delà du propos de ce fascicule. On se contentera donc d'une description « zoologique ».

Classification des protocoles

En dehors de leur classification en protocoles de routage interne ou externe, on distingue deux types de protocoles, en fonction de leur stratégie pour constituer leur table de routages : les protocoles « distant-vector » et les protocoles « link state ».

distance vector

Dans les protocoles « distant vector » ou encore « Bellman-Ford », du nom des auteurs, chaque routeur diffuse périodiquement à tous les routeurs avec lesquels il est en contact la liste des réseaux qu'il connaît, ainsi que le routeur permettant de les atteindre et le nombre de routeurs à traverser.

L'avantage évident est la simplicité.

Le prix à payer est lourd :

- chaque routeur diffuse sa vision du réseau et le routage global est souvent constitué d'une série de vues inconsistantes. En particulier, tout changement de topologie du réseau s'accompagne de phases d'instabilité pendant lesquelles des routages aberrants apparaissent.
- la taille des informations échangées dépend directement de la taille du réseau et devient rapidement prohibitive.

Malgré ces inconvénients, ce type de protocole, accompagné d'une série de mesures correctives (« heuristiques ») est très souvent utilisé pour le routage interne.

link state

Dans cette famille de protocoles : « link state » ou encore SPF : « Short Path First », chaque routeur surveille l'état de ses liaisons avec les routeurs adjacents et diffuse cet état. Chaque routeur constitue alors une carte globale du réseau à l'aide de ces informations. Il est alors en mesure de déterminer le chemin le plus court pour atteindre l'ensemble des réseaux à l'aide d'un algorithme connu sous le nom de son auteur : Dijskra.

Deux avantages à cette stratégie :

- tous les routeurs établissent leur table de routage à partir des même informations. Ils ont donc une vue cohérente et réaliste du réseau.
- la taille des message échangés est indépendante de la taille du réseau.

Il y a toutefois un inconvénient de taille : le calcul du plus court chemin est connu comme un problème dont la résolution demande des ressources énormes en puissance de calcul.

Ces protocoles ont donc du attendre l 'apparition des processeurs puissants à bas prix pour devenir utilisables.

Protocoles externes

EGP

EGP ou External Gateway Protocol a longtemps été utilisé sur l'épine dorsale de l'Internet.

Il souffre toutefois de nombreuses limitations : il ne sait gérer qu'une topologie de système autonomes en arbre et il ne sait pas gérer plusieurs voies pour atteindre un système autonome (il ne sait en particuliers pas gérer des voies de secours).

Pour communiquer, chaque systéme autonome dispose d'un routeur désigné qui dialogue avec un routeur correspondant du système autonome partenaire. Ces deux routeurs échangent leur informations à l'aide de paquets IP de numéro de protocole 8 (donc ni udp, ni tcp).

Il est décrit dans le RFC904.

BGP

BGP, Border Gateway Protocol est le protocole de routage externe moderne de l'Internet. De par son fonctionnement, il n'est pas vraiment classable comme « link state » ou « distance vector » : il s'agit plus d'un moyen de diffusion de données utilisées par ailleurs que d'une méthode de calcul des tables de routages.

Le protocole décrit le dialogue entre des routeurs désignés de système autonomes coopératifs. Ces routeurs disposent d'un protocole d'authentification permettant d'éviter les intrus.

Le dialogue entre routeurs se fait par une connexion TCP sur le port 179 et le fait de disposer d'une connexion fiable et permanente permet des mises à jour incrémentales d'où un trafic limité au minimum.

Ce protocole supporte CIDR dans la version actuellement déployée : la version 4, décrite dans le RFC1771.

Protocoles Internes

RIP

Il s'agit d'un des premiers protocoles de routage interne déployé sur l'Internet (le premier fut HELLO, utilisé sur l'épine dorsale de la NSFNET, qui n'eut pas de successeur en tant que protocole de routage, mais qui a servi de base au protocole de diffusion du temps : NTP). Il fut conçu par Xerox, puis implémenté dans Unix BSD sous le nom de routed. Bien qu'il souffre à priori des défauts inhérents au protocoles « distance vector », les programmeurs qui réalisèrent routed utilisèrent tout leur savoir faire et inclurent une série d'algorithme qui le rendaient efficace.

Ce protocole servit de modèle pour les protocoles de routage de nombreux réseaux : AppleTalk, Novel,3Com,…

Les messages échangés sont des paquets UDP diffusés sur le port 520.

Ce protocole est décrit dans le RFC1058 pour sa remière version, puis 1388 pour sa seconde version.

IGRP

IGRP (InterGateway Routing Protocol) n'est pas vraiment un protocole de l'Internet, mais le protocole utilisé par les routeurs du constructeur Cisco. L'importance de ce constructeur dans l'Internet lui vaut de figurer dans les protocoles décrits.

A la base, il est dérivé de RIP, mais avec certaines extensions aboutissant à un protocole très efficace :

- une métrique (la « distance » à laquelle un routeur voit un réseau) tenant compte du délai de propagation, de la bande passante, de la fiabilité, de la charge et du nombre de routeurs traversés.
- la possibilité de répartir le traffic vers une destination sur plusieurs chemins en parallèle.
- la possibilité de transmettre des informations de routage sur les sous-réseaux.

Les messages sont diffusés dans des paquets IP avec un numéro de protocole de 63.

Cisco utilise maintenant une extension de ce protocole : EIGRP : Extended IGRP.

OSPF

L'Internet utilise un protocole de type link-state : OSPF : « Open Shortest Path First ».

Le système autonome est divisé en « area » à l'intérieur desquelles le routage est fait par des « intra-area routers ».

Les différentes area communiquent par des « area border routers » qui constituent le « backbone » du système autonome.

Il est possible de répartir le traffic sur des chemins de coût équivalent, le chemin peut être choisi en fonction d'un type de service et le routage des sous-réseaux est prévu.

Le protocole prévoit un système d'authentification évitant les intrus et les messages sont échangés dans des paquets IP de protocole 89.

Ce protocole est décrit dans le RFC2328.

Utilisation d'adresses locales

Un des problème rencontré dans l'expansion de l'Internet est la pénurie d'adresses. Comme on l'a vu, le redéploiement a permis d'enrayer ce phénomène. Cependant, une meilleure utilisation des adresses disponibles est possible par l'utilisation d'adresses locales : dans de nombreux cas, par exemple une entreprise souhaitant donner un accès à l'Internet depuis ses postes de travail, l'attribution d'adresses « mondiales » à tous ses équipements n'est ni nécessaire, ni souhaitable. Le RFC 1597 prévoit dans ce but la réservation d'une série de réseaux (un de classe A, 16 de classe B et 256 de classe C) pour des usages privés. Ces réseaux sont réputés « non routables » (les routeurs doivent ignorer les informations les concernant) et ils peuvent ainsi être réutilisés à volonté.

Un nouveau protocole permet alors d'utiliser ces réseau de manière à constituer des réseaux privés dont les machines peuvent se connecter à l'Internet, mais qui restent invisibles de l'extérieur. En plus d'apporter une solution à la pénurie d'adresses, cette solution apporte de nombreux avantages quand à la sécurité de ce réseau privé.

NAT

NAT : Network Address Translator décrit la manière dont une machine d'un réseau (ce peut être le routeur) utilise une seule adresse connue du monde extérieur (« routée ») pour permettre à toutes les machines de son réseau interne, dotées d'adresses privées, d'établir des connexions sortantes sur l'Interne. La machine repère les appels TCP sortants, établit à son compte la connexion demandée (avec son adresse) et mémorise cette connexion. Elle traduit ensuite tous les paquets afférents à cette connexion dans les deux sens.

Des équipements spécifiques réalisent maintenant cette fonction, combinée aux tâches de routage en utilisant par exemple une connexion RNIS. Ils offrent de plus un accès entrant redirigeant les connexions arrivant sur les protocoles connus : Web, ftp et mail. Ils constituent une manière économique de connecter une petite unité au réseau.

NAT est décrit dans le RFC 1631.

Chapitre 8 Le service de nom

Les adresses IP numériques sur 4 octets sont pratiques en ce qui concerne le routage, mais présentent l'inconénient d'être fort peu utilisables par des êtres humains. Dès les débuts de l'Internet, la nécessité de désigner les machines par un nom est apparue. C'est l'objet du DNS : Domain Name System.

Historiquement, la correspondance entre nom de machine et adresse IP était stockée dans un fichier « host », LE fichier host maintenu sur une machine américaine et que les administrateurs transféraient à l'occasion sur leur machine (il existe toujours un fichier host sur les machines Unix).

Avec le développement de l'Internet, le besoin d'une solution plus globale est très vite apparu, mais l'ampleur de la tâche est immense : imaginer une sorte de base de données répartie sur l'ensemble de la planète ou il est possible d'ajouter des noms de machine depuis n'importe où tout en gardant l'ensemble à jour.

Le DNS apporte pourtant une solution particulièrement élégante : il a traversé toutes les évolutions de l'Internet depuis 1987 sans la moindre modification !

Le DNS est déccrit dans 2 RFC :

- RFC1034 : définit les principes
- RFC 1035 : décrit l'implémentation

Le modèle hiérarchique

Le principe de base pour l'organisation du DNS a été la délégation : l'espace de nommage (l'ensemble des noms de machines) est divisé en domaines principaux. Chacun de ces domaine est administré par une autorité de nommage qui divise à son tour cet espace en déléguant à d'autres organismes l'administration de ces sous-domaines, et ainsi de suite.

Par exemple, le domaine « fr » est administré par un organisme nommé l'AFNIC(Association Française pour le Nommage Internet en Coopération). Cet organisme (en réalité son prédécesseur à l'époque) a délégué au Centre de Calcul de l'In2p3 la gestion du domaine in2p3.fr.Celui-ci a à son tour délégué la gestion du domaine lal.in2p3.fr aux administrateurs réseau du Laboratoire de l'Accélérateur Linéaire d'Orsay.

Les noms de la hiérarchie au sommet sont :

- les 7 « grands » :
  - com : les organismes commerciaux
  - edu : les établissements d'enseignement
  - gov : le gouvernement américain
  - int : les organismes internationaux
  - mil : les organismes militaires américains
  - net : les organisations liées au fonctionnement du réseau
  - org : les organisations non commerciales
- les pays :
  - uk : le Royaume Unis
  - fr : la France
  - …
- le domaine arpa servant aux résolutions inverses (trouver le nom correspondant à une adresse).

Pour les domaines com, net et org, l'attribution des adresses et des noms est maintenant confiée à l'ICANN : Internet Corporation for Assigned Names and Numbers.

Il est à noter que, quand on est dans le même domaine, on peut ne donner que le nom de la machine. Par exemple, pour se connecter sur la machine ccali du Centre de Calcul dont l'adresse complète est ccali.in2p3.fr, il suffit de donner le nom ccali depuis la machine lyopc72.in2p3.fr. Par contre pour aller sur la machine cms du Cern, il faut donner en entier cms.cern.ch.

Fonctionnement

Les serveurs de nom sont organisé suivant une hiérarchie :

- au sommet, se trouvent les « root servers » qui maintiennent la liste des serveurs de tous les domaines principaux. Il sont actuellement au nombre de 13 et leur liste peut être obtenue sur la machine ftp.rs.internic.net dans le fichier root-servers.txt.
- chaque domaine maintient un serveur primaire détenant la liste de référence des noms qu'il administre.
- ces serveurs sont répliqués sur des serveurs secondaires qui récupèrent régulièrement une copie de la configuration du serveur primaire. En général, on place un serveur secondaire proche du serveur primaire (il sert de serveur de secours), un serveur secondaire à l'extérieur du domaine administré (il sert si une partie du domaine est isolée) et des serveurs secondaires aux endroits où des requêtes seront fréquentes.

Ces serveurs constituent une base de données répartie qui gère différents types d'enregistrements, parmi lesquels :

- SAO : Start Of Authority. C'est la définition d'un sous-domaine : le responsable, le serveur de nom primaire, etc…
- A : Address : Le nom associé à une adresse IP.
- NS : Name Server : Un serveur pour un domaine
- CNAME : Canonical NAME : La définition d'un alias.
- MX : Mail eXchanger : La machine recevant le courrier associé à une pseudo-adresse. Ils jouent un rôle considérable dans l'acheminement du courrier électronique en permettant (entre autres), de centraliser la réception des courriers électroniques sur des machines dédiées à cette tâche, et donc administrées en conséquent.
- PTR : PoinTeR : Utilisés dans les résolutions inverses.

Dialogue avec les serveurs

Le dialogue avec les serveurs de nom se fait par une connexion UDP ou TCP sur le port 53 :

- une machine désirant obtenir l'adresse associée à un nom (ou l'inverse) utilisera un échange UDP.
- un serveur secondaire désirant obtenir une copie de la configuration du primaire utilisera TCP (cela s'appelle un « zone transfert »)

Bien que d'autres types de fonctionnement soient prévus, la configuration utilisée dans la quasi totalité des réseaux est :

- chaque site maintient un serveur de nom, secondaire du réseau auquel il appartient et éventuellement des domaines avec lesquels il a un fort trafic. Ce serveur est en charge de toutes les résolutions de noms et d'adresses, il est en temps normal le seul qui dialogue avec les autres serveurs.
- les machines sont configurées pour transmettre toutes leurs requêtes au serveur désigné, ou a un éventuel serveur de secours.

Cette organisation hiérarchisée à le mérite de simplifier les rôles, avec une configuration très simple sur les machine et une seule machine par site demandant une configuration suivie.

Sur Unix, un utilitaire : nslookup, permet le dialogue avec les serveurs de nom.

Résolutions inverses

En général, le serveur de nom permet d'obtenir l'adresse IP correspondant à un nom, mais le problème inverse se pose parfois. Dans ce bût, un domaine spécial a été créé : in-addr.arpa.

Pour chaque réseau, on crée dans ce domaine une entrée correspondante. Par exemple, pour le réseau 134.158.0.0 (réseau de classe B de l'In2p3), on crée un domaine 158.134.in-addr.arpa et on en délègue l'administration à l'administrateur du réseau de l'In2p3.

Celui-ci, pour chaque machine qu'il déclare dans son domaine, crée, en plus de l'entrée « normale » dans in2p3 .fr, une entrée correspondante, sous forme d'un enregistrement PTR, dans le domaine 158.134.in-addr.arpa. Par exemple, pour la machine d'adresse 134.158.136.1, il créera un pointeur :

1.136.158.134 PTR lyon1.in2p3.fr

qui permettra de retrouver le nom de la machine à partir de son adresse.

Configuration élémentaire

Une machine demande en général simplement le nom de son domaine (par exemple « in2p3.fr ») et l'adresse du serveur de nom de son site, avec éventuellement une liste de serveurs de secours.

La configuration d'un serveur secondaire demande avant tout l'installation d'un logiciel adapté (par exemple bind sous Unix), puis la configuration du ou des domaines que l'on veut répliquer et la fréquence de duplication, ainsi que la liste des serveurs « root », nécessaires à toute résolution.

Pour un serveur primaire, il faut de plus maintenir à jour un fichier contenant tous les enregistrement afférents au domaine : le SAO bien sur, la liste des serveurs de nom (NS), puis la liste des machines et de leurs adresse, ainsi que leurs alias et les serveur de courrier. Il faut de plus maintenir pour chaque machine les entrées dans le domaine in-addr.arpa correspondant.

Chapitre 9 Connexion à distance

Une des premières utilisations d'un réseau est le possibilité de se connecter à une autre machine, c'est à dire, en travaillant physiquement sur un ordinateur, de pouvoir avoir un écran ou une fenêtre dans laquelle les commandes que l'on passe s'exécutent sur une autre machine, le réseau servant d'intermédiaire. C'est par exemple de cette manière que depuis un ordinateur personnel, on peut accéder aux fonctions d'un serveur.

Historiquement, deux protocoles ont été utilisé d'abord : rlogin et telnet. Ils sont maintenant progressivement remplacés par ssh.

Pour établir une telle connexion, il faut avoir un moyen tel que les caractères que l'on tape au clavier soient acheminé sur l'ordinateur auquel on est connecté, et que les caractères que celui-ci envoie soient affichés sur notre écran. Une connexion TCP offre toutes les caractéristiques nécessaires et ces méthodes de connexion en font toutes usage : on dispose d'un « tuyau » bidirectionnel acheminant les données entre nos deux machines. Les différents protocoles vont différés par leurs fonctionnalités.

Ce type de protocole fonctionne, comme la plupart des applications réseau, suivant un mode dit « client-serveur » : un programme appelé serveur attend une connexion sur un port connu, correspondant au service qu'il offre, et des programmes appelés clients le contactent sur se port.

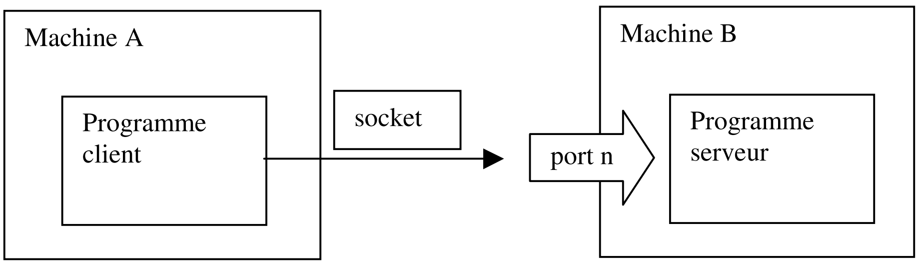

Figure 9 - Client-serveur

telnet et rlogin

rlogin est un protocole qui remonte aux premiers temps d'Unix et qui appartient à la séries des « r-commandes » : rsh, rexec, rcp,… qui furent les premiers protocoles réseau sur ce système. Ses fonctionnalités sont minimales : on se contente d'établir une connexion TCP sur un port bien connu du serveur : le port 513, puis les données sont transmises entre les deux machines.

Pour faciliter l'accès à des séries de machines (par exemple, une série d'ordinateurs utilisés pour faire du calcul numérique) sans avoir à fournir un mot de passe à chaque connexion, rlogin prévoit l'utilisation d'un fichier .rhosts sur la machine où s'exécute le serveur : ce fichier contient une liste de machines et d'utilisateurs pour lesquels il ne sera pas demandé d'identification. Ce fichier se présente sous la forme d'une série de ligne indiquant un nom de machine et d'utilisateur. Les clients rlogin se connectant depuis le compte de ces utilisateurs, venant de ces machines, n'ont pas à s'identifier. Pour s'assurer qu'il ne s'agit pas d'une mascarade, le serveur rlogin vérifie que la connexion provient d'un port privilégié (c'est à dire inférieur à 1024). Cette précaution n'est cependant pas incontournable et l'utilisation de rlogin peut entraîner de sérieux problèmes de sécurité.

rlogin est décrit dans le RFC1282.

telnet se propose de permettre les connexions entre machines dans un cadre beaucoup plus large. Il décrit la connexions depuis n'importe quel type de terminal sur n'importe quel type de machine. Pour cela, il utilise le concept de NVT : Network Virtual Terminal, un terminal doté des fonctions les plus primitives (en gros : un clavier et une Télétype). Toutes les fonctionnalités supplémentaires seront négociées sous la forme d'option. L'échange de ces options entre le client et le serveur se font par des données intercalées au milieu des données du terminal et identifiées en les faisant précéder d'un caractère spécial : IAC : Interpret As Command, de code 255 (0xFF en hexadécimal). Par exemple, le NVT est par défaut uniquement capable de fonctionner sous le mode « go ahead » : il n'émet des données que quand il y est invité par le serveur. Une des premières options négociées par le client est alors de supprimer ce mode « maitre-esclave ».

Dans le fonctionnement « normal » de telnet, chaque caractère frappé sur le clavier du client en envoyé au serveur par TCP, ce qui entraîne l'ajout de 20 octets pour TCP, puis 20 pour IP, sans tenir compte de ce qui est ajouté par la couche liaison (par exemple Ethernet). Ces données sont acheminées jusqu'au serveur qui remet cet octet à l'application qui se trouve à l'autre bout de la connexion, par exemple un éditeur de texte. Celui-ci renvoie alors la série d'octets servant de réponse, par exemple le contenu de la dernière commande tapée si on a appuyé sur la touche « curseur vers le haut ». Ces données sont à leur tour enveloppées dans TCP, IP,… La surcharge de trafic est supportable sur un réseau local à haut débit, mais prohibitive sur une liaison lente. Les protocoles tels que TCP et PPP contiennent des algorithmes complexes pour minimiser cette surcharge.

Un client telnet est disponible sur toutes les plates-formes, soit directement (Unix et les version récentes de Windows), soit comme programme additionnel, par exemple BetterTelnet pour MacIntosh ou Ewan sur Windows. Un serveur telnet n'est disponible que sur les machines multi-utilisateurs : Unix et Windows NT/2000.

telnet utilise le port 23.

telnet est décrit dans le RFC854, la négociation des option dans le RFC855 et les nombreuses options dans une multitude de RFCs, par exemple :

- echo : RFC857
- suppress go ahead : RFC 858
- terminal type : RFC1091

ssh

rlogin, comme telnet, présentent potentiellement un très gros risque pour la sécurité : dans tous les cas, le mot de passe circule sur le réseau et peut être facilement lu (sur un réseau Ethernet, toutes les stations « voient » les datagrammes de toutes les autres). Des logiciels, appelés « sniffer », sont spécialisés à ce petit jeu. Un nouveau protocole a été développé pour offrir une méthode de connexion plus sure : SSH : Secure SHell.

SSH utilise plusieurs méthodes successives pour augmenter la sécurité de la connexion :

- tout d'abord, le client et le serveur s'identifient l'un et l'autre au moyen d'un système de clés numériques, pour limiter la vulnérabilité à une mascarade.
- ensuite, ils échangent une clé secrète à l'aide de l'algorithme RSA (Rivest, Shamir, Adleman : il permet d'échanger une information secrète sans possibilité d'interception par un tiers à l'écoute).
- à partir de là, toute le communication est encryptée à l'aide de cette clé

L'ensemble des données échangées sur une connexion ssh est ainsi protégée d'une écoute par un tiers.

Ssh offre de plus des possibilités d'utiliser le « tuyau » ainsi créé pour transmettre d'autre type de données, en particulier la connexion à un écran graphique « X-Windows ».

Le protocole ssh permet aussi l'exécution de commandes à distance.

Ssh utilise le port 22 et est décrit non pas dans un RFC, mais dans un Internet Draft..

L'utilastion typique de ssh est :

ssh user@machine

Un client ssh est disponible sur la plupart des systèmes, par exemple, « putty » pour Windows.

Chapitre 10 Transfert de fichier s

Une autre application fort utile dans un réseau est le transfert de fichiers entre machines. Celà peut être aussi bien l'échange de fichiers entre un ordinateur portable et une station fixe ou la récupération d'un document se trouvant sur une machine lointaine.

Le premier protocole utilisé dans ce but dans l'Internet est FTP : File Transfert Protocol. Cependant, il pose le même problème de sécurité que telnet : les mots de passe circulent en clair sur le réseau. Pour éviter cela, le protocole ssh prévoit la possibilité de transférer des fichiers de manière sure : scp : Secure CoPy.

Il est à noter que si le Web utilise son propre protocole, il utilise aussi beaucoup ftp pour rapatrier des documents.

Un protocole nommé tftp : Trivial File Transfert Protocol permet lui aussi de transférer des fichiers entre deux machine, mais n'est pas destiné à un usage par des humain : il est destiné à la récupération automatique de fichiers par les machines, par exemple l'image de son système d'exploitation pour une machine sans disque.

ftp

Pour permettre le transfert de fichiers, ftp va bien sur utiliser une connexion TCP. Il va même en utiliser plusieurs ! A la connexion, ftp va ouvrir une première connexion sur le port 21 de la machine, ou doit normalement attendre le serveur ftp. Cette connexion est appelée circuit de commande et c'est par elle que vont transiter les ordres passés au serveur. Le dialogue entre le client et le serveur utilise le même format que telnet (on peut d'ailleurs très bien se connecter sur un serveur ftp à l'aide d'un client telnet : il suffit de contacter le port 21 au lieu du port 23).

Le dialogue comprend :

- Une série de commandes que le client passe au serveur :
  - Identification : USER, PASS
  - Gestion des répertoires : CWD, MKD, RMD, PWD, LIST.
  - Envoi et réception de fichiers : STOR, RETR
  - Fin de connexion : QUIT
  - …

- Des réponses faites par le serveur sous la forme d'un nombre de 3 chiffres suivi d'un texte explicatif. Le premier chiffre indique le type de réponse : 1,2,3 : réponse positive, 4 : réponse négative temporaire, 5 : réponse négative définitive. Le deuxième chiffre indique ce qui est concerné par cette réponse, par exemple 0 concerne la syntaxe et le dernier chiffre permet de préciser.

Pour transférer les fichiers, ftp utilise pour chaque transfert un nouveau socket : le circuit de données . Le client choisi un port (non privilégié), se met en attente dessus, puis l'indique au serveur par une commande PORT. Celui-ci prend alors l'initiative de la connexion sur le port indiqué, en utilisant le port 20 de son coté et le transfert de donnée se fait, dans un sens ou l'autre suivant que l'ordre a été RETR ou STOR.

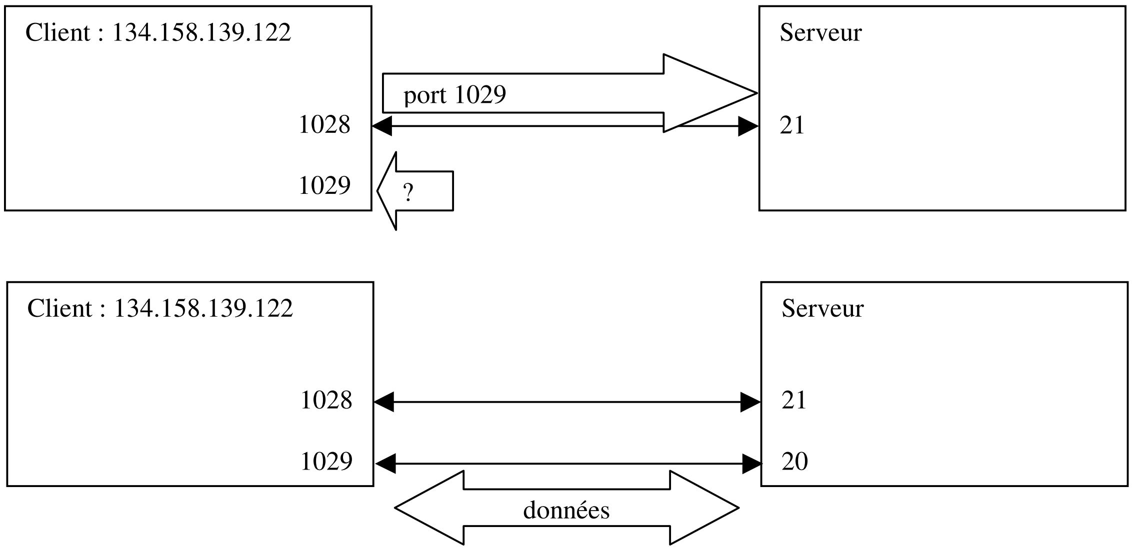

Figure 10 - Circuit de données ftp

Une commande : TYPE permet d'indiquer le type de fichier transféré : ASCII ou binaire. Dans un transfert ASCII, les caractères de fin de ligne peuvent être modifiés suivant le système de fichiers utilisé. Un fichier binaire transféré en mode ASCII risque donc fortement d'être altéré.

Les commandes du client ftp sont :

- open, close : gestion de la connexion
- user, pass : identification
- cd, pwd, mkdir, rmdir : gestion des répertoires
- ls, dir : affichage des fichiers
- binary, ascii : type de fichiers
- get, mget, put, mput : transfert de fichiers (m pour multiple)
- delete, mdelete : suppression de fichiers
- prompt : suppression de la demande de confirmation pour les opérations sur plusieurs fichiers (mget, mput, mdelete)
- lcd : changement de répertoire sur le client
- ! exécution d'une commande sur le client.
- reget : reprise d'un transfert interrompu.

Normalement, ftp demande une identification de l'utilisateur par un mot de passe. On a alors accès aux même fichiers que si l'on était connecté directement sur la machine. Il existe cependant une possibilité, pour des serveurs mettant des documents en accès public, d'un mode de fonctionnement dit « ftp anonyme ». Dans ce mode, on fournit « anonymous » comme nom d'utilisateur. Le serveur demande alors de fournir une adresse de courrier électronique. La validité de celle-ci n'est (en général) pas vérifiée, mais il est de coutume de fournir sa véritable adresse. En général, l'ensemble des commandes passées dans ce mode est gardé dans un journal pour des raisons de sécurité. Ce type de fonctionnement est très utilisé pour mettre des document en accès au plus grand nombre : les RFCs sont diffusés ainsi, mais aussi de nombreux logiciels gratuits (Linux, les logiciels « GNU »,…), des versions de démonstration de logiciels commerciaux, des correctifs (« patch »), de la documentation, etc…

Ftp utilise le port 21 pour le circuit de commande et 20 pour le circuit de données.

Ftp est décrit dans le RFC 959 datant de 1985 à la suite d'une série de 42 RFC dont le premier remonte à 1971 !

Un client ftp est disponible sur les machines Unix et les versions récentes de Windows. Sur MacIntosh, on peut utiliser fetch. Un serveur est disponible sur les machines multi-utilisateur : Unix, Windows NT/2000.

scp

Comme on l'a vu, ftp pose un sérieux problème de sécurité lié à la circulation en clair des mots de passe. Scp permet d'un remédier en utilisant la même méthode que ssh : les machines s'identifient mutuellement, puis conviennent d'une clé secrète qui sera utilisée ensuite pour encryter l'ensemble du transfert, y compris le passage du mot de passe.

Un client scp est disponible sur les machines Unix et Windows.

Chapitre 11 Le courrier électronique

Le courrier électronique est probablement la plus ancienne application utilisée sur les réseau, bien avant l'apparition de TCP/IP.

Un des premiers protocole utilisé fut NJE : Network Job Entry, un protocole d'IBM destiné , comme son nom l'indique, à faire exécuter un travail par lot (« batch ») sur une machine distante. L'envoie d'un message consistait alors à faire exécuter un batch un peu particulier dont le travail était de remettre le corps du message au destinataire. Ce protocole donna lieu à un des premier réseau accadémique reliant les universités américaines : Bitnet. Ce réseau eut bientôt un pendant européen : Earn : European Accademic Research Network.Ces deux réseaux étaient reliés par une liaison à 9600 bits/s arrivant à Montpellier, ce qui permettait tout de même d'acheminer environ 40000 messages par jour !

Dans le monde Unix, un protocole nommé uucp : Unix to Unix CoPy permettait lui aussi l'acheminement de messages. L'envoi était fait sur des lignes téléphoniques par modems : à intervalles régulier, un système vérifiait s'il y avait des messages à acheminer et un appel téléphonique était éventuellement fait pour transmettre le message à une autre machine, et ainsi de suite jusqu'à la remise au destinataire. Il est à noter que cela demandait l'organisation d'une structure analogue au routage de IP, mais avec des messages à la place des paquets.

Les divers systèmes d'exploitation eurent ainsi très tôt leur système de courrier électronique.

Le modèle OSI normalisa aussi le courrier électronique : X400. Ce fut la seule application du modèle qui reçu vraiment une réalisation pratique et des réseaux publics basés sus X400 furent mis en place, par exemple Atlas400 en France.

Tous ces réseaux cohabitaient et l'échange entre eux se faisait au travers de passerelles : des logiciels spécialisés recevaient les courriers électroniques dans un protocole et les traduisaient dans un autre de manière à pouvoir atteindre un correspondant connecté à un autre réseau.

Le monde de l'Internet avait lui aussi son protocole : SMTP : Simple Mail Transfer Protocol ainsi qu'un modèle sophistiqué de l'organisation d'un système de messagerie. Ce protocole profita de l'expansion de l'Internet et su évoluer pour répondre à la demande de transport de documents plus complexes qu'un simple texte ASCII.

Certaines applications reposent sur l'utilisation du courrier électronique, et en particulier les listes de diffusion : des serveurs spécialisés, tels que listserv ou majordomo diffusent à un ensemble de destinataires les messages qu'ils reçoivent sur un sujet particulier. On s'inscrit sur une telle liste en envoyant un message au serveur avec comme contenu un texte du genre : « subscribe une_liste ». Ces liste de diffusion servent aussi bien à des débats qu'à la diffusion de messages importants, par exemple la description de nouveaux virus.

Un dernier point de vocabulaire : en anglais, on appelle mail ou e-mail un courrier électronique, et en français : mél : Message ELectronique

Modèle du courrier électronique

Le modèle actuellement utilisé pour le courrier électronique repose sur une répartition entre organisations. Chacune d'elle est vue de l'extérieur sous le forme d'un MTA (Message Transfert Agent) relai par lequel transitent tous les messages. Ces MTAs relais assument tous les échanges de messages entre organisations au moyen du protocole SMTP. Au sein de l'organisation, des MTAs locaux acheminent les messages, soit entre eux, soit à destination des MTAs relais pour les messages destinés à l'extérieur, toujours par le protocole SMTP. Les utilisateurs accèdent au courrier à l'aide d'un UA : User Agent.

Celui-ci permet de confier un message à un MTA local. Pour la consultation du courrier, différentes solutions sont possibles :

- la consultation directe dans un « spool » ou un système de fichiers partagés.
- le relevé à distance grâce à un protocole pouvant être SMTP, ou un protocole spécialisé tel que POP ou IMAP :

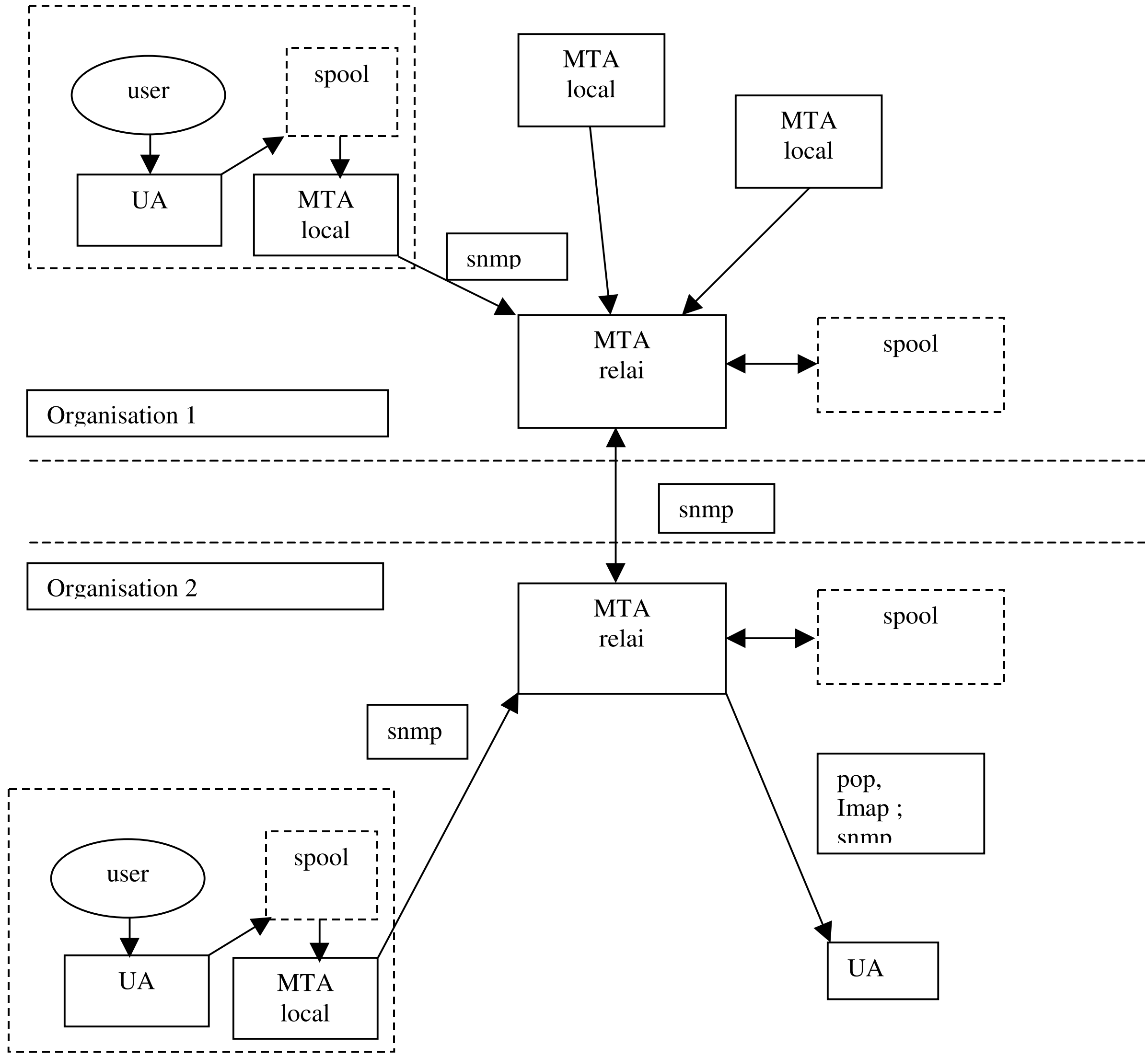

Figure 11 - Modèle pour le courrier électronique

Un des MTAs les plus utilisés est le sendmail de Unix, connu par sa puissance et la complexité de sa configuration.

Les adresses

Pour désigner le destinataire, la forme suivante d'adresses est utilisée :

Destinataire@localisation

Le destinataire est à priori le nom d'un utilisateur sur une machine, mais par le jeu des réécritures dans les MTA, peut être un nom patronymique, par exemple : Steve.Jobs, ou un nom générique, par exemple : user.support, ou webmaster.

La localisation est à priori le nom d'une machine telle qu'elle est connue du serveur de nom, mais peut être, par l'emploi des enregistrements MX dans le serveur de nom, un nom organisationnel tel que ipnl.in2p3.fr : bien qu'aucune machine ne porte ce nom, il existe par contre un enregistrement MX dans le serveur de nom indiquant que tout courrier adressé à cette localisation doit être remis à la machine gérant le MTA relais pour cet organisme. D'une manière générale, un organisme s'insérant dans le schéma précédent configure un MTA relais pour son courrier et le déclare à l'aide d'un enregistrement MX dans le serveur de nom.

Format d'un courrier : RFC 822

Le format des messages échangés est décrit dans le RCF822. En simplifiant, le message est constitué :
- d'un corps, contenant le message proprement dit
- d'un en-tête, rempli par le UA et contenant des informations telles que l'auteur, la date, etc… Cet en-tête est complété au fur et à mesure par les MTAs de transit.
- d'une enveloppe remplie par les MTA par lesquels transite le message.

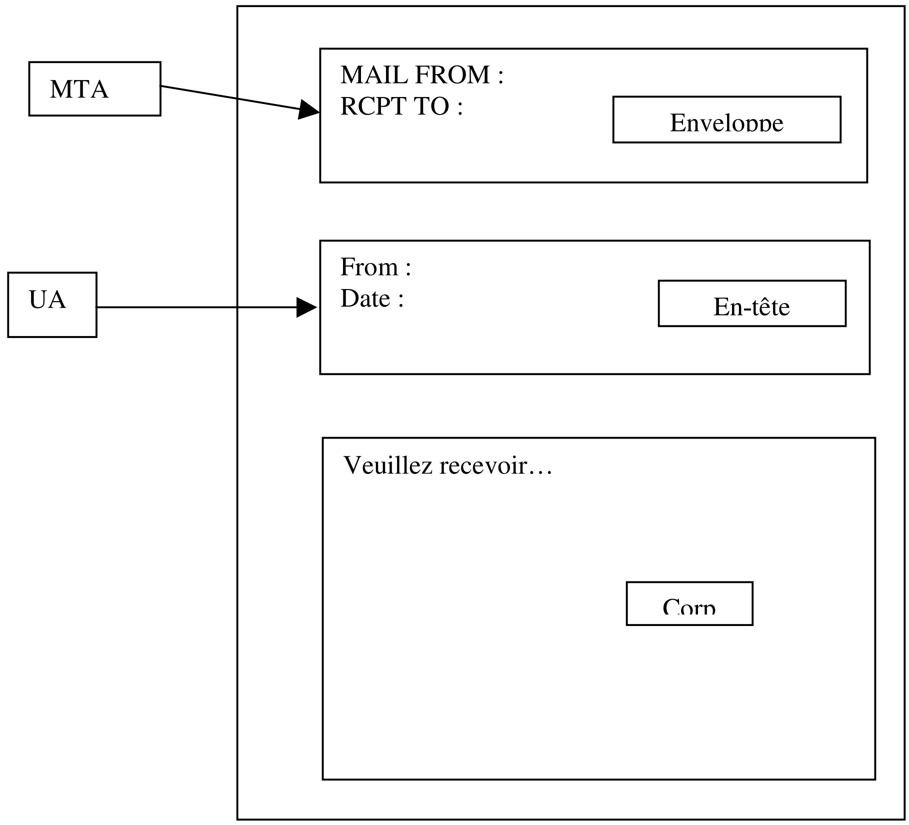

Figure 12 - Format d'un message

Le format des lignes de l'en-tête est décrit dans le RFC822. Il comprend des champs tels que :

- les traces des relais : Received from… by…
- les informations sur l'expéditeur : From :…, Sender :…, Reply-to :…,…
- le ou les destinataires : TO/…, cc :…, bcc :…

- des informations : Message-ID :…, Date :…, Subject :…, Comment :…,…
- l'adresse de retour : Return-Path :…
- des références : In-Reply-To :…,…
- des champs privés : X-Face :…,…

Des extensions ont été prévues pour placer dans ces champs des caractères plus généraux que l'ASCII 7 bits : RFC1522. Il permettent l'utilisation de caractère des alphabets ISO-8859 en les codant soit en « quoted printable » (le caractère est codé sous la forme « =nn » où nn est son code hexadécimal), soit en Base64.

Transfert de courrier : smtp

Le protocole servant à tous les niveaux au transfert de messages est SMTP : Simple Mail Transfert Protocol..

Ce protocole utilise une connexion TCP sur le port 25.

Le dialogue se fait sous la même forme qu'un dialogue telnet, avec des messages en ASCII. On peut parfaitement transmettre un courrier en se connectant par telnet sur le port 25 d'une machine opérant un MTA.

Les commandes émises pour transmettre un message sont de la forme/

```
HELO hostname
MAIL FROM : user@localisation
RCPT TO : user-d@localisation-d
DATA
Veuillez, très cher….
….
.
QUIT
```

A chaque commande, une réponse sous la forme d'un nombre de trois chiffres et d'un texte est envoyé, d'une manière similaire à ftp (il existe cependant quelques différences).

SMTP est décrit dans le RFC821.

SMTP est toujours utilisé, mais une version étendue existe : ESMTP : Extended SMTP. Les extensions sont introduites sous la forme d'option. La modification est très simple : si le dialogue commence par « EHLO » au lieu de « HELO », le serveur répond par la liste des options qu'il supporte. Chaque option est décrite dans un nouveau RFC.

ESMTP est décrit dans le RFC1869, quelques options sont décrites dans les RFs suivants :

- 8BITMIME : RFC1426, le corps du message contient des caractères 8 bit (au lieu de l'ASCII 7bit standard)
- SIZE : RFC 1870, le client annonce la taille du message avant de le transmettre.

Transport de données : Mime

Pour transportées des données plus riches que le simple ASCII 7 bits, on utilise une extension concernant le contenu du corps du message : MIME : Multipurpose Internet Mail Extensions.

Le principe est d'ajouter des en-têtes précisant :

- la version de MIME : Mime-Version :…
- le type du contenu : Content-Type :… Les types prévus sont : text, multipart, message, application, image, audio, video. Le champ décrivant le type peut être complété par un sous-type et des paramètres, par exemple : « TEXT/PLAIN ;charset=US-SACII ».

• la manière dont ce contenu a été codé pour le transférer : Content-Transfer-Encoding. Les codages prévus sont : 7bit, quoted-printable, base64, 8bit, binary.

MIME est décrit dans la série des RFC 2045 à 2049..

Accès au courrier

Pour relever son courrier, un UA peut utiliser essentiellement deux protocole : pop et imap. Pop est maintenant dépassé et progressivement remplacé.

POP

POP : Post Office Protocol en est à sa version 3 et est décrit dans le RFC 1939.

POP utilise une connexion TCP sur le port 110 pour atteindre le serveur.

Le dialogue utilisé est similaire à celui de FTP :

- identification par USER, PASS
- liste des messages : LIST
- récupération de message : RETR
- effacement d'un message : DELE
- fin de session et mise à jour : QUIT

Les réponses du serveur se limitent à :

- +OK texte
- -ERR texte

IMAP

IMAP : Internet Message Access Protocol en est à sa version 4 et est décrit dans le RFC2060.

Il utilise une connexion TCP sur le port 143.

Comme POP, il permet de se connecter à une boite à lettre et de relever son contenu, mais il offre plusieurs facilités :

- gestion des messages sur le serveur : POP se contente de permettre de rapatrier les courriers. IMAP permet de les gérer sur le serveur, avec en particulier la possibilité de gérer des dossier. Un serveur IMAP est donc destiné à contenir l'ensemble des courriers que l'on peut consulter et manipuler depuis n'importe quelle machine.
- le dialogue avec le serveur IMAP peut être encrypté.

La confidentialité

Comme dans tous les protocoles Internet de la première génération, les protocoles de courrier font transiter les messages en clair (sauf en option IMAP). La nécessité d'avoir une confidentialité au niveau des messages échangés a conduit au développement de PGP : Pretty Good Privacy par P. Zimmerman.

PGP fonctionne d'une manière similaire à SSH :

- échange d'une clé secrète par RSA. Contrairement à SSH, au moment de l'émission du message, le client n'est pas en connexion directe avec la machine destinataire. La génération de la clé secrète se fait donc en consultant la clé publique de ce destinataire. Celui-ci doit la publier par un moyen quelconque : finger, site web, dépositaire,…

- cryptage du message par un algorithme fort (IDEA) et envoi par les procédures standard (SMTP).
- à réception du message, le destinataire retrouve la clé secrète en utilisant la clé publique de l'auteur.
- le système de clés publiques permet de garantir la provenance du message et d'éviter les mascarades.
- le système prévoit une possibilité de signature électronique (le message ne peut pas être répudié).

Dans l'état actuel de la législation, l'utilisation de PGP n'est pas autorisée en France.

Les logiciels de courrier

Un nombre impressionnant de logiciels de courrier électronique existe, depuis le primitif mail jusqu'à Outlook. Pour n'en citer que quelques-uns :

Pine

Sous Unix il travaille dans un terminal en mode caractère : pas de fioritures, juste de l'efficacité. Il connaît les caractères 8 bit, sait gérer les extensions MIME, utilise pop ou imap, est paramétrable à l'infini (mais est utilisable tel quel).

Netscape

Le navigateur universel sait aussi faire du courrier. Pas très bien, certes, mais il le fait.

Outlook

Le courrier électronique vu par Microsoft : l'artillerie lourde. Il sait tout faire, y compris les macros VBA qui en font une cible en or pour les virus. Cest le lecteur de courrier qu'utilise Internet Explorer.

Eudora

Une interface graphique de courrier, utilisable sous Windows et MacIntosh, il ne fait que du courrier, mais le fait bien.

Chapitre 12 Les « news »

Les « news » sont un des plus vieux domaines de l'Internet. C'est aussi le domaine qui a le plus de folklore, avec ses règles, son histoire, ses traditions et ses personnages. Il s'agit de forums de discussion, organisés par domaine avec toute une hiérarchie.

Organisation des forums

Les news sont organisés en groupes de discussion : par exemple comp pour ce qui a trait aux ordinateurs, sci pour les groupes scientifiques, fr pour les groupes francophones. Chaque domaine est découpé en groupes avec une hiérarchie : par exemple fr.comp.sys.mac pour les discussions en français sur le système MacIntosh. L'ensemble de ces groupes ou forums forme Usenet. On a coutume de désigner les groupes par leurs initiales, par exemple FUFE pour fr.usenet.forum.evolution.

La création de nouveaux groupes obéit à un rituel formalisé sous le nom de vote. L'intitulé du nouveau groupe et sa charte sont discutés dans un premier temps (par exemple, dans FUFE) et quand un texte est adopté, on passe au vote pour ou contre la création du nouveau groupe. Celui-ci est éventuellement créé en cas de vote positif.

Un groupe peut être modéré, c'est à dire que les messages sont adressés à un modérateur qui juge s'il sont conformes à la charte du groupe.

Un domaine particulier est alt (pour alternatif) où les règles usuelles ne s'appliquent pas trop : pas de modérateur, bien sur, mais aussi création anarchique des groupes. Ce domaine n'est plus trop propagé à la suite d'abus, en particulier la propagation d'images illégales.

A coté des grands domaines propagés de partout, on peut créer aussi des groupes locaux avec ses propres règles.

Architecture des news

Les news ont une organisation anarchique dans le sens où il n 'existe aucune hiérarchie. Les serveurs de news propagent les groupes qu'ils souhaitent, sous leur seule responsabilité. Pour se raccorder à Usenet, il suffit d'obtenir l'accord d'un serveur avec lequel on s'accorde sur les groupes qu'il vous envoie et ce que vous pouvez lui transmettre.

La plupart des organisations et les fournisseurs d'accès Internet installent en général un serveur de news qu'ils raccordent à un serveur avec lequel ils ont passé un accord. Il créent en général des groupes locaux qu'ils ne propagent pas à l'extérieur, par exemple in2p3.reseau. Le serveur conserve les articles qu'il a collecté pendant un temps limité.

Le nombre de groupes, même si l'on se limite aux groupes propagés à peu près partout, est difficile à évaluer à cause de alt qui est propagé de manière anarchique, de l'ordre de quelques milliers.

L'ensemble de ces serveurs interconnectés avec leurs règles locale constituent un système coopératif qui propage les message quasiment sans contrôle.

Le protocole utilisé pour la propagation des messages est NNTP : Network News Transfert Protocol.

Il est très proche de FTP et SMTP dans son dialogue : il s'agit de commandes textuelles avec des réponses sous forme d'un nombre de trois chiffres, d'un texte et éventuellement d'un message ou d'une liste.

Le protocole utilise une connexion TCP sur le port 119.

Les commandes permettent :

- d'afficher le liste des messages d'un groupe
- d'obtenir un message soit individuellement, soit les uns à la suite des autres.
- d'obtenir la liste des groupes disponibles
- d'annoncer la création ou la suppression de groupes.
- de poster un nouveau message

Le protocole est aussi bien utilisé pour le dialogue entre les serveurs que pour le dialogue entre un client et son serveur.

Le protocole est décrit dans le RFC977.

Logiciels de news

L'accès aux news se fait par un logiciel permettant de se connecter à un serveur de manière à récupérer les groupes voulus et éventuellement de poster des nouveaux messages.

Il est à noter que si l'on utilise le terme de s'abonner à un groupe, il n'y a aucune notion d'abonnement à un serveur. L'abonnement consiste juste à indiquer au logiciel les groupes qu'il doit demander au serveur.

Dans les logiciels d'accès au news, il est possible de masquer systématiquement les messages des auteurs que l'on considère comme indésirables : le « kill file ».

Enfin, il est possible de rapatrier en une seule fois le contenu des groupes souhaités, puis de consulter leur contenu « hors connexion » de manière à économiser le temps de connexion.

Netscape

Netscape contient bien sur un lecteur de news. Il n'est toutefois pas très pratique : on ne sait pas trop si un message est bien posté.

Outlook Express

Le logiciel de news utilisé par Internet Explorer.

tin

Un logiciel de news en mode texte : rapide, complet et efficace.

Archivage des news

Il existe des serveurs qui archivent les news pendant un temps plus ou moins long. Il permettent des recherches par auteur, par groupe, par thème, etc…

L'un des plus connu est DejaNews : www.dejanews.com ; récemment (Février 2001) racheté par Google.

Chapitre 13 Le « web »

Au début des années 90, des chercheurs du Cern cherchent à mettre au point une méthode pour réaliser une documentation hypertexte où les mots clés deviennent des liens actif permettant la navigation dans les document. Complété par une méthode pour transférer les documents sur le réseau, cela donne naissance au World Wide Web : WWW ou simplement le Web.

Format d'une page : html

Les pages Web sont écrites dans un langage nommé HTML : HyperText Markup Language.. Ce langage permet de décrire des documents structurés, c'est à dire organisés en sections hiérarchiques. L'élément de base du langage est la balise ou tag : ce sont des marqueurs, allant par paire, et qui marquent les limites d'une section sous la forme :

<SECTION> contenu de la section </SECTION>

On trouve des balises permettant :

- une hiérarchie chapitres, sous chapitres,… jusqu'aux paragraphes.
- différents éléments de décoration : caractères gras, séparateurs de zones, …
- des listes
- des images
- une structuration de page au moyen de cadres : les « frames ».

Une balise particulière est l'ancre : on indique une référence à un autre document (ou partie de document). Le champ spécifié apparaîtra marqué sur la page avec un moyen (un clic de souris) d'afficher ce document, ce qui constitue la base de la navigation hypertexte. Le format de l'ancre est :

<A HREF= « http://www.in2p3.fr/ »> In2p3 </A>

Dans cet exemple, le mot « In2p3 » sera marqué (en surbrillance par exemple) par le lecteur HTML et un clic sur ce mot entrainera le transfert du document référencé et son affichage.

HTML a connu plusieurs versions successives définies maintenant par le W3C. Sa version 2 est décrite dans le RFC1866.

Désignation des objets : les urls

La manière de désigner les document dans les ancres s'appelle URL : Universal Resource Locator.

La forme la plus générale d'une URL est :

<schéma :partie_spécifique_au schéma>

Le schéma indique à quelle action correspond l'URL. Il en existe de nombreux définis, le plus commun étant http. On trouve entre autres :

- http : pointe sur un document Web
- ftp : pointe sur un fichier récupérable par ftp
- news : pointe sur un article des news
- maito : indiquent une adresse de courrier électronique
- file : pointent sur un fichier local

La partie spécifique, quand elle concerne les protocoles Internet, a en général la forme :

//<utilisateur> :<mot de passe>@<machine> :<port>/<désignation>

Tous les champs sont facultatifs et peuvent être employés dans diverses combinaisons en fonction du besoin. Par exemple, le numéro de port a pour valeur par défaut le numéro utilisé habituellement par le mode de transfert concerné : 80 pour HTTP.

Les URLs sont décrits dans le RFC1738.

Transfert des informations : http

Pour récupérer le contenu d'une URL de type http, il faut un protocole de transport : HTTP: HyperText Transport Protocol Le protocole est particulièrement complexe. Il va en effet bien au-delà du simple « GET » permettant de transférer la ressource. Il permet de plus:

- l'identification du correspondant
- l'identification du serveur par un tiers de confiance de manière à pouvoir réaliser des transactions financières sécurisées
- le cryptage de certaines données
- la récupération d'informations chez le client au moyen de « formes de saisie » (FORMs).
- le transfert de fichiers du client vers le serveur
- le fonctionnement des « caches » : ce sont des serveurs placés à des endroits stratégiques qui stockent les pages demandées de manière à les réutiliser si elle sont à nouveau demandées.
- les redirection, c'est à dire le renvoi sur un autre URL quand on en demande un.

Mis à part ces complications, le protocole est du même type que FTP : une série de commandes sous forme de texte et des réponses avec un nombre de 3 chiffres suivi d'un texte de réponse.

Par défaut, HTTP utilise le port 80.

HTTP est décrit dans le RFC2068.

Les cookies

Une des possibilités offerte par HTTP est le transfert de « cookies » : un serveur remet au client une suite de caractères, le cookie, qui permet ensuite au serveur de « reconnaître » le client.

Pages dynamiques : JavaScript et Java

HTML permet l'exécution de programmes sur le client (le navigateur). Deux langages sont particulièrement utilisés :

- JavaScript : un langage interprété mis au point par Netscape et permettant l'exécution de taches simples dans le serveur.
- Java : un langage très puissant développé par Sun. Il s'agit d'un langage objet qui est compilé en un fichier binaire pouvant être exécuté sur n'importe quelle machine par une « machine virtuelle Java ». On peut ainsi réaliser des programmes universels. Bien que le cadre de Java dépasse largement le Web, il permet toutefois d'écrire des programmes appelés applets qui s 'exécutent dans le navigateur.

Les lecteurs

Il existe de nombreux logiciels permettant de naviguer sur le Web. Parmi ceux-ci, on peut noter :

Netscape

La compagnie Netscape a été fondées pour développer des logiciels autour du Web, et en particulier leur navigateur Netscape Navigator/Communicator. Il s'agit de logiciels a vocation universelle permettant aussi bien la navigation que le courrier électronique, la consultation des news ou l'éditions de pages Web.

Netscape réalise des navigateurs pour la plupart des plates-formes, aussi bien Unix que Mac et PC.

Internet explorer

Internet Explorer est la réponse de Microsoft à Netscape. Il offre le même genre de possibilités, mais ne fonctionne que sur Mac et PC. Dans ce dernier cas, il est entièrement intégré au système d'exploitation depuis Windows 98.

Lynx

Les navigateurs Web ont aussi leur outil en mode text : Lynx. Il ne sait bien sur ni afficher des images, ni exécuter du Java ou du JavaScript. Par contre, il est imbattable pour naviguer rapidement dans des pages de texte.

Les indexeurs

L'information contenue dans le Web est prodigieuse et ne cesse de croître. Mais la recherche de cette information , n'est possible qu'à l'aide de moteurs de recherche : les indexeurs : ce sont des logiciels qui parcourent sans arrêt le Web en suivant les URLs les une après les autres, analyse le contenu des pages et stockent cette information dans des bases de données. Des serveurs accédant à ces bases peuvent alors fournir des listes d'URLs se référant à un ou des mots clés donnés.

Parmis ces indexeurs, on peut citer :

Altavista

Le premier vraiment efficace et sans doute celui qui gère la base de données la plus importante.

google

Il stocke ses information en essayant de les classer par thèmes. Il répond à une vitesse invraisemblable et donne très souvent l'URL que l'on cherchait dans ses premières réponses.

Chapitre 14 « X Window »

La connexion aux ordinateurs s'est longtemps faite à l'aide de terminaux fonctionnant en mode texte pour deux raisons : la faible vitesse des liaisons et le coût des équipements graphiques.

Vers le milieu des années 80, l'apparition des réseau locaux rapides et des stations de travail a rendu possible le développement du travail à l'aide de poste graphiques. Le résultat des études menées au MIT : Massachusetts Institute of Technology ont conduit au protocole X Window ou encore X11.

Le modèle du MIT

Le modèle du MIT repose sur le concept de serveur X : il s'agit d'un écran graphique, d'un clavier et d'une souris, avec un logiciel offrant les fonctions prévues par X11 : les événements se produisant sur le terminal sont envoyés au programmes qui en font la demande et ces mêmes programmes peuvent réaliser un affichage sur l'écran.

Ces programmes sont appelés les clients X.

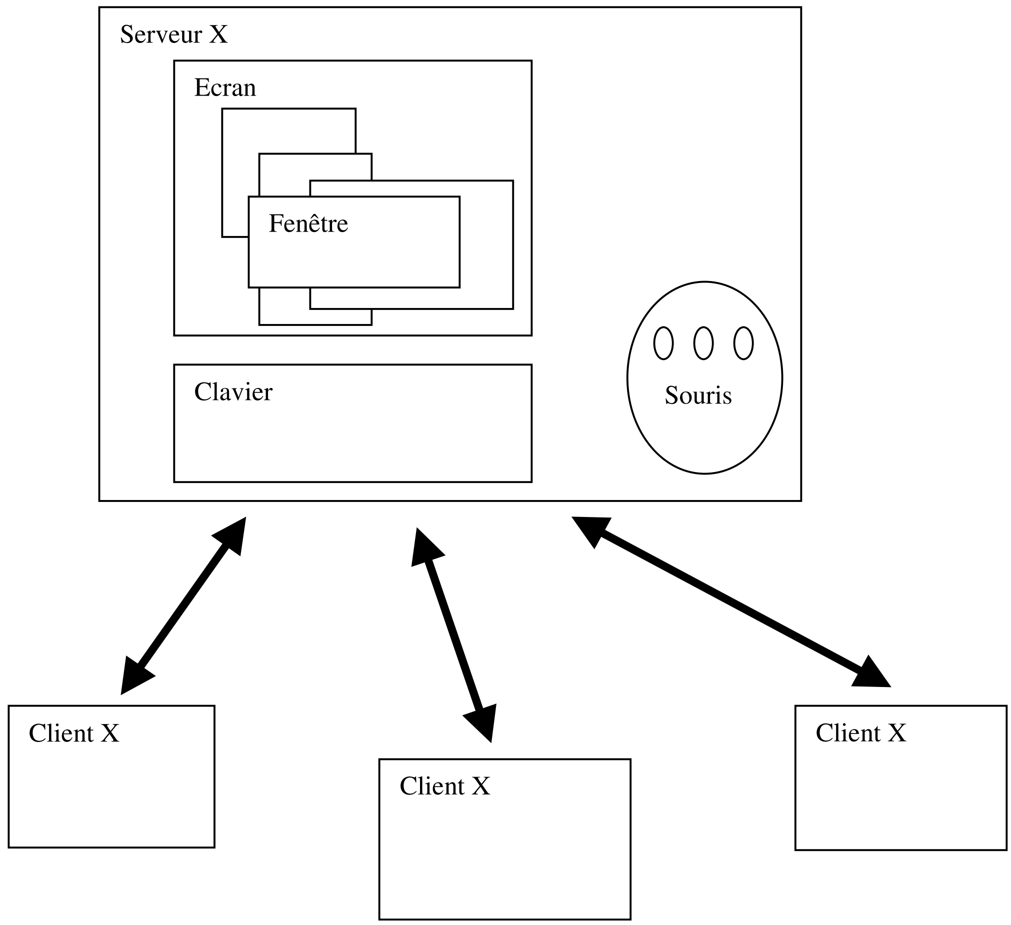

Figure 13 - Le modèle X Window

Le serveur X window

Le serveur X offre donc aux clients des services pour l'affichage et la gestion du clavier et de l'écran.

Les clients se connectent au serveur par une connexion TCP sur le port 6000.

La protection du serveur est critique dans ce modèle : le serveur offre à priori ses services à tous les clients qui en font la demande, et ses services vont du contenu des fenêtres au touches pressées sur le clavier. Il est donc indispensable qu'il y ait un mécanisme protégeant le serveur de clients mal intentionnés. Deux mécanismes sont prévus :

- la commande xhost gère la liste des machines autorisées à se connecter au serveur X. Il est à noter que cette protection est loin d'être suffisante, puisque ce sont l'ensemble des clients tournant sur ces machines qui sont autorisés, ce qui n'est pas assez restrictif dans le cas de machines sur lesquelles de nombreux utilisateurs peuvent se connecter.

- les « magic cookies » : il s'agit d'un nombre pseudo-aléatoire généré par le serveur au moment où il démarre. Tout client doit fournir ce nombre pour être autorisé à utiliser les services du serveur X. Les cookies sont stockés dans le fichier .Xauthority et manipulés à l'aide de la commande xauth. Quand on veut utiliser un client X sur une machine, on copie le cookie a l'aide de la commande xauth et on en place une copie sur la machine où doit tourner le client. Celui-ci cherche automatiquement le cookie correspondant au serveur sur lequel il doit afficher.

Le dialogue entre le client et le serveur comprend :

- des requêtes demandées au serveur par le client, par exemple : « create window », « get motion events », etc…

- des événements envoyés au client par le serveur, tels que « mouse button pushed », « keyboard key pressed », « window exposed », « window resized », etc…

Il existe ainsi plus de 150 messages possibles.

X Window est décrit dans le RFC1013 et il existe une abondante documentation livrée avec X Window. La version utilisée actuellement est X Version 11 Release 6, aussi nommée X11R6.

Les clients

Les clients X sont extrêmement nombreux et offrent une très grande variété d'applications, depuis l'indispensable xterm qui offre un terminal en mode texte à de nombreux jeux. Il est de plus facile d'en écrire à l'aide de « toolkits » tels que Tcl/Tk ou GT.

Pour identifier le serveur X qu'il doit contacter, le client utilise une variable nommée DISPLAY, ou si elle n'existe pas, les paramètre qu'on lui a passé. La variable identifie le serveur sous la forme :

machine :0 :0

où « machine » est le nom internet ou l'adresse IP de la machine.

Il est à noter que si l'on utilise ssh pour se connecter sur une machine, celui-ci gère lui-même les variables DISPLAY et achemine les événements X dans un canal encrypté. Le lancement des clients depuis une telle session est donc transparent et sur.

XDMCP et CDE

Un serveur X sait offrir des services à des clients, mais pour lancer ceux-ci, il faut au préalable s'être connecté sur une machine. XDMCP (X Display Manager Control Protocol) permet d'offrir au démarrage du serveur une liste des machines sur lesquelles il peut se connecter, puis ayant choisi l'une d'elle, d'offrir une bannière de login pour se connecter et lancer une série de client offrant un environnement graphique complet, tel que CDE : Common Desktop Environment.

XDMCP utilise des messages UDP sur le port 117.

Serveurs X

De nombreux serveurs X, gratuits ou payants, sont disponibles sur le marché. On peut noter parmi eux :

XFree86

Surtout connu pour son utilisation avec Linux, il est disponible avec bien d'autres systèmes d'exploitation. Il est gratuit et en perpétuelle évolution, notamment pour le support des nouvelles cartes graphiques.

Exceed

Ce logiciel commercial de la société Hummingbird permet d'avoir un serveur X sur une machine Windows et d'en faire un terminal X.

Chapitre 15 Synchronisation des horloges

Le fonctionnement normal d'un réseau de machines nécessite, ne serait ce que pour synchroniser leurs journaux, d'avoir les horloges de ces machines synchronisées.

Le protocole : ntp

Le protocole utilisé pour la synchronisation des hormoges s'appelle NTP : Network Time Protocol. Le principe en est relativement simple : une série de centres spécialisés dans le monde entretiennent des horloges atomiques et, par comparaison entre eux, détermine un temps de référence, dit de niveau 0.

Ce temps de référence est diffusé par différents moyen, dont les ondes radio en grandes ondes. Des machine équipées de récepteurs adaptés synchronisent leurs horloges et constituent des serveur de niveau 1.

A leur tour, elle servent à d'autres machine qui se connectent à elles pour synchroniser des horloges de niveau 2, et ainsi de suite.

Le dialogue entre les machines se fait par des paquets UDP sur le port 123. Tous les paquets comportent des marqueurs temporels qui permettent, par une analyse statistique assez complexe, d'obtenir les temps de propagation et leur précision de manière à effectuer les corrections nécessaires.

NTP est décrit dans le RFC1305.

Configuration simple

La première chose à faire pour utiliser NTP est de choisir un serveur qui diffusera le temps sur le réseau local. Ce serveur devra se connecter à un serveur de précision supérieure, soit sur le campus même s'il en existe un, soit à un serveur national tel que canon.inria.fr.

Le serveur peut alors diffuser le temps sur le réseau par des broadcasts, solution qui ne cause qu'un trafic très faible sur le réseau.

Les machines qui savent le faire peuvent alors récupérer ces broadcasts et les utiliser pour synchroniser leurs horloges.

Pour d'autres machines, il peut être utile de placer une procédure qui, par exemple au démarrage de la machine, appelle le serveur pour obtenir l'heure et fixer la valeur de l'horloge locale.

Chapitre 16 Partage des comptes : NIS

La participation de Sun

Avec le développement des stations de travail, la nécessité de nouveaux outils pour administrer les machine est apparue. Il n'est en effet pas raisonnable d'avoir à créer des comptes pour les utilisateurs sur chaque machine. En dehors du travail que cela nécessiterait de la part de l'administrateur, cela est inexploitable en pratique, ne serait ce quand un utilisateur souhaite changer son mot de passe !

Le premier fabricant de stations de travail de l'époque, Sun, développa donc une série de protocoles simplifiant l'administration de machines nombreuses :

- un protocole de représentation des données neutre : XDR
- un protocole d'applications réseau permettant à un client d'invoquer des procédures sur un serveur distant : RPC
- un protocole permettant de partager les fichiers de configuration des machines : NIS
- un protocole de partage de disques en réseau : NFS

L'ensemble de ces protocoles a été publié et est utilisé largement par les différents constructeurs, y compris sur les systèmes gratuits tels que Linux.

Applications réparties

Représentation neutre de données : XDR

Le premier protocole, XDR : eXternal Data Representation fournit une manière neutre, c'est à dire fonctionnant sur toutes les machines et les architecture, de représentation des données.

Le protocole permet de décrire des données simples : entiers, flottants de diverses longueurs, chaines de caractères, vecteurs, énumérations, puis, à partir de là, de définir des données plus complexes à l'aide de structures.

C'est un langage de description, dont la syntaxe ressemble beaucoup à la partie description de données du langage C.

XDR est décrit dans le RFC1832.

Appel d'applications à distance : RPC

Ayant une représentation universelle des données, on peut envisager l'appel de procédures (ou fonctions) sur un serveur. Le modèle utilisé est celui du client/serveur : Des programmes appelés serveurs implémentent des fonctions ou procédures qui peuvent être appelées à distance par des clients.

Ce protocole est RPC : Remote Procedure Call. Chaque programme est désigné par un numéro et attend sur un port de son choix. Il peut utiliser aussi bien UDP que TCP pour communiquer. Il implémente une série de fonctions, désignées par un nombre commençant à 0. Il doit au moins implémenter la fonction 0 qui est une fonction de test, sans argument et donnant en réponse juste une absence d'erreur.

Il est à noter que RPC utilise en plus un numéro de version pour chaque programme, ce qui permet de faire cohabiter des versions différentes.

RPC décrit (en XDR, d'ailleurs), le format des paquets utilisés pour appeler une fonction en lui passant des arguments, et pour répondre au client en renvoyant le résultat. Bien sur des champs permettant l'identification du client sont prévus.

La seule difficulté reste de trouver le port sur lequel attend le serveur. Le portmaper résoud ce problème. Il s'agit d'un serveur, qui doit toujours être présent sur une machine utilisant les RPC et qui attend, lui, sur un port connu : le 111.

Son numéro de programme est 100000 et il implémente, en dehors de la procédure 0 obligatoire, 4 procédures :

- PMAPPROC_SET
- PMAPPROC_UNSET
- PMAPPROC_GETPORT
- PMAPPROC_DUMP

La première procédure permet à un programme d'enregistrer le port sur lequel il attends. La seconde permet à ce même programme de se retirer, la troisième permet à un client de savoir sur quel port un programme attend et la dernière sert à obtenir la liste des programmes enregistrés.

Un client voulant contacter un programme contacte donc d'abord le portmaper pour connaître le port sur lequel le serveur attend, puis envoie sur le bon port un paquet contenant le programme à utiliser, la procédure qu'il veut invoquer en les éventuels arguments. Le serveur, après vérification des droits, effectue la requête demandée et renvoie le résultat au client.

RPC est décrit dans le RFC1831.

Deux outils permettent de contrôler le fonctionnement du portmaper :

- pmap_dump qui invoque la fonction PMAPPROC_DUMP du portmaper, donnant ainsi la liste de tous les programmes enregistrés.

- rpc_info qui permet de tester les fonctions du portmaper, et en particulier d'émettre des broadcasts permettant de trouver les différentes machines qui font fonctionner un portmaper sur un réseau.

D'autre part, le programme rpcgen permet, à partir d'une description en XDR d'un programme de générer un squelette de programme C ou il ne reste plus qu'à écrire le programme des fonctions.

Les « maps » NIS

La première utilisation de RPC est le partage des fichiers de configuration système sur un réseau : il s'agit de NIS : Network Information Service aussi connu sous le nom de yellow pages, d'où le nom de la plupart des programmes et qui commencent par yp.

NIS repose sur deux programmes :

- ypserv, de numéro 100004 qui fournit les fichiers voulus
- ypbind de nu,éro 100007 qui permet de s'enregistrer dans un domaine.

NIS est organisé en domaines qui partagent les mêmes fichiers de configuration. La configuration de référence est détenue par une machine dite serveur maître. Des serveurs esclave détiennent des copies des fichiers de la machine maître. Enfin, les clients contactent un serveur (maître ou esclave) pour obtenir les entrées des fichiers de configuration.

Les fichiers de configuration qui peuvent être partagés dans un domaine NIS sont (dans /etc) :

- passwd : la description des comptes utilisateurs
- group : la description des groupes d'utilisateurs
- hosts : la liste locale des noms de machines
- services : la liste des services du réseau et les numéros de port associés
- …

Plusieurs commandes permettent de contrôler le fonctionnement d'un domaine NIS :

- domainname : donne le nom du domaine NIS auquel la machine participe.
- ypwhich : donne le nom du serveur NIS auquel on s'est connecté.
- ypcat : donne le contenu d'un des fichiers partagé (ypcat –x donne la liste de tous les fichiers partagés dans le domaine)
- ypmatch : donne une entrée d'un des fichiers partagés.

Les fichiers de référence sur le serveur sont fabriqué à partir des fichiers de configuration de la machine par une commande make passée dans le répertoire de NIS (/var/yp par exemple). Cette commande doit être passée chaque fois qu'un des fichiers change , par exemple si l'on crée ou modifie un compte.

Pour modifier son mot de passe, un utilisateur doit utiliser une nouvelle commande : yppasswd qui modifie le mot de passe sur les fichiers partagés.

Chapitre 17 Partage des disques : NFS

L'autre grande application conçue par Sun est le système de partage des disques en réseau NFS : Network File System. Grâce à lui, un système de fichier, par exemple les répertoires « home » des utilisateurs peut être vu depuis l'ensemble des machines d'un réseau.

Principes

NFS utilise bien sur les RPCs. Il repose sur deux programmes , l'un permet le montage des systèmes de fichiers, l'autre réalise les fonctions de NFS proprement dit :

- mountd, le programme numéro 100005 qui offre 5 procédures (en plus de la procédure 0) :
  - MNTPROC_MNT : réalise le montage proprement dit en fournissant en réponse un « file handle » qui donne accès au répertoire demandé. Ce « file handle » est un nombre qui doit être fourni pour accéder au répertoire. Il sera par la suite utilisé par toutes les opérations NFS.
  - MNTPROC_DUMP : donne la liste de tous les répertoire qui ont été monté par mountd.
  - MNTPROC-UMNT : démonte un système de fichier.
  - MNTPROC_UMNTALL : démonte tous les système de fichier monté par la machine.
  - MNTPROC_EXPORT : donne la liste de tous les systèmes de fichiers exportables avec la liste des machines pouvant les monter.

- nfs, le programme de numéro 100003 qui offre une série de procédures permettant les opérations sur le système de fichier précédemment monté : lecture, écriture, manipulation des répertoires, etc.… Le nombre de procédures de nfs dépend de la version : 17 pour la version 2, couramment utilisée, et 21 pour la version 3.

Le fonctionnement de NFS est le suivant :

- sur le serveur, les deux programmes mountd et nfs démarrent et s'enregistrent auprès du portmapper.
- sur le client, la commande mount déclenche un appel au portmapper pour trouver le port sur lequel attend montd, puis un appel à la procédure 1 de mountd pour obtenir le « file handle » du répertoire à monter.
- à partir de la, toutes les opérations sur ce système de fichier depuis le client seront traduites en des appels aux procédures du serveur nfs.

Le dialogue entre le client et le serveur se fait par appels des procédures du serveur nfs par rpc. Les données sont échangées dans des paquets UDP. Si le client ne reçoit pas de réponse à un appel, il recommence indéfiniment.

Il est à noter que le serveur nfs ne garde aucune mémoire des opérations (on dit qu'il est « stateless »). Chaque opération est synchrone et se termine soit par un succès et le système de fichier à été modifié conformément, soit par un échec et le système de fichier est dans l 'état où il était avant l'opération.

Par exemple, pour accéder à un fichier, le client va commencer par se déplacer dans l'arborescence des répertoires à partir du file handle que lui à fournit mountd. Pour cela, il fait des appels successifs à la procédure 4 de nfs : « LOOKUP » qui donne en réponse un nouveau file-handle. Ayant le file handle pointant sur le fichier voulu, il peut alors faire appel à la procédure 6 de nfs : « READ » qui retourne une portion du fichier à partir d'un point donné.

Il est important de noter que READ ou WRITE lisent et écrivent à un endroit précis du fichier. Ces opérations peuvent donc être répétées sans causer d'erreur, ce qui est impératif avec UDP : si le client appelle une procédure et que le paquet rpc en réponse est perdu, le client va reémettre sa commande.

Nfs (version 2) et mountd (version 1) sont décrit dans le RFC1094. Nfs (version 3) est décrit dans le RFC1813.

Mise en œuvre

La mise en œuvre de nfs sur une machine Unix est relativement simple :

Sur un serveur, la liste des répertoires à exporter et la liste des machines qui auront le droit de les monter ainsi que la manière de gérer les droits d'accès se trouve en général dans le fichier /etc/export.

Sur un client, les répertoires à monter sont ajouter dans /etc/fstab avec la syntaxe :

machine:/répertoire_à_monter   point_de_montage nfs   paramètres

Des versions commerciales de nfs existent aussi sur PC (PCNFS par exemple), permettant d'utiliser un disque d'une machine Unix comme répertoire commun. En général, on préfère utiliser la stratégie inverse et utiliser samba pour utiliser le disque Unix comme disque partagé Windows.

Chapitre 18 Partage d'imprimantes : lpd

Les imprimantes font parti des ressources coûteuses dans un réseau informatique et leur partage par l'ensemble des machine permet de constituer un service d'impression efficace, préférable à une multiplication d'imprimantes personnelles fort coûteuses en fournitures. Le protocole LPD : Line Printer Daemon permet ce partage.

LPD utilise une connexion TCP sur le port 515. Pour s'assurer que le client est bien celui qu'il prétend être, de manière à pouvoir tenir une comptabilité, le serveur exige que la connexion vienne d'un port privilégié compris entre 721 et 731.

Le client LPD envoie des commandes au serveur et attend une réponse. Les commandes permettent :

- d'envoyer une impression
- d'avoir la liste des impressions dans la file d'attente
- de supprimer une impression

L'exécution d'une impression se fait par la transmission de deux fichiers :

- un fichier de contrôle contenant toutes les informations sur cette impression : machine d'origine, nom de l'utilisateur, titre de l'impression, type du fichier, etc… Il doit avoir un nom de la forme cfAnnn où nnn est un nombre de 3 chiffres.

- le fichier de données proprement dit contenant les données à envoyer à l'imprimante, après passage éventuel dans un filtre suivant le type de fichier, par exemple un fichier texte destiné à une imprimante Postscript sera converti par un utilitaire tel que a2ps. Il doit avoir un nom de la forme dfAnnn.

Sur une machine Unix, les imprimantes, y compris les imprimantes partagées, sont décrites dans le fichier /etc/printcap.

LPD est décrit dans le RFC1179.

Les commandes concernant l'impression sont :

- lpr : envoi d'un fichier en impression
- lpq : interrogation de la file d'attente sur une imprimante
- lprm : suppression d'une impression
- lpc : utilitaire d'administration des imprimantes.

Sur les machines Windows, l'accès aux imprimantes LPD peut se faire à l'aide du logiciel WinLPR distribué comme « contributiel » (shareware).

Chapitre 19 Configuration dynamique des machines

Certaines machines, telles que les terminaux X, les stations sans disque ou certaines imprimantes, doivent charger leur système d'exploitation par le réseau. Ces machines n'ont en général pas de quoi stocker une configuration telle que leur adresse IP. Tout le logiciel qui s'exécute quand une telle machine démarre doit résider dans une ROM est doit donc être très simple.

Deux protocoles ont été définis pour ces machines : bootp pour qu'elles obtiennent leur configuration, tftp pour charger l'image de leur système d'exploitation.

Chargement de logiciel : TFTP

TFTP : Trivial File Transfer Protocol permet donc de lire (ou d'écrire) un fichier à travers le réseau de la manière la plus simple possible. Il utilise pour cela des paquets UDP, bien plus simple à gérer qu'une connexion TCP.

Le protocole ne prévoit aucune authentification. Il ne permet pas non plus d'obtenir la liste des fichiers sur le serveur : les seuls ordres sont l'envoi et la réception de fichiers.

Quatre types de paquets sont prévus :

- demande de lecture ou d'écriture (Read/Write Request)
- paquets de données
- accusé de reception de données
- message d'erreur

Après une demande de lecture/écriture, les données sont envoyées par blocs de 512 octets numérotés. Après l'envoi de chaque bloc, on attend un accusé de réception. S'il tarde, le bloc est ré-émis. Un bloc de moins de 512 octets marque la fin du transfert.

Le serveur attend les demandes de lecture/écriture sur le port 69. Il utilise ensuite un port quelconque. Le client utilise un port de son choix qui identifie le transfert (TID : Transfer ID).

Par exemple, le dialogue entre une machine A, cliente qui demande le fichier test.c (de longueur 1147) à la machine B, serveur serait :

A :1234	>	B :69	RRQ « test.c »
B :2345	>	A :1234	bloc 1 (512octets)
A :1234	>	B :2345	ACK bloc 1
B :2345	>	A :1234	bloc 2 (512 octets)
A :1234	>	B :2345	ACK bloc 2
B :2345	>	A :1234	bloc 3 (123 octets)
A :1234	>	B :2345	ACK bloc 3

TFTP est décrit dans le RFC783, mais cette description contient une erreur connue sous le nom de « syndrome de l'apprenti sorcier »: si à la suite d'une surcharge du réseau les accusés de réception arrivent avec du retard, on arrive à une situation où les blocs de données sont émis 2 fois, puis 4 fois, etc… ce qui conduit inévitablement à rendre le réseau inutilisable. Le RFC1350 donne une nouvelle description de TFTP sans ce syndrome.

La configuration d'un serveur TFTP demande un minimum de précautions : le serveur doit fonctionner dans un environnement où il ne peut accéder qu'à un répertoire bien défini dans lequel on place les fichiers qui peuvent être lus.

Chargement de configuration : BOOTP

Avant de pouvoir charger son système, la machine doit obtenir un minimum sur sa configuration : son adresse IP, l'adresse de son routeur, l'adresse de la machine sur laquelle elle doit lire son système et le nom du fichier à transférer.

Cela est fait à l'aide du protocole BOOTP : Bootstrap Protocol. Comme TFTP, BOOTP utilise des paquets UDP. Le client utilise le port 68, le serveur le port 67. Le fonctionnement est le suivant :

- le client émet une requête BOOTP dans un paquet UDP à destination de l'adresse de broadcast.
- ce paquet est donc lu par toutes les machines du réseau. Parmi celles-ci, un serveur configuré consulte la liste des machines qu'il doit servir pour y rechercher l'adresse Ethernet du client. S'il la trouve, il utilise les informations correspondantes (adresse ip, etc…) et les renvoie au client.
- celui-ci fixe alors son adresse ip et récupère son système par TFTP.

BOOTP est décrit dans le RFC951.

Sur une machine Unix, la configuration du serveur BOOTP se fait en général dans le fichier /etc/bootptab.

Attribution dynamique d'adresses : DHCP

BOOTP offre, dans le paquet émis en réponse par le serveur, un champ optionnel prévu au départ pour y placer des informations spécifiques aux fabricants de matériel. Le protocole DHCP : Dynamic Host Configuration Protocol l'utilise pour une autre application : l'attribution dynamique d'adresse IP.

Les serveurs DHCP disposent de plages d'adresses qu'ils peuvent « louer » aux machines qui se connectent au réseau. Une machine nouvellement connectée émet une requête DHCP et le serveur lui attribue une adresse temporaire.

DHCP est particulièrement utilisé pour configurer les ordinateurs portables des visiteurs et par les fournisseurs d'accès Internet qui ont en général plus d'abonnés que d'adresses disponibles.

DHCP est décrit dans le RFC2131 :

Chapitre 20 Administration réseau

Par principe, la gestion d'un réseau informatique est une tâche complexe à cause de la dispersion géographique du matériel. L'OSI a dans un premier temps décomposé ces fonctions en 5 aires fonctionnelles (ISO 74984):

- la gestion des anomalies
- la gestion des informations comptables
- la gestion des configurations
- la gestion des performances
- la gestion de la sécurité

Le déluge de normes qui suivit aboutit au résultat que l'on sait.

Parallèlement, le monde Internet adopta un protocole d'administration de réseau : SNMP : Simple Network Managment Protocol.

Le protocole définit le dialogue entre une station d'administration et des objets à administrer. Ces dernier communiquent avec la station au moyen d'un agent snmp.

Les objets à administrer (routeurs, machines, etc.…) sont vus au moyen de variables exprimant leur caractéristiques. Ces variables sont classées en groupes :

- system : les caractéristiques de la machine, telles que son nom, le temps depuis lequel elle fonctionne.
- interfaces : les interfaces réseau de la machine, avec leur nom, leur type, leur débit, le nombre d'octets qui y ont transité, le nombre d'erreurs, etc.…
- at : « address translation » autrement dit la table ARP.
- ip : les données relatives au trafic ip de la machine.
- icmp : même chose pour icmp
- tcp : même chose pour tcp.
- udp : même chose pour udp

L'ensemble de ces variable est décrite dans une MIB :Managment Information Base. Le langage utilisé pour décrire ces variables est un sous-ensemble de ASN1. Elles sont nommées à l'aide de désignations ISO sous la forme :

1.3.6.1.2.1.groupe.variable.instance

avec la hiérarchie :

1 :ISO
3 :organisme
6 : DoD
1 :Internet
2 :administration
1 :MIB

Il existe une MIB de base, sensée être présente dans tous les objets administrés, et des extensions permettant de gérer nimporte quel type d'équipement, par exemple X25, port parallèle, Decnet, AppleTalk, …

SNMP définit le dialogue entre une station d'administration et les équipements. Le dialogue consiste essentiellement à consulter la valeur des variables et à les modifier. Les équipement peuvent aussi émettre des notifications asynchrone d'événements particuliers, (traps) tels que la perte d'une liaison.

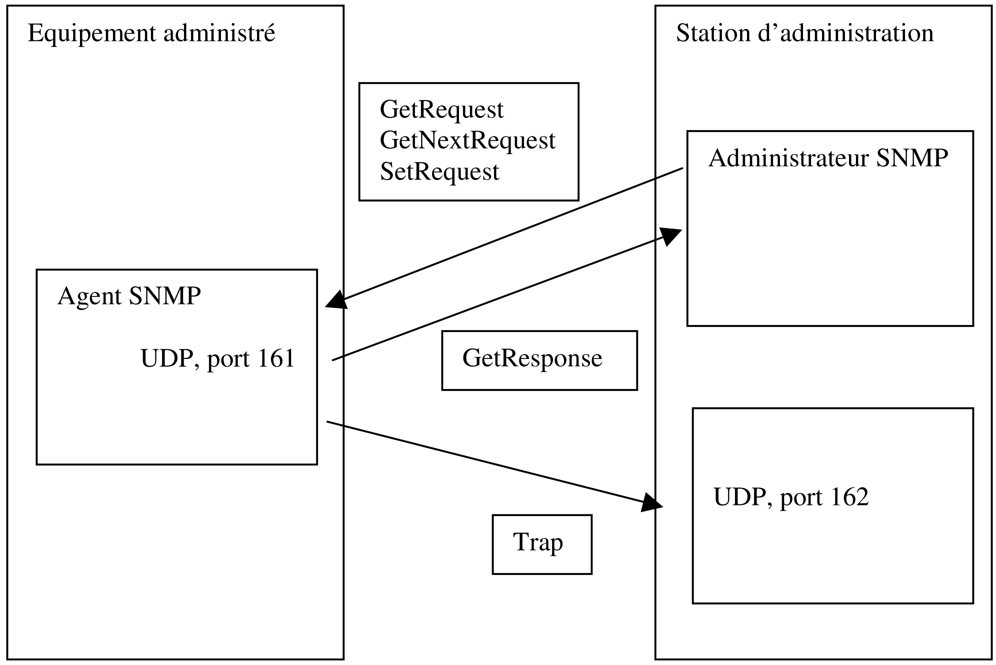

Figure 14 - Principes de SNMP

Les messages SNMP transitent par des paquets UDP. Les messages émis par la station d'administration sont adressés au port 161 de l'agent SNMP, les « traps » émis par l'agent sont à destination du port 16é de la station.

Les commandes sont au nombre de trois :

- GetRequest : demande de la valeur d'une variable
- GetNextRequest : demande de la valeur de la variable suivante pour explorer un tableau
- SetRequest : modification de la valeur d'une variable

Les réponses de l'agent consistent en des messages GetResponse donnant la valeur d'une variable. L'agent peut aussi émettre un « trap » indiquant un événement anormal : redémarrage avec perte de configuration (« cold start »), redémarrage sans perte de configuration (« warm start »), perte d'une liaison (« link down »), rétablissement d'une liaison (« link up »), accès frauduleux (« authentication failure »),…

Malgré son aspect primitif, ce protocole permet d'effectuer toutes les tâches utiles à l'administration d'un réseau : par exemple, bien qu'il n'y ait aucune commande permettant de commander le redémarrage d'un équipement, il suffit de prévoir une variable indiquant l'heure à laquelle l'équipement doit redémarrer et de la modifier en conséquent.

SNMP présente cependant des faiblesses :

- l'identification de la station d'administration se fait par un simple mot de passe, ce qui constitue une sérieuse faille de sécurité.
- il n'est pas adapté à la collecte d'un grand volume d'informations, tel que le journal comptable d'un équipement.
- il suppose un modèle dans lequel il y a une seule station d'administration.

Ces faiblesse ont été corrigées dans la deuxième version de SNMP.

SNMP est décrit dans le RFC1157.

La MIB est décrite dans le RFC1156, puis dans le RFC1213 (MIB II).

SNMP Version 2 est décrit dans la série de RFC1441 à RFC1452.

Chapitre 21 Le monde « Window s »

Microsoft a développé une série de protocoles pour organiser des machines travaillant avec le système d'exploitation Windows en réseau. Le protocole utilisé a été NetBEUI : NetBios Extended User Interface dans un premier temps, puis avec l'émergence d'Internet, la suite de protocoles TCP/IP.

Les fonctionnalités offertes par ce type de réseau concernent avant tout le partage des comptes ; de disques et d'imprimantes. Le développement d'applications suivant le modèle client/serveur est aussi prévu et depuis Windows 2000, un protocole de service de nom (Directory Services) permet un nommage des ressources disponibles offertes sur le réseau.

Hiérarchie : W95/W98/W98SE,W98ME , WNT,W2000

Les variantes du système Windows sont à classer en deux catégories :

- Windows95, puis Windows98 et ses variantes, Second Edition et Millenium sont destinés aux poste « client » utilisant les offres de serveurs.
- Windows NT, puis Windows 2000, systèmes conçus pour une utilisation dans un contexte de serveurs (ces systèmes sont aussi diffusés dans une version dite « WorkStation » destinée à un usage individuel).

Accès aux ressources réseau

Sous Windows, l'accès aux ressources réseau se fait essentiellement à partir de l'icône « voisinage réseau ». On y trouve l'ensemble des machines offrant un service, puis les différents services offerts : disques partagés ou imprimantes. Le contrôle d'accès est fait à partir du mot de passe qui a été demandé au début de la session Windows (pour Windows 95/98, ce mot de passe n'est demandé que pour une machine installée sur un réseau, ou si la machine est configurée pour servir à plusieurs personnes voulant disposer d'un environnement personnalisé).

La configuration (imprimantes partagées, disques partagés) que l'on souhaite offrir sur une machine se configurent à partir des icônes Imprimantes et Partage de fichiers du menu Panneau de Configuration.

Windows et Unix : SAMBA

Dans un réseau local, comportant des machines sous Windows, il peur être avantageux d'installer un serveur Linux fournissant à ces machines l'accès aux imprimantes et à des disques partagés. Le logiciel SAMBA permet ce genre de configuration. Il implémente les protocoles de réseau Windows et permet :

- aux machines Windows de voir les imprimantes gérées par LPD comme des imprimantes partagées.
- aux machines Windows de voir des répertoires de la machine Linux comme des disques partagés.
- aux machines utilisant LPD de voir les imprimantes Windows.
- aux machines utilisant NFS de voir des disques Windows.

Connection d'une machine Windows

En réseau local

La configuration se fait dans le menu Réseau du Panneau de Configuration. Il faut activer le protocole TCP/IP, puis, en fonction de la configuration, choisir une adresse IP (attribuée par l'administrateur du réseau) ou choisir une attribution dynamique d'adresse. De la même manière, il faut définir le nom du domaine et l'adresse du serveur de nom et du routeur (passerelle). Dans le cas ou on choisit une adresse IP fixe, il faut éventuellement indiquer le masque de sous-réseau.

La carte Ethernet par laquelle se fait la connexion est choisie automatiquement s'il n'y en a qu'une, sinon, il faut la définir.

A distance

La configuration PPP se fait par le menu Nouvelle Connexion dans Accès réseau à distance du Panneau de Configuration. Il est possible de créer diverses connexion pour pouvoir accéder à plusieurs fournisseurs (ou appeler différents numéros pour accéder à son laboratoire). Les paramètres demandés sont le numéro de téléphone à composer et le nom sous lequel la connexion s'établit. Le mot de passe est normalement demandé à chaque connexion.

Chapitre 22 Le monde « MacIntosh »

Le système d'exploitation du MacIntosh remonte à 1984. Pourtant, dès le début, il a été doté de fonctions réseau avancées. La première version de ces protocoles, baptisée AppleTalk, utilisait une liaison entre les machine sous forme d'une simple paire de câbles métalliques et fonctionnait à 230000bits/s, avec une variante nommée LocalTalk utilisant des cables téléphoniques. Avec l'émergence d'Ethernet, une variante utilisant ce media : EtherTalk prit la suite.

L'activation d'AppleTalk se fait dans le Sélecteur : on y voit apparaître les imprimantes et les disques partagés. Il suffit de les sélectionner, avec éventuellement un mot de passe, pour y accéder.

Pour offrir ces services aux autres machines, les menus Disques Partagés et Gestion des Utilisateurs sont utilisés.

AppleTalk et Unix :

De la même façon qu'il est possible d'offrir un service de disques et d'imprimantes à des machines Windows depuis Unix avec Samba, NETATALK offre le même service au réseau AppleTalk.

Connection d'une machine MacIntosh

La longue histoire de MacOS a connu diverses manière de relier un MacIntosh à un réseau TCP/IP. La première utilisait MacTCP, avec éventuellement MacPPP pour établir une connexion par modem. Depuis OpenTransport permet une connexion quasi automatique.

En réseau local

Le panneau de configuration MacTCP demande de remplir les éléments indispensables : adresse IP, masque de sous-réseau (éventuellement), adresse du routeur (passerelle), nom du domaine et serveur de nom.

A distance

La configuration de MacPPP demande juste le numéro de téléphone à composer, le nom sous lequel la connexion s'établit et le mot de passe. Il n'y a normalement pas à toucher aux autres paramètres.

Les évolutions de Mac OS vont changer tout cela : MacOS X est à la base un système Unix et se configure d'une toute autre manière.

Appendice : Principaux RFCs

.768	UDP
.783	TFTP
.791	IP
.792	ICMP
.793	TCP
.821	SMTP
.822	En têtes de mail
.826	ARP
.854	TELNET
.855	Options TELNET
.857	Option « echo » TELNET
.858	Option « suppress go ahead » TELNET
.859	Option « status » TELNET
.860	Option « timing mark » TELNET
.894	IP sur Ethernet
.904	EGP
.951	BOOTP
.959	FTP
.977	NNTP
.1013	X Window
.1034	DNS : concepts
.1035	DNS : implémentation
.1042	IP sur 802.3
.1055	SLIP
.1091	Option « terminal type » TELNET
.1094	NFS
.1144	Compression Van Jacobson
.1156	MIB
.1157	SNMP
.1179	LPD
.1213	MIB II
.1282	RLOGIN
.1305	NTP
.1321	MD5
.1332	PPP (IP)
.1334	PAP
.1350	TFTP (corrigé)
.1388	RIP
.1441-1452	SNMP V2
.1519	CIDR
.1597	Adresses IP locales
.1631	NAT
.1661	PPP
.1700	Numéros attribués
.1738	URLs
.1771	BGP
.1813	NFS V3
.1831	RPC V2
.1832	XDR
.1866	HTML
.1869	ESMTP
.1939	POP
.2045-2049	MIME
.2060	IMAP
.2068	HTTP
.2131	DHCP
.2251	LDAP

.2328	OSPF
.2400	Protocoles officiels
.2460	IP V6

Références

- Formation aux réseaux – JoAnne Woodcock – Ed. Microsoft Press

Un livre très simple d'initiation, il vient de chez Microsoft et pourtant il parle aussi bien de Netscape et de Linux !

- Les réseaux – Guy Pujolle –Ed. Eyrolles

Un monument en français. Il parle de tout, vraiment tout. Plutôt un ouvrage de référence que pour apprendre

- Les réseaux – Andrew Tanenbaum

Probablement le meilleur pour apprendre. Ecrit par un expert, l'auteur du système d'exploitation Minix.

- Et dieux créa l'Internet – Christian Huitemas – Ed. Eyrolles

L'auteur était à l'époque (1995) président de l'IAB : il sait de quoi il parle.

- UNGI-http://ungi.nirvanet.net/

Un site Web qui regorge d'information : vous y apprendrez Perl, JavaScript, Java, …

- les RFC

- TCP/IP Illustrated – W. Richard Stevens – Ed. Addison Wesley

L'ouvrage de référence sur TCP/IP : le premier volume décortique les protocole. Le second est une lecture commentée de l'implémentation de TCP/IP dans Unix (NetBSD) et est réservé à un public très averti. Le troisième est un complément du premier, en particulier sur le Web.

- The Open Book – Marshal T. Rose – Ed. Prentice-Hall

Une des farouches opposant au modèle OSI défend son point de vue en réalisant la première (et unique) implémentation du modèle. Très technique

- The Simple Book – Marshall T. Rose – Ed. Prentice-Hall

Tout simplement le meilleur livre sur SNMP

Index

.rhosts,53
.Xauthority,74
100BaseT,23
10Base2,22
10Base5,21
10BaseFL,23
10BaseT,23
802.1,24
802.11,24
802.14,24
802.2,24
802.3,12, 24
802.4,24
802.5,24
802.6,24
A,50
Abstract Syntax Number 1,30
Address,50
Address Resolution Protocol,38
adresses IP,41
adresses locales,46
ADSL,26
AFNIC,49
AFNOR,29
agent snmp,89
AIFF,16
American National Standards Institute,13
American Standard Code for Information Interchange,15
ancre,69
ANSI,13
applets,70
area,46
area border routers,46
ARP,38
ARPA,33
Arpanet,12, 33
AS,42
ASCII,15
ASN1,30, 89
Association Française pour le Nommage Internet en Coopération,49
Asymetric Digital Subscriber Line,26
Atlas 400,31
Atlas400,61
ATM,26
Autonomous System,42
balise,69
base 64,17
Base64,64
baud,17
Bellman-Ford,44
BER,30
BGP,45
Binary Encoding Rules,30
bind,51
binhex,17
Bitnet,61
BOOTP,88
Bootstrap Protocol,88
Border Gateway Protocol,45
bouchon,21
broadcast,22, 42
câble,26
caches,70
Canonical NAME,50
Carrier Detect,17
Carrier Sence Multiple Acces/ Collision Detection,21
CCITT,13
CCITT n° 5,15
CD,17
CDE,75
charte,67
CIDR,44, 45
Cisco,46
Classless Inter Domain Routing,44
Clear To Send,17
client/serveur,79
client-serveur,53
CMIP/CMIS,11
CMIP/CMIS,32
CNAME,50
coaxial,21
Comité Consultatif International des Télégraphes et Téléphones,13
Common Desktop Environment,75
Common Management Information Protocol/ Common Management Information Services,32
Connexion,53
cookies,70
Couche application,31
Couche liaison,30
Couche physique,30
Couche présentation,30
Couche réseau,30
Couche session,30
Couche transport,30
couches,29, 33
CRC,19
CSMA/CD,21
CTS,17
Cyclic Redondancy Check,19
d'accusé de réception,36
Data Set Ready,17
Data Terminal Ready,17
DB15,19
DB25,17
DB9,17
de Van Jacobson,39
décimale pointée,41
Decnet,11
DHCP,88
Dijskra,44
distant vector,44

Distribured Queue Dual Bus,24
DNS,49
DoD,33
Domain Name System,49
domainname,81
DQDB,24
drop cable,21
DSR,17
DTR,17
durée de vie,34
Dynamic Host Configuration Protocol,88
Earn,61
EBCDIC,15
EGP,45
EIGRP,46
ESMTP,64
Ethernet,12, 21
European Accademic Research Network,61
Extended Binary Coded Decimal Interchange Code,15
Extended IGRP,46
Extended SMTP,64
eXternal Data Representation,79
External Gateway Protocol,45
FAI,12
fanions,19
FDDI,24
fenêtre,37
Fiber Distributed Data Interface,24
fibre optique,23
fichier host,49
File Transfer, Access and Management,31
File Transfert Protocol,57
Flag,19
FORM,70
formes de saisie,70
forum,67
fournisseur d'accès internet,12
fragmenter,34
frame,69
FTAM,31
FTP,57
ftp anonyme,58
GIF,16
Gigabit Ethernet,23
GOSIP,11
Graphics Interchange Format,16
groupes,67
Hayes,27
HDLC,19
HELLO,45
High level Data Link Control,19
hors connexion,68
HTML,12, 69
HTTP,12, 70
HyperText Markup Langage,12
HyperText Markup Language.,69
HyperText Transport Protocol,12, 70
IAB,13
IAC,54
IBM,11
IBM,61
ICANN,49
ICMP,37
IDEA,66
IEEE,13
IETF,13
ifconfig,42
IGRP,46
images,16
IMAP,62, 65
indexeurs,71
Institute of Electrical and Electronics Engineers,13
InterGateway Routing Protocol,46
International Organization for Standardization,13
International Telecommunication Union,13
Internet,12
Internet Architecture Board,13
Internet Control Message Protocole,37
Internet Corporation for Assigned Names and Numbers,49
Internet Engineering Task Force,13
Internet Message Access Protocol,65
Internet Protocol,34
Interpret As Command,54
intra-area routers,46
IP,34
IP Next Generation.,43
IP Version 4,43
IP Version 6,43
IPNG,43
ISO,11
ISO,13, 16
ISO-8859,16, 64
ITU,13
Java,70
JavaScript,70
Join Photographic Exchange Group,16
JPEG,16
Kermit,18
kill file,68
LAPB,19
LCP,39
LDAP,31
liaison spécialisée,11
Lighweight Directory Access Protocol,31
Line Printer Daemon,85
Link Access Protocol version B,19
Link Control Protocol,39
link state,44
listes de diffusion,61
listserv,61
local loopback,42
lpc,85
LPD,85
lpq,85
lpr,85
lprm,85

magic cookies,74
mail,61
Mail eXchanger,50
mainframes,11
majordomo,61
make,81
Managment Information Base,89
Manchester,21
mascarade,37
masque de sous-réseau,42
MAU,24
Media Access Unit,24
mél,61
Message ELectronique,61
Message Transfert Agent,61
MIB,89
MIME,64
mini-ordinateurs,12
Minitel,25
MIT,73
modem,25
modéré,67
mountd,83
MP3,16
MPEG,16
MTA,61
multicast,22, 42
Multipurpose Internet Mail Extensions,64
MX,50, 63
Name Server,50
NAT,47
National Science Fondation,12
NCP,39
NETATALK,95
NetBEUI,93
NetBios Extended User Interface,93
netstat,35, 41
Network Address Translator,47
Network Control Protocol,39
Network File System,83
Network Information Service,80
Network Job Entry,61
Network News Transfert Protocol,67
Network Time Protocol,77
Network Virtual Terminal,54
news,67
NFS,83
NIS,80
NJE,61
NNTP,67
NS,50
NSF,12
NSFNET,45
nslookup,51
NTP,45, 77
NVT,54
Open Shortest Path First,46
Open System Interconection,11, 29
OSI,11
OSI,11, 29, 61
OSPF,46
outed,45
PAP,39
parité,18
passerelles,61
Password Authentication Protocol,39
PGP,65
Ping,38
Point to Point Protocol,39
PoinTeR,50
POP,62, 65
port,35
portmaper,80
Post Office Protocol,65
PPP,39
préambule,22
Pretty Good Privacy,65
prise vampire,21
protocole de routage,41
protocole de routage externe,43
protocole de routage interne,42
PTR,50
QuickTime,16
quoted printable,64
redirection,70
Remote Procedure Call,79
répéteurs,21
Request For Comment,12
Request To Send,17
réseau,41
Réseau Numérique à Intégration de Services,26
réseaux locaux,21
Retransmission Time Out,36
RFC,12, 33
RG58,22
RI,17
Ring Indicator,17
Rivest, Shamir, Adleman,54
rlogin,53
RNIS,26
root servers,50
Round TripTime,36
routage,41
route,41
routeur,41
RPC,79
RS232,17
RS422,18
RSA,54
RTO,36
RTS,17
RTT,36
S0,26
samba,84
SAMBA,93
SAO,50
scp,57
SDLC,19
Secure CoPy,57
Secure SHell,54

sendmail,62
séquence,36
Serial Line IP,39
série,15
serveur X,73
Short Path First,44
Simple Mail Transfer Protocol,61
Simple Mail Transfert Protocol,64
Simple Network Managment Protocol,89
SLIP,39
SMTP,61, 64
SNA,11
SNA,15
sniffer,54
SNMP,89
socket,35
sons,16
sous-réseau,42
SPF,44
SSH,54
start,18
Start Of Authority,50
station d'administration,89
stop,18
Synchronous Data Link Control,19
systèmes autonome,42
table de routage,41
tag,69
TCP,36
TCP/IP,12, 33
telnet,54
terminaux lourds,11
tftp,57
TFTP,87
three way handshake,37
Tim Berners-Lee,12
Time To Live,34
Token Ring,24
TOS,34
Traceroute,38
trame,22
tranceiver,21
Transmission Control Protocol,36
Transpac,11, 31
trap,89
treillis,25
Trivial File Transfer Protocol,87
Trivial File Transfert Protocol,57
TTL,34
Type Of Service,34
UA,61
UDP,36
Unicode,16
Uniform Resource Locator,12
Universal Resource Locator,69
Unix to Unix CoPy,61
URL,12, 69
Usenet,67
User Agent,61
User Datagram Protocol,36
uucp,61
uuencode,17
V11,18
V23,25
V24,17
V32,25
V34bis,26
V90,26
vidéo,16
vidéoconférences,42
Virtual Terminal,31
vote,67
VT,31
VT100,18
W3C,13, 69
WAW,16
Web,12, 69
WinLPR,85
Wold Wide Web,12
World Wide Web,69
World Wide Web Consortium,14
WWW,69
X Display Manager Control Protocol,75
X Window,73
X11,73
X121,31
X25,11, 31
X400,31, 61
X500,31
xauth,74
XDMCP,75
XDR,79
xhost,74
xterm,74
X-Windows,55
yellow pages,80
ypcat,81
ypmatch,81
yppasswd,81
ypwhich,81
zone transfert,50