Deuxième partie

Vue d'ensemble sur TCP/IP

Cette seconde partie est, contrairement à la première, indispensable à la compréhension de l'implémentation du sous-système réseau de Linux. Elle regroupe les notions générales liées à cette dernière.

Chapitre 5

L'architecture TCP/IP

Nous avons abordé l'aspect matériel de la mise en place des réseaux au chapitres 2. Voyons maintenant ce qu'il en est de l'aspect logiciel. Nous avons vu au chapitre un qu'il existe deux grands modèles: OSI et TCP/IP. Ces modèles ne font que définir les fonctionnalités mais n'indiquent rien sur la réalisation de celles-ci. Il existe de nombreuses suites de logiciels réseau, mettant en œuvre tout ou partie de ces modèles. On peut citer, parmi les systèmes propriétaires, **XNS** (pour *Xerox Networking Systems*), **SNA** (déjà cité) et **NetBIOS** d'IBM et, parmi les systèmes ouverts, **UUCP** (pour *Unix-to-Unix Copy Protocol*) et l'architecture TCP/IP (souvent également appelée **architecture Internet** car la suite de protocoles TCP/IP et l'interréseau Internet sont étroitement liés).

Il ne faut pas confondre le modèle TCP/IP (qui est un modèle), l'architecture TCP/IP (suite de protocoles) et l'implémentation de TCP/IP sur tel ou tel système (suite de logiciels).

5.1 Vue d'ensemble

5.1.1 Historique

5.1.1.1 Origine : 1974

Lors des premières années d'utilisation du premier réseau, l'ARPANET, de nombreuses – et parfois vigoureuses – discussions portèrent sur la finalité et l'utilité du réseau, ce qui eut pour effet d'affiner et de modifier le logiciel réseau à mesure que les utilisateurs demandaient de nouvelles fonctionnalités.

Les deux fonctionnalités les plus demandées furent la possibilité de transférer des fichiers d'une machine à une autre et la possibilité de se connecter à distance. Les connexions à distance permettaient à un utilisateur de Santa Barbara de se connecter par le biais du réseau à une

machine située à Los Angeles, et d'utiliser cette dernière comme s'il se trouvait devant. Les logiciels et protocoles utilisés alors n'étaient pas en mesure de gérer ces nouvelles fonctionnalités. Il fallut donc continuellement développer, affiner et tester de nouveaux protocoles.

La connexion à distance fut finalement implémentée grâce à un protocole appelé **NCP** (déjà rencontré au chapitre 1) et le transfert de fichiers à distance au moyen du **FTP** (pour *File Transfer Protocol*).

Vers 1973, il devint clair que l'ensemble de protocoles utilisés n'était pas capable de gérer le volume de trafic et les nouvelles fonctionnalités requises par les utilisateurs. On s'attela au développement d'une nouvelle suite de protocoles. L'architecture TCP/IP fut proposée pour la première fois en 1974. L'article [CK–74] publié par Vincent Cerf et Kahn décrivait un système fournissant un protocole d'applications standardisé, qui prenait aussi en compte les besoins des utilisateurs.

Cerf et Kahn proposaient de plus que la nouvelle suite de protocoles soit indépendante du réseau sous-jacent et du matériel informatique. Il s'agissait d'une idée audacieuse dans un monde informatique où le logiciel et le matériel étaient propriétaires, car elles permettaient de faire participer n'importe quelle plate-forme au réseau. La suite de protocoles fut développée et reçut par la suite le nom de TCP/IP.

5.1.1.2 Envolée : 1982

Une série de RFC, dont nous reparlerons tout au long de ce livre, fut proposée en 1981 pour standardiser la version 4 de TCP/IP, plus particulièrement destinée à l'ARPANET. En 1982, TCP/IP a pris la place de NCP comme protocole dominant dans un réseau maintenant étendu, puisqu'il était composé de machines situées un peu partout aux États-Unis. On estime en effet qu'au cours de la première décennie d'existence d'ARPANET, un IMP y était raccordée tous les 20 jours.

TCP/IP devint important lorsque le Département de la Défense commença à inclure les protocoles de TCP/IP dans les standards militaires, qui étaient obligatoires dans de nombreux contrats.

5.1.1.3 L'implémentation BSD : 1983

La définition des protocoles, c'est bien, leur implémentation c'est mieux. L'université de Californie à Berkeley (UCB) reçut au début des années 1980 une subvention de la DARPA pour qu'elle modifie son système d'exploitation UNIX, connu sous le nom de **BSD** (pour *Berkeley System Distribution*), de manière à ce qu'il inclut la prise en charge d'IP. La version 4.2BSD sortit en 1983 avec une implémentation des quatres protocoles TCP, IP, SMTP et ARP.

Cette version de BSD fut mise dans le domaine public. Le succès de 4.2BSD entraîna celui de TCP/IP, lui-même dopé par le développement de l'ARPANET.

La prise en charge d'IP par 4.2BSD était fort bonne mais l'usage en était limité aux seuls petits réseaux locaux. Pour augmenter les capacités de prise en charge d'IP, BSD ajouta des capacités de retransmission, des informations de durée de vie (TTL pour *Time To Live*) et des messages de redirection. D'autres fonctions furent également ajoutées pour fonctionner avec des réseaux plus grands, des interréseaux et des systèmes étendus connectés par des lignes spécialisées. Ceci donna lieu à une version améliorée (qui contenait ce qu'on appelle les **utilitaires de Berkeley**, ou *Berkeley Utilities*) de son système en 1986, sous le nom de 4.3BSD. Une implémentation optimisée de TCP suivit en 1988 (4.3BSD/Tahoe). Pratiquement toutes les implémentation de TCP/IP actuellement disponibles plongent leurs racines dans les versions de Berkeley, même si BSD n'a pas connu de nouvelle version depuis 1993.

5.1.2 Définition des standards

5.1.2.1 Les RFC

Nous avons vu ci-dessus que relier un IMP à l'ARPANET était l'objet d'un protocole. Par contre rien n'était défini pour savoir comment un utilisateur (*hôte* dans le langage des réseaux) se reliait à l'IMP. En 1968 le réseau de l'ARPA commençait à se mettre en place. Plusieurs nœuds étaient opérationnels et cela commençait à se savoir. Au cours de l'été 1968, un petit groupe d'étudiants de second cycle s'est réuni à Santa Barbara; ils venaient des quatre sites hôtes – UCLA, le SRI, l'université de Californie à Santa Barbara et l'université de l'Utah. Ils savaient que le réseau était en préparation, mais n'avaient guère de détails. Il est sorti de cette réunion un corps de jeunes chercheurs qui allaient se consacrer aux communications d'hôte à hôte sur le réseau. Pour activer les choses, ils avaient décidé de se rencontrer régulièrement. Un mois après la formation du groupe, il est devenu évident qu'ils avaient tout intérêt à rassembler des notes sur leurs discussions. L'un d'entre eux, Crocker, s'offrit pour rédiger les premiers comptes rendus. Pour ne pas froisser les concepteurs officiels du réseau et pour éviter de paraître trop catégorique, il appela sa première note *Request for Comments* (ou **RFC**, demande de commentaires), et l'expédia le 7 avril 1969. Intitulée "*le logiciel de l'hôte*", la note [RFC 1] a été distribuée aux autres sites comme allaient l'être les premiers RFC : par la poste, dans une enveloppe en léchant le timbre.

"Demande de commentaires" s'est révélé un choix parfait pour un titre. Cela avait l'air à la fois attentif et sérieux. Et cela a tenu. Le langage du RFC était chaleureux et accueillant. L'idée était de favoriser la coopération, sans pontifier. Le fait que Crocker ait laissé son ego à l'écart lors du premier RFC a donné le ton et inspiré ceux qui ont suivi le mouvement dans les centaines de RFC amicaux et serviables qui sont venus après. Les RFC allaient devenir le principal moyen de libre expression chez les gens du réseau. Il y a maintenant plus de 3 000 RFC.

5.1.2.2 Le NWG

Bientôt, le collectif né au cours de l'été 1968 s'est désigné sous le nom de **Network Working Group** (ou **NWG**, le groupe de travail du réseau). Le défi qu'il fallait relever était de trouver un accord de principe sur les protocoles – comment partager les ressources, comment transférer les données, comment faire fonctionner le système. Cela signifiait écrire des programmes ou, du moins, adopter certaines règles sur la façon de les écrire, des règles qui recueilleraient une large adhésion.

L'accord était la condition *sine qua non*. On était en présence d'une communauté fondée sur l'égalité des compétences. Tout le monde pouvait écrire un code – ou réécrire le code déjà écrit par un autre. Le NWG était une "adhocratie", une aristocratie de spécialistes, une aristocratie égalitaire pour férus de l'informatique.

Devançant la construction du réseau, le NWG a continué à se réunir régulièrement, et des termes neufs comme des inventions nouvelles sont souvent nés d'un commun accord.

5.1.2.3 Le NIC

En 1967, lorsque Taylor et Larry Roberts avaient annoncé au colloque d'Ann Arbor leur projet de réseau en douze sites, Doug Engelbart se trouvait dans l'auditoire. Il dirigeait à l'époque un laboratoire de recherches informatiques au SRI. Ce qui l'intéressait, c'était l'emploi de l'ordinateur pour élargir l'intellect humain. Sous contrat avec l'ARPA, il s'occupait de mettre au point un système appelé **NLS** (pour *oNLine System*, système en ligne). Engelbart considérait le NLS comme l'instrument naturel d'un bureau central de renseignements pour le réseau de l'ARPA. Si on devait partager des ressources, il était important de faire savoir à chacun ce qui

était disponible. À la réunion du Michigan, il offrit de monter le centre d'information du réseau, le *Network Information Center*, lequel allait finir par être connu sous le nom de **NIC** ([HL–96], pp.93–94).

5.1.2.4 Les organismes de contrôle

Lorsque le DARPA fut créé en 1980, un groupe fut formé pour développer un ensemble de standards pour Internet. Ce groupe, appelé **Internet Configuration Control Board** (**ICCB**) fut réorganisé en 1983, date à laquelle il devint l'**Internet Activities Board** (**IAB**), dont la tâche devint de concevoir, de mettre en œuvre et de gérer Internet.

En 1986, l'IAB se déchargea du développement des standards sur l'**Internet Engineering Task Force** (**IETF**) et la recherche à long terme fut confiée à l'**Internet Research Task Force** (**IRTF**). L'IAB se réserve le droit de donner ou non son autorisation à tout ce que peuvent proposer les deux organisations qui dépendent de lui.

La dernière étape de cette saga fut la formation de l'**Internet Society** en 1992, date à laquelle l'IAB devint l'**Internet Architecture Board**. Ce groupe reste responsable des standards existants et à venir, et soumet ses travaux à la direction de l'Internet Society.

Pratiquement dès ses origines, Internet fut défini comme une "collaboration internationale à organisation souple de réseaux autonomes et interconnectés", qui prenait en charge des communications d'hôte à hôte "par le biais de l'adoption volontaire de protocoles et procédures ouverts" définis dans un document technique appelé *Internet Standards*, RFC 1310,2 ([RFC 1310]).

L'IETF continue à affiner les standards de communication sur Internet par le biais de divers groupes de travail, chacun spécialisé dans un aspect du protocole. Certains groupes sont spécialisés dans la gestion des réseaux, d'autres dans la sécurité, les services utilisateurs ou le routage. La plupart du temps, les groupes de l'IETF se forment, créent une recommandation et se dissolvent en moins d'un an.

5.1.3 Vue d'ensemble de l'architecture TCP/IP

5.1.3.1 Noms des PDU

Rappelons que l'architecture TCP/IP utilise quatre couches au lieu des sept du modèle OSI. Plutôt que d'utiliser la terminologie générale des n-PDU du modèle OSI pour désigner les données encapsulées au niveau de chaque couche, TCP/IP utilise les dénominations suivantes :

Application	Message	
Transport	Segment Datagramme	dans le cas connecté dans le cas non connecté
Réseau	Paquet	
Accès	Trame	

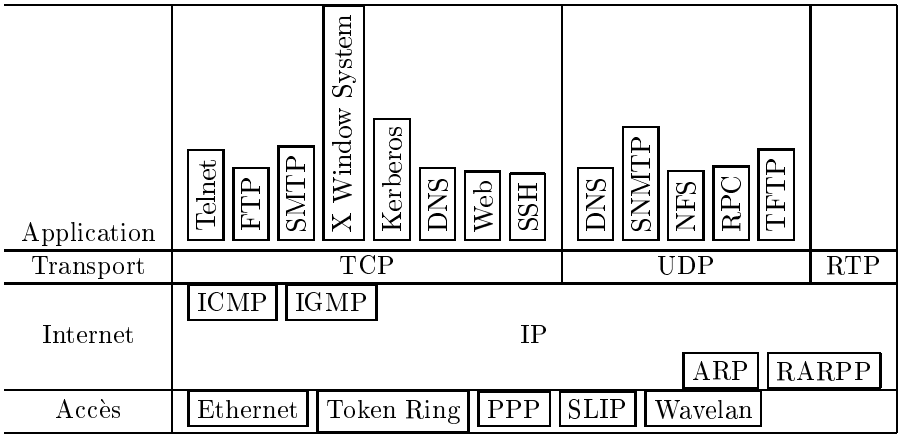
trame se disant *frame* en anglais.

5.1.3.2 Mode d'implémentation

Le traitement d'une couche de protocole peut être implémenté logiciellement ou matériellement ou en utilisant une combinaison des deux. Les couches application et transport sont presque toujours traitées logiciellement. Les couches physique et de liaison sont presque toujours traitées matériellement sur la carte réseau. La couche réseau est souvent implémentée de façon mixte.

5.2 Protocoles de la suite TCP/IP

La figure suivante montre les composants de base de l'architecture TCP/IP :



5.2.1 Protocoles de la couche d'accès

L'architecture TCP/IP proprement dite ne s'occupe ni de la couche physique, ni de la couche liaison. Les cartes réseau et leurs pilotes se trouvent sous la couche Internet. Pour TCP/IP au sens étendu, c'est-à-dire les protocoles définis par des RFC, les protocoles des couches physique et de liaison sont fortement liés : ils forment la couche d'accès. Il s'agit des protocoles **Ethernet** et **Token Ring** pour les réseaux à diffusion, **PPP** (pour *Point-to-Point Protocol*) pour les réseaux point à point et **SLIP** (pour *Serial Line Internet Protocol*) pour les liaisons série (par modem ou par RNIS, numeris en France).

Chacun de ces protocoles possède une partie qui dépend du support, et donc de la couche physique. Par exemple, Ethernet possède des protocoles pour les paires torsadées, pour les câbles coaxiaux, pour les fibres optiques, et ainsi de suite.

Linux implémente également **ATM** (pour*Asynchronous Transfer Mode*), **X.25** et **Frame Relay** pour les réseaux étendus par exemple, même si ces protocoles ne font pas partie de la suite TCP/IP.

5.2.2 Protocoles de la couche réseau

La couche réseau comprend trois protocoles :

- Le protocole principal **IP** (pour *Internet Protocol*) est chargé de déplacer sur les réseaux les paquets assemblés par les protocoles de la couche du dessus. Il utilise un ensemble d'adresses uniques, appelées **adresses IP**, pour chaque composant du réseau afin de déterminer le routage et les destinations.
- Le protocole **ICMP** (pour *Internet Control Message Protocol*) est chargé de vérifier l'état des composants sur un réseau et de générer des messages sur ces états. Il peut être utilisé pour informer d'autres composants du dysfonctionnement d'une machine donnée. Lorsque IP ne peut pas transmettre un paquet à destination, il charge ICMP d'en avertir l'expéditeur et en reste là (la couche supérieure décidera s'il faut renvoyer le paquet ou non). Bien que considéré comme un protocole de la couche réseau, ICMP repose sur IP dans la mesure où ses messages sont envoyés *via* IP. On l'utilise directement à travers le service **ping**.

– La plupart du temps, les protocoles concernent un expéditeur et un destinaire. On parle de **protocole à diffusion unique** (*unicast* en anglais). Quelquefois on veut envoyer des messages à tout un groupe de destinataires. C'est le cas du transfert audio, vidéo, des téléconférences ou des jeux en ligne. On parle alors de **protocole à diffusion restreinte** ou de **protocole de multidiffusion** (*multicast* en anglais).

Le protocole **IGMP** (pour *Internet Group Management Protocol*) est le protocole réseau de multidiffusion de TCP/IP.

En fait IP se décline en plusieurs versions dont trois sont actuellement utilisées :

– la version 4, **IPv4**, avec ses adresses IP sur 32 bits ;
– la version 6, **IPv6** ou **IPng** (pour *IP New Generation*), avec ses adresses IP sur 128 bits ;
– la version sécurisée, **IPsec**, qui n'envoie pas les informations en clair.

Par défaut, IP signifie IPv4.

ARP (pour *Address Resolution Protocol*) est installé entre la couche d'accès et la couche Internet. Ce protocole est mis en place pour traduire les adresses réseau (les adresses IP par exemple) en adresses physiques (les deux couches, réseau et liaison, n'utilisant pas le même format d'adresses).

5.2.3 Protocoles de la couche de transport

La couche de transport comprend deux protocoles fondamentaux :

– **TCP** (pour *Transmission Control Protocol*) est un protocole de communication qui fait au mieux pour permettre un transfert de données fiable. Il se charge d'assembler les données provenant des applications en les plaçant dans des segments standard et en essayant de s'assurer qu'elles sont transférées correctement.
– **UDP** (pour *User Datagram Protocol*) est un protocole qui n'assure pas la retransmission des datagrammes si ceux-ci n'arrivent pas à destination. Le composant émetteur n'a même aucun moyen de savoir si un message a été reçu correctement ou non. UDP est donc moins fiable que TCP mais plus rapide et surtout plus facile à mettre en œuvre. Il n'a pas non plus de fonction de réparation d'erreur.

On dit que UDP est **orienté sans connexion** (*connectionless* en anglais) alors que TCP est **orienté connexion**.

Il existe un troisième protocole qui sert à envoyer les flux radio ou télé à travers Internet. Appelé **RTP** (pour *Real-Time Protocol*), il est décrit dans [RFC 1889]. En fait les datagrammes RTP sont souvent acheminés dans des segments UDP avec RTP implémenté dans l'espace utilisateur et non dans le noyau.

5.2.4 Les protocoles d'application

Les applications s'appuient sur TCP (telnet ou FTP, par exemple), sur UDP (TFTP et NFS, par exemple) ou sur les deux (DNS, par exemple) :

– Les **connexions à distance** (*remote connexion* en anglais) permettent à un utilisateur basé sur un système de se connecter – par l'intermédiaire du réseau – à un autre système l'acceptant comme utilisateur. C'est différent d'une **connexion distribuée**, comme dans le système SAGE, dans lequel le terminal est situé à une certaine distance mais les systèmes sont les mêmes. Le protocole **telnet** fut le premier à permettre la connexion à distance sous TCP/IP. En raison de la transmission sans garantie du mot de passe et des données, on préfère aujourd'hui le protocole **SSH** (*Secure Socket Shell*) à telnet.

- Les **transferts de fichiers** permettent aux utilisateurs de partager des fichiers rapidement et efficacement, sans duplication excessive ni nécessité de se préoccuper de la méthode de transport. Il est bien plus rapide de transférer un fichier par réseau que par la poste, et même que de copier le fichier sur une disquette pour la porter d'une pièce à une autre. **FTP** (pour *File Transport Protocol*) est le premier protocole utilisé sous TCP/IP. Ayant l'inconvénient de transmettre le mot de passe en clair, il est de plus en plus fréquemment remplacé par **SCP** (pour *Secure Copy Protocol*) et **SFTP** (pour *Secure File Transport Protocol*).
- Le **courrier électronique** est bon marché (pas d'enveloppe, ni papier, ni timbre) et rapide (le tour du monde en une minute ou presque). Le premier protocole fut **SMTP** (pour *Simple Mail Transfer Protocol*), totalement transparent pour l'utilisateur. En coulisses, SMTP se connecte à une machine distante et transfère les messages électroniques.
- Au fil du temps, d'autres protocoles s'y sont ajoutés, comme **DNS** (pour *Domain Name System*) qui convertit les noms (tels que `www.linux-france.org`) en adresses IP et *vice-versa*.
- **HTTP** (pour *HyperText Transfer Protocol*) est le protocole le plus fréquemment utilisé actuellement dans la couche application. Il permet l'échange de données dans le **World Wide Web**, c'est-à-dire le chargement de pages web par l'intermédiaire d'un navigateur (Netscape, Mozilla, Lynx, etc.).
- **NFS** (pour *Network File System*) permet à plusieurs ordinateurs d'accéder à un seul et même système de fichiers. Le service NFS représente une extension des systèmes de fichiers locaux au-delà des limites du réseau.

5.3 Les adresses réseau Internet

Dans le cas d'Internet, une adresse réseau est précisée par deux paramètres: l'adresse IP liée à la couche réseau et le port lié à la couche transport.

5.3.1 Adresse IP

5.3.1.1 Définition

IPv4 utilise une adresse 32 bits pour identifier une machine sur l'interréseau, appelée **adresse IP** ou **adresse Internet**.

Les adresses IP sont attribuées par **NIC**. Un réseau qui n'est pas connecté à Internet peut déterminer son propre adressage mais, pour tous les accès Internet, l'adresse doit être enregistrée auprès du NIC.

Le NIC désigne un représentant par pays, et une plage d'adresses, celui-ci étant chargé d'attribuer les adresses dans ce pays, par exemple:

```
http://www.afnic.fr
```

pour la France.

5.3.1.2 Représentation

Puisqu'il est difficile de lire un nombre de 32 bits, on représente souvent ces nombres dans les applications sous la forme de quatre nombres décimaux de 8 bits, séparés par des points, par exemple:

```
127.40.8.72
```

127 représentant dans ce cas l'octet de poids fort.

5.3.2 Multiplexage au niveau transport

5.3.2.1 Notion de port

Un ordinateur peut communiquer en même temps avec plusieurs autres (en utilisant des processus différents et du pseudo-parallélisme), par exemple recevoir une page web, du courrier tout en envoyant un fichier par `ftp`. Les paquets qui arrivent sont triés :

- ceux qui ne correspondent pas à l'adresse IP sont rejetés ;
- les autres sont envoyés vers telle ou telle (instance d') application.

Comment déterminer l'instance d'application vers laquelle on doit envoyer tel paquet ? L'en-tête de la couche de transport (que ce soit TCP ou UDP) contient un champ indiquant celle-ci. Il s'agit d'une entier de 16 bits, donc compris entre 0 et 65 535, appelé **port**.

Sous Internet, une **connexion** est caractérisée, de façon univoque, par un quadruplet : adresse IP source, numéro de port source, adresse IP de destination, numéro de port de destination. Un couple adresse IP, numéro de port s'appelle en anglais un *peer*.

5.3.2.2 Modèle client/serveur

La manipulation des sockets se fait toujours, sauf dans le cas des paires de sockets locales, dans le cadre du modèle client/serveur : la **socket client** est celle qui tente d'établir la communication ; la **socket serveur** est celle qui attend que l'on tente d'établir une communication. Il faut donc prévoir pour la socket serveur un procédé pour cette attente, ne serait-ce qu'une simple boucle.

5.3.2.3 Attribution des numéros de port aux serveurs

Si un client situé sur un ordinateur A veut envoyer une requête à un serveur situé sur un ordinateur B, comment peut-il connaître son port ? Chaque service peut-il choisir son propre numéro ? L'attribution des numéros est-elle centralisée ? Comment une application peut-elle obtenir le numéro de port d'une autre application ?

Un organisme, appelé **IANA** pour *Internet Assigned Number Authority*, attribue des ports fixes aux serveurs des applications connues. On parle des ports bien connus (*well bnown ports*), auxquels chaque application peut se référer lors de l'établissement de la connexion. Au départ, les valeurs situées entre 0 et 256 étaient réservées à l'attribution des ports normalisés. Mais, en 1994, ce quota a été étendu aux valeurs contenues entre 0 et 1 023, en raison de la demande toujours croissante de ports réservés. La liste des ports normalisés peut être consultée dans [RFC 1700] de 1994. La figure 5.1 montre quelques-uns des numéros de ports les plus utilisés par les serveurs.

5.3.2.4 Attribution dynamique pour les clients

Si certains services des numéros fixés par IANA, les clients par contre se voient attribuer un numéro de port de façon dynamique par le sous-système réseau du système d'exploitation au moment où ils ont besoin (situés évidemment situé au-delà de 1 024).

Application	Port
daytime (heure du jour)	13
ftp	21
telnet	23
smtp	25
time (heure)	37
DNS	53
finger	79
web	80
pop3	110
nntp (news)	119
snmp	161
irc (Internet Relay Chat)	194

FIG. 5.1 – *Numéros de port bien connus des serveurs*