Démarrer son réseau IP

Des architectures viennent d'être élaborées, des réseaux viennent d'être construits, des protocoles sont utilisés. Mais comment tout cela s'imbrique-t-il ? Comment IP, TCP et Ethernet fonctionnent-ils ensemble ?

Ce chapitre est l'occasion d'aller plus loin dans la connaissance de ces protocoles et de votre réseau, et de vous donner par la même occasion une vision plus globale des réseaux. Car, comprendre, c'est pouvoir construire des réseaux de plus en plus complexes comme le requièrent les applications multimédias d'aujourd'hui.

Jusqu'à présent, nous ne nous sommes préoccupés que du matériel mais, avec le spanning tree, introduit au chapitre précédent, nous devons désormais nous préoccuper du paramétrage logiciel des équipements réseau.

Comprendre, c'est donc maîtriser le fonctionnement de son réseau.

Deuxième exemple, celui de l'adresse IP que nous avons utilisé sans en bien comprendre les tenants et aboutissants. Cet aspect logiciel doit maintenant être expliqué, car les choix que vous prenez lorsque vous commencez par construire un petit réseau peuvent ensuite peser bien des années plus tard, lorsque celui-ci a pris de l'ampleur.

Comprendre, c'est donc anticiper et faire les bons choix pour l'avenir.

Dans ce chapitre, vous apprendrez ainsi :

- à définir un plan d'adressage IP ;
- à comprendre et à paramétrer le spanning tree ;
- le fonctionnement d'un réseau local ;
- le fonctionnement des protocole IP, TCP et UDP.

Le plan d'adressage IP

À plusieurs reprises déjà, nous avons parlé d'adresses IP sans vraiment nous en préoccuper. Il est vrai que nous n'en avions pas réellement l'usage ; il suffisait simplement de saisir une adresse unique pour chaque station du réseau.

Mais notre réseau prend de l'ampleur et nous devons désormais organiser l'affectation des adresses. Les ISP ne procèdent pas autrement au sein de l'Internet.

Comme cela a été dit au chapitre 4, une adresse IP s'écrit avec quatre numéros, compris entre 1 et 255, séparés par des points, par exemple 192.162.0.1. Une partie de cette adresse désigne un réseau, l'autre le numéro de station au sein de ce réseau. Jusqu'à présent, nous nous sommes arrangés pour configurer toutes nos stations dans le même réseau IP.

On peut se poser la question suivante : pourquoi faut-il des adresses IP alors qu'il existe déjà des adresses MAC ? D'abord, Ethernet est un réseau local, qui n'a donc qu'une portée géographique limitée. Ensuite, il existe des dizaines de réseaux de niveau 1 et 2 différents avec chacun un adressage physique qui lui est propre. Or, les PC, même situés sur des réseaux différents, doivent pouvoir communiquer ensemble. Il faut donc un protocole de niveau supérieur, dit de niveau 3 (couche réseau), qui permet de fédérer ces réseaux avec un adressage unique. On trouve ainsi IP sur Ethernet et PPP, mais aussi sur Token-Ring, ATM, etc.

POURQUOI UN PLAN D'ADRESSAGE ?

L'objectif premier du plan d'adressage est d'éviter la **duplication** accidentelle des adresses. Pour l'adressage MAC, un plan n'est pas utile car les adresses sont affectées aux cartes par les constructeurs. En revanche, l'affectation des adresses IP relève de votre responsabilité, ou de celle du NIC pour le réseau public Internet.

Le plan d'adressage permet également de **contrôler** le fonctionnement de votre réseau IP. En effet, l'affectation des adresses IP doit répondre à des règles précises sous peine d'aboutir à des dysfonctionnements (connexions impossibles, voire intermittentes, etc.).

En définitive, le plan d'adressage permet **d'organiser l'exploitation** de votre intranet.

IP permet aussi de partitionner les réseaux. En effet, de nombreux protocoles utilisent abondamment les broadcasts et multicasts, et il est préférable de limiter la diffusion de ces types de trames. Si votre intranet est connecté à l'Internet, il n'est pas envisageable de recevoir des trames multicast et broadcast émises par un employé de la société X.

De plus, l'interconnexion des sites coûte cher compte tenu des distances. Il est donc judicieux de limiter le trafic afin de ne pas surcharger inutilement les liaisons par des broadcasts.

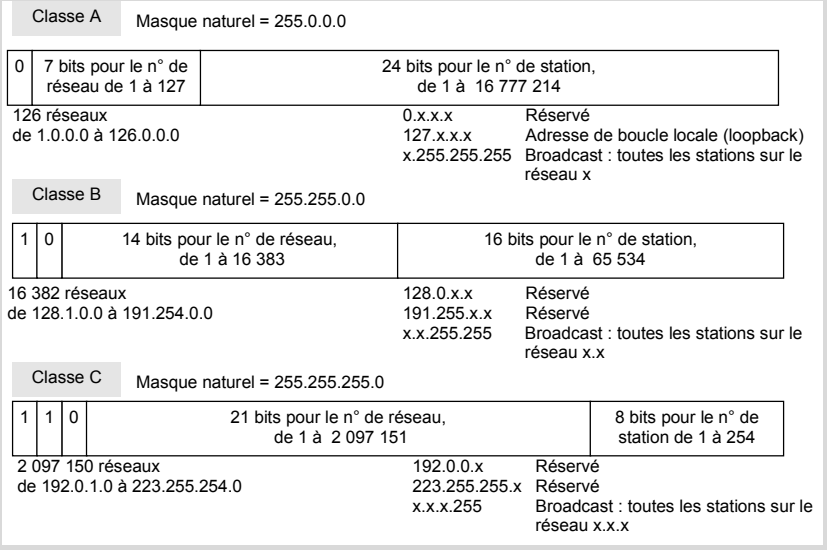
La démarche

Tout d'abord, il est conseillé de retenir un adressage privé, c'est-à-dire complètement séparé de celui de l'Internet, ceci pour des questions de simplicité et de sécurité. Il est toujours possible d'opter pour un adressage publique, mais l'obtention de telles adresses est très difficile car il faut justifier de leur usage auprès des organismes de régulation de l'Internet.

L'ADRESSAGE IP (RFC 791)

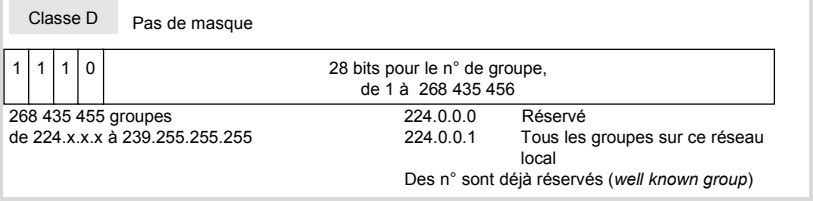
IP (*Internet Protocol*) définit **un réseau virtuel** reposant sur des réseaux physiques de différente nature (Ethernet et PPP, par exemple). Pour ce faire, IP utilise un **adressage logique** différent de l'adressage physique (MAC, PPP ou autre).

Une adresse IP est découpée en un numéro de réseau et un numéro de station au sein de ce réseau. Il existe trois **classes d'adresses** unicast en fonction de la taille du réseau (c'est-à-dire du nombre de stations par réseau). Pour différencier la partie réseau (*subnet*) de la partie station (*host*), IP utilise un **masque** dont tous les bits à 1 représentent la partie réseau.



Deux valeurs sont réservées dans la partie station de l'adresse : 0 pour désigner le réseau lui-même et 255 (tous les bits à 1) pour désigner toutes les stations au sein de ce réseau (broadcast).

Il existe également une classe d'adresses multicast permettant de désigner des groupes de stations.



La classe E (premiers à bits 11110) définit une classe d'adresses expérimentales. Elle n'est jamais utilisée.

L'adresse 255.255.255.255 désigne toutes les stations sur le réseau de l'émetteur du paquet (broadcast IP).

Il se peut donc que vous utilisiez des adresses déjà affectées sur l'Internet, mais cela n'a pas d'importance car votre intranet est isolé. Cela ne vous empêchera cependant pas de l'interconnecter avec l'Internet.

La seconde décision concerne le choix de la classe d'adresse IP. Ce choix dépend du nombre de stations présentes sur votre réseau. Si ce nombre dépasse 254, une classe B s'impose. Une classe A n'est pas utile, car une classe B offre 65 534 adresses de stations, ce qui est largement suffisant. De plus, une classe A est limitée à 126 réseaux IP, ce qui, pour les grands réseaux, peut être un handicap.

En résumé, notre choix s'est provisoirement porté sur un plan d'adressage privé de classe B, ce qui nous donne 16 382 réseaux possibles contenant chacun 65 534 stations. Aux sections suivantes, d'autres considérations viendront modifier ce choix.

Les principes de base

L'adressage IP est très souple et permet de faire tout ce que l'on veut. Afin d'éviter toute mauvaise surprise, il est conseillé de suivre les principes suivants :

- Règle 1 : un réseau IP ne doit pas chevaucher plusieurs sites.
- Règle 2 : il peut y avoir plusieurs réseaux IP sur un site.
- Règle 3 : s'il y a plusieurs réseaux IP sur un site, choisir des numéros contigus. Cela simplifiera le routage.
- Règle 4 : limiter le nombre de réseaux IP. Cela simplifiera les connexions à l'Internet.

Le protocole IP impose qu'une station se trouvant dans un réseau IP ne puisse pas communiquer directement avec une station se trouvant dans un autre réseau IP, même si elles sont connectées au même segment Ethernet. Les réseaux sont segmentés de manière logique ; en d'autres termes, ils sont partitionnés.

La solution repose sur l'utilisation d'un **routeur** dont le rôle est d'interconnecter les réseaux IP, quelle que soit leur localisation géographique.

On verra au chapitre 11 qu'il existe un moyen de lever cette contrainte imposée par IP.

De toute façon, l'utilisation d'un routeur s'impose dès que vous devez relier deux sites sur de longues distances. L'Internet comporte des dizaines de milliers de routeurs. Donc, autant prendre en compte cette contrainte dès le début de l'élaboration du plan d'adressage.

QU'EST-CE QU'UN ROUTEUR ?

Un routeur est un commutateur de niveau 3, c'est-à-dire qui commute les protocoles de la couche réseau, tels que IP. La commutation des paquets IP est plus complexe que celle des trames Ethernet. On emploiera donc plutôt le terme de **routage**.

Ce mécanisme consiste à analyser l'adresse de destination du paquet IP et à le transmettre sur le bon port (appelé **interface**). Il utilise pour cela des algorithmes de routage, tels que **OSPF** (*Open Shortest Path First*) qui permettent de calculer les meilleures routes en fonction des numéros de réseau IP.

Comme pour les PC, une interface routeur est associée à au moins un réseau IP.

Impact sur l'Internet

Tôt ou tard, l'interconnexion de votre réseau avec l'Internet sera nécessaire. Comment éviter que vos adresses internes entrent en conflit avec celles de l'Internet ?

La solution repose sur l'utilisation de la **translation d'adresses**. Cette technique permet de masquer votre plan d'adressage privé vis-à-vis des utilisateurs situés sur l'Internet.

Une solution complémentaire à la première repose sur le non-routage de certaines adresses. La RFC 1918 précise que certaines adresses ont été réservées pour l'adressage privé. Le respect par tous les ISP de cette RFC garantit que ces adresses ne seront jamais routées sur l'Internet.

Réseaux réservés (RFC 1918)	Espace d'adressage
10.0.0.0	1 réseau de classe A
De 172.16.0.0 à 172.31.0.0	16 réseaux de classe B
De 192.168.0.0 à 192.168.255.0	256 réseaux de classe C

On peut donc utiliser ces adresses pour notre réseau privé, sans que cela soit pour autant une obligation. L'essentiel d'une interconnexion avec l'Internet repose, en effet, sur la translation d'adresses.

Or, pour les grands réseaux, le nombre de réseaux IP à translater est source de complexité : s'il y a quarante sites mais un seul point de sortie vers l'Internet, le firewall devra prendre en compte quarante réseaux IP dans ses règles de translation d'adresses.

Afin de simplifier cette configuration, il faudrait donc pouvoir ne translater qu'un réseau IP au niveau du firewall (respect de la règle 4) tout en ayant autant de subnets IP que nécessaire pour notre intranet. La solution repose sur la création de sous-réseaux IP.

Les sous-réseaux IP

Le principe des sous-réseaux (*subnet*) consiste à étendre le nombre de bits désignant la partie réseau. Le nombre de stations par sous-réseau diminue donc d'autant.

Classe	Masque naturel	Nombre de bits affectés au numéro de réseau	Extension possible : nombre de bits affectés au sous-réseau
A	255.0.0.0	8	+ 1 à + 22 bits
B	255.255.0.0	16	+ 1 à + 14 bits
C	255.255.255.0	24	+ 1 à + 6 bits

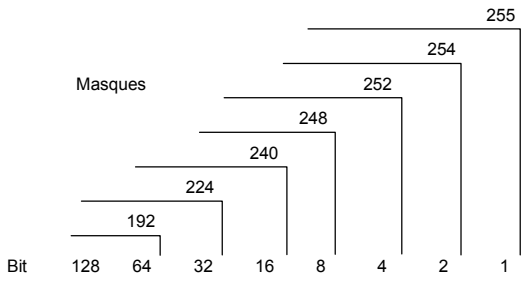
La partie station de l'adresse doit comporter au moins 2 bits afin que cette dernière soit valide.

	Numéro de station	Conclusion
1 bit	1 = broadcast 0 = ce réseau	**Interdit**. Il ne reste aucun bit pour désigner un sous-réseau ou une station.
2 bits	11 = broadcast 00 = ce réseau 01 = station n° 1 10 = station n° 2	**OK**. 2 est le nombre minimal de bits devant être réservés aux sous-réseaux et stations.

La notation décimale (octet par octet) est rendue difficile lorsque le sous-réseau ne porte par sur un multiple de 8 bits. C'est pourquoi la notation « / [*nombre de bits affectés à la partie réseau*] » est plus souvent utilisée.

Numéro de réseau / nombre de bits réservés à la partie réseau	Masque	Commentaire
10.0.0.0 / 10 subnet de +2 bits	255.192.0.0	Permet de créer 4 sous-réseaux, de 10.0 à 10.3
10.0.0.0 / 16 subnet de +8 bits	255.255.0.0	On dit que la classe A est « subnettée » sur une classe B
194.50.0.0 / 19 subnet de +3 bits	255.255.224.0	Permet de créer 8 sous-réseaux
194.50.0.0 / 24 subnet de +8 bits	255.255.255.0	On dit que la classe B est « subnettée » sur une classe C

Figure 7-1.
Les masques de sous-réseaux.



Notre choix initial portait sur une classe B. Si nous voulons limiter le nombre de réseaux IP et conserver la même souplesse que la classe B, il faut donc retenir une classe A « subnettée » sur une classe B.

Cela nous offrirait 256 sous-réseaux. Si, dans le futur, ce chiffre était dépassé, on pourrait toujours ajouter un autre réseau de classe A (il ne ferait pas partie de la RFC 1918, mais ce n'est pas réellement important) et le « subnetter », ou ajouter une classe B à notre plan d'adressage. Notre but est simplement de limiter le nombre de réseaux IP.

Nous choisissons donc l'adresse de classe A, 10.0.0.0, issue de la RFC 1918. Étant donné le subnet choisi, notre masque sera donc : 255.255.0.0. Mais ce choix est encore provisoire.

Méthode d'affectation des réseaux LAN

Le plus simple est d'affecter les réseaux par site (respect de la règle 1). Au lieu d'affecter séquentiellement le numéro, on peut l'incrémenter de 4 ou 8, ce qui laisse la possibilité d'étendre le subnet affecté au site (respect de la règle 2). L'ajout d'un réseau sur un site se traduira donc par l'affectation du numéro de réseau suivant (respect de la règle 3).

Une première version de notre plan d'adressage serait donc la suivante :

Réseau	Site
10.0.0.0/16	Paris : 1 réseau de 65 534 stations
De 10.1.0.0/16 à 10.3.0.0/16	Non affecté (réservé aux extensions de Paris)
10.4.0.0/16	Toulouse : 1 réseau de 65 534 stations
De 10.5.0.0/16 à 10.7.0.0/16	Non affecté (réservé aux extensions de Toulouse)
Etc.	
De 10.248.0.0 à 10.255.0.0	Réseaux non affectés

L'incrément de 4 a été soigneusement choisi, de manière à obtenir des réseaux contigus. Ainsi, le site de Paris dispose de quatre réseaux : 10.0.0.0, 10.1.0.0, 10.2.0.0 et 10.3.0.0, avec chacun un masque à 255.255.0.0. Mais cette manière de découper les réseaux est quelque peu rigide, car la région de Paris peut comprendre à la fois des petits sites et des gros sites.

Une autre façon de voir les choses est de considérer le réseau 10.0.0.0 avec le masque 255.252.0.0 (soit 10.0.0.0/14), ce qui offre 262 142 adresses (65 536 × 4 –2) pour le subnet 10.0.0.0 affecté à Paris (de 10.0.0.0 à 10.3.255.255).

Figure 7-2.
Création d'un subnet.

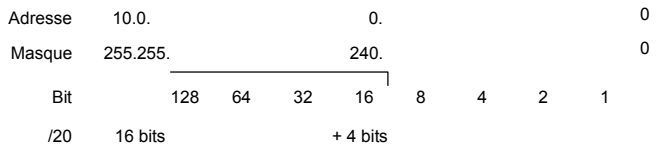
En définitive, notre plan d'adressage se présente en réalité sous la forme suivante :

Subnets du réseau 10.0.0.0/8	Site
10.0.0.0/14 255.252.0.0	Région parisienne
10.4.0.0/14 255.252.0.0	Région toulousaine
10.8.0.0/14 255.252.0.0	Strasbourg
Etc.	
De 10.248.0.0 à 10.255.252.0	Réseaux non affectés

Au sein de ce réseau, il est alors possible de créer d'autres subnets dont la taille varie en fonction de l'importance du site. En faisant varier la longueur du masque on crée ainsi des **subnets variables** (RFC 1219).

Par exemple, au sein de la plage d'adresses affectée à la région parisienne, on peut réserver le subnet suivant à un site de moyenne importance : 10.0.0.0/20 (masque égal à 255.255.240.0), soit 4 094 adresses (16 × 256 –2), de 10.0.0.1 à 10.15.255.254.

Figure 7-3.
Création d'un deuxième subnet.



Au sein de ce site, il peut ensuite être nécessaire de créer des réseaux de différentes tailles, par exemple un réseau principal et de nombreux petits sous-réseaux dédiés connectés, par exemple, à un firewall.

Subnets du réseau 10.0.0.0/14	Fonction
10.0.0.0/22 255.255.252.0	Réseau principal (1 022 adresses)
10.0.4.0/22 255.255.252.0	Réservé à l'extension du réseau principal (*) ou à la création d'un deuxième réseau
10.0.8.0/24 255.255.255.0	Réseaux dédiés au firewall (254 adresses)

(*) Si le réseau principal est étendu, il suffit de changer le masque qui devient 255.255.248.0, ce qui donne le réseau 10.0.0.0/21.

Le réseau 10.0.10.0/23 peut également être découpé en deux subnets de classe C (masque de 24 bits) 10.0.10.0 et 10.0.11.0.

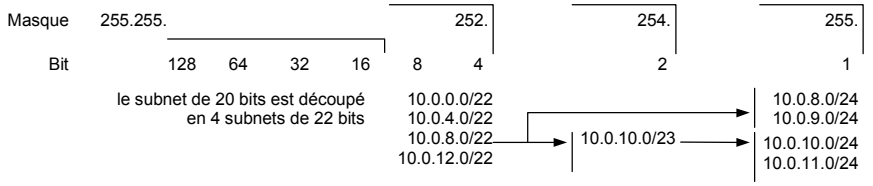


Figure 7-4
Extension des subnets.

Les subnets de classe C ainsi créés (10.0.8.0, 10.0.9.0, etc.) peuvent à leur tour être découpés en de très petits réseaux, juste assez grands pour connecter un routeur et quelques machines.

Subnets du réseau 10.0.8.0/24	Fonction du réseau dédié
10.0.8.0/27 255.255.255.224	Serveurs publics (30 adresses)
10.0.8.32/27 255.255.255.224	Réservé (30 adresses)
10.0.8.64/26 255.255.255.192	Accès distants (62 adresses)
10.0.8.128/28 255.255.255.240	Accès externes (14 adresses)
10.0.8.144/28 255.255.255.240	Réservé (14 adresses)
10.0.8.160/27 255.255.255.224	PABX (30 adresses)
10.0.8.192/26 255.255.255.192	Réservé (62 adresses)

Une autre manière d'appréhender la subtilité du subnetting qui vient d'être opéré est de considérer la grille de découpage suivante.

	2 subnets de 128 (– 2) adresses	4 subnets de 64 (– 2) adresses	8 subnets de 32 (– 2) adresses	16 subnets de 16 (– 2) adresses
Plage d'adresses au sein du subnet	0 – 127	0 – 63	0 – 31	0 – 15
				16 – 31
			32 – 63	32 – 47
				48 – 63
		64 – 127	64 – 95	64 – 79
				80 – 95
			96 – 127	96 – 111
				112 – 127
	128 – 255	128 – 192	126 – 159	128 – 143
				144 – 159
			160 – 191	160 – 175
				176 – 191
		192 – 255	192 – 223	192 – 207
				208 – 223
			224 – 240	224 – 239
				240 – 255
Masque	/25	/26	/27	/28

Les plages réservées permettront d'étendre les plages déjà affectées si le nombre de stations devient plus important que prévu. Ainsi, le réseau " Serveurs publics " pourra être étendu en diminuant le masque de 1 bit, afin de donner le subnet 10.0.8.0/26 (255.255.255.192).

Il est à noter que la création d'un sous-réseau fait perdre chaque fois deux adresses.

La technique du *subnetting* permet de gérer la pénurie d'adresses publiques sur l'Internet. En effet, la création de réseaux IP taillés sur mesure évite le gaspillage d'adresses ; par exemple, le réseau 10.0.0.0/16 offre 65 534 adresses qui seront loin d'être toutes utilisées. Sur votre réseau privé, vous avez cependant plus de latitude. Mais attention aux évolutions qui peuvent être rapides, par exemple lors de la fusion de deux sociétés.

Méthode d'affectation des réseaux WAN

L'interconnexion des réseaux (abordée aux chapitres suivants) nécessite également des adresses, mais en moins grand nombre que pour les réseaux LAN.

Par exemple, sur une liaison point à point, seules deux adresses sont nécessaires, une pour chaque extrémité. Le subnetting sur 30 bits, qui offre deux adresses, permet de créer un réseau juste dimensionné pour ce besoin.

Nous pourrions utiliser une des plages de notre réseau 10, mais il est cependant plus intéressant d'utiliser un autre réseau IP, et cela pour plusieurs raisons :

- Les adresses des réseaux WAN ne sont pas diffusées sur l'ensemble du réseau ; elles ne sont connues qu'entre routeurs adjacents.
- Les adresses n'ont donc pas besoin d'être connues des réseaux utilisateurs.
- Utiliser une plage d'adresses distincte permet de mieux identifier les liaisons WAN.

Bien que cela ne soit pas une obligation, nous préférons donc utiliser une autre plage d'adresses de la RFC 1918. Une classe B suffira amplement.

Nous pouvons donc réserver une plage de notre réseau 172.16.0.0/16, que nous « subnetterons » comme suit :

Subnets de 172.16.0.0/16	Fonction
172.16.0.0/30 255.255.255.252	Liaison Paris-Toulouse
172.16.0.4/30 255.255.255.252	Liaison Paris-Strasbourg
etc.	En tout : 16 384 subnets de 2 adresses

Pour les interconnexions multipoints, il suffira de réduire le masque d'autant de bits que nécessaire pour les subnets considérés. En général, les réseaux multipoints WAN sont rares et comprennent peu d'adresses en comparaison des LAN.

Méthode d'affectation des stations au sein des réseaux

Chaque nœud IP doit posséder une adresse IP. Cela concerne les PC et les Macintosh, les serveurs (NT, Unix, etc.), les imprimantes, les routeurs, les concentrateurs et commutateurs administrables (pour les agents SNMP), etc.

Il est tentant de découper la plage d'adresses en autant de parties qu'il y a de types de matériels. Cela n'apporterait cependant rien ni sur un plan technique, ni sur un plan organisationnel.

L'expérience montre, de plus, qu'une telle pratique n'est par pérenne : soit la taille de la plage que l'on avait réservée est insuffisante (davantage de PC que prévu, par exemple), soit, à la longue, personne ne respecte une marche à suivre qui est trop contraignante (par exemple, s'il faut installer un PC en urgence, on prend la première adresse disponible).

Il est, en revanche, intéressant de prévoir un découpage simple entre les équipements terminaux (PC, serveurs, imprimantes, etc.) et les équipements réseau (les routeurs, les agents SNMP des concentrateurs et des commutateurs, etc.). Cela permet de mieux contrôler les flux du réseau. Dans le cas d'un subnet de classe B, on peut se risquer à créer une troisième plage réservée aux serveurs.

Plage d'adresses	Affectation
De 0.1 à 24.255	Équipements réseau (routeurs, hubs, switches, etc.). 6 399 adresses (de 1 à 6 399)
De 25.0 à 49.255	Serveurs NT, Unix, etc. 6 400 adresses (de 6 400 à 12 799)
De 50.0 à 255.254	Postes de travail (PC, etc.) 52 735 adresses (de 12 800 à 65 534)

Dans le cas d'un subnet sur une classe C, le plus simple est de ne pas affecter de plage d'adresses par type d'équipement, car la probabilité de collision est encore plus forte qu'avec une classe B. L'affectation des adresses pour les équipements réseau et serveur peut commencer par le bas de la plage et s'incrémenter ensuite, tandis que celle pour les PC peut commencer par le haut de la plage et se décrémenter ensuite.

Plage d'adresses	Affectation
De .1 à .254	Équipements réseau (routeurs, hubs, switches, etc.) et serveurs NT, Unix, etc.
De .254 à .1	Stations de travail (PC, etc.)

On peut constater que le plan d'adressage doit prendre en compte de nombreux paramètres liés à des notions qui n'ont pas été introduites : routage, translation d'adresse, affectation dynamique, connexion à l'Internet et contrôle de flux. Les chapitres suivants vous permettront de juger de la pertinence ou non du plan d'adressage qui vous est proposé.

L'encapsulation des protocoles

Il existe différents supports de transmission (câbles en cuivre ou en fibre optique, faisceaux hertziens) et différents moyens d'accéder à ces supports (accès partagé par détection de collision, par jeton, par partage fixe de bande passante, etc.). Cela implique l'utilisation de nombreux protocoles de niveau 1 (couche physique) adaptés à chaque situation.

La couche liaison, telle que PPP, permet de masquer aux couches supérieures les particularités du niveau physique et ses contraintes. Mais il arrive qu'une norme spécifie les couches 1 et 2 : c'est le cas d'Ethernet et d'ATM (*Asynchronous Transfert Mode*).

La couche de niveau 3 (couche réseau), telle que IP, peut donc utiliser différents réseaux en recourant aux services de PPP ou en s'adaptant directement sur une autre couche liaison.

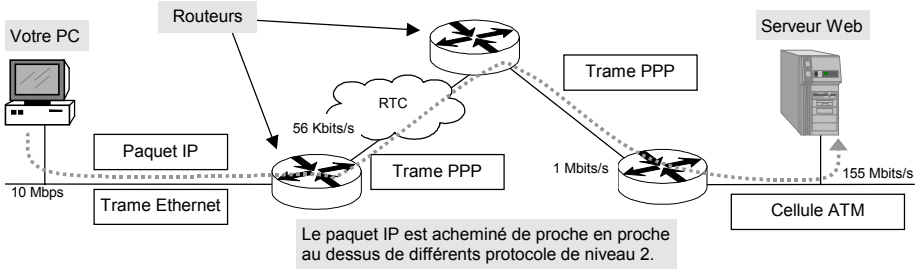


Figure 7-5.
Paquet IP au-dessus de différents réseaux.

On peut établir l'analogie suivante : le paquet IP est une voiture ; les pneus et les suspensions sont les protocoles de niveau 2 qui réalisent l'adaptation aux routes que sont les réseaux physiques. Vous roulez ainsi sur un chemin de terre (le RTC), puis sur une nationale (Ethernet) et enfin sur une autoroute (ATM), mais toujours avec la même voiture. Éventuellement, vous changez de pneus ou de suspensions, afin de vous adapter au terrain. De même, le paquet IP peut emprunter le RTC (avec une trame PPP), un réseau Ethernet (avec une trame Ethernet) ou un réseau ATM (avec une cellule ATM).

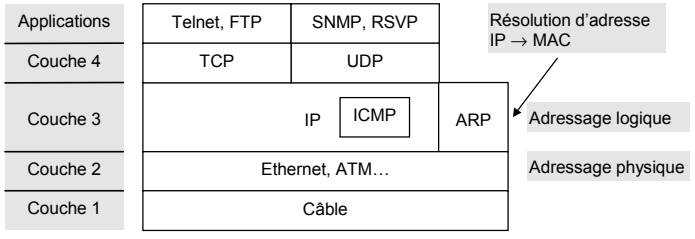


Figure 7-6.
Modèle en couches des protocoles Internet.

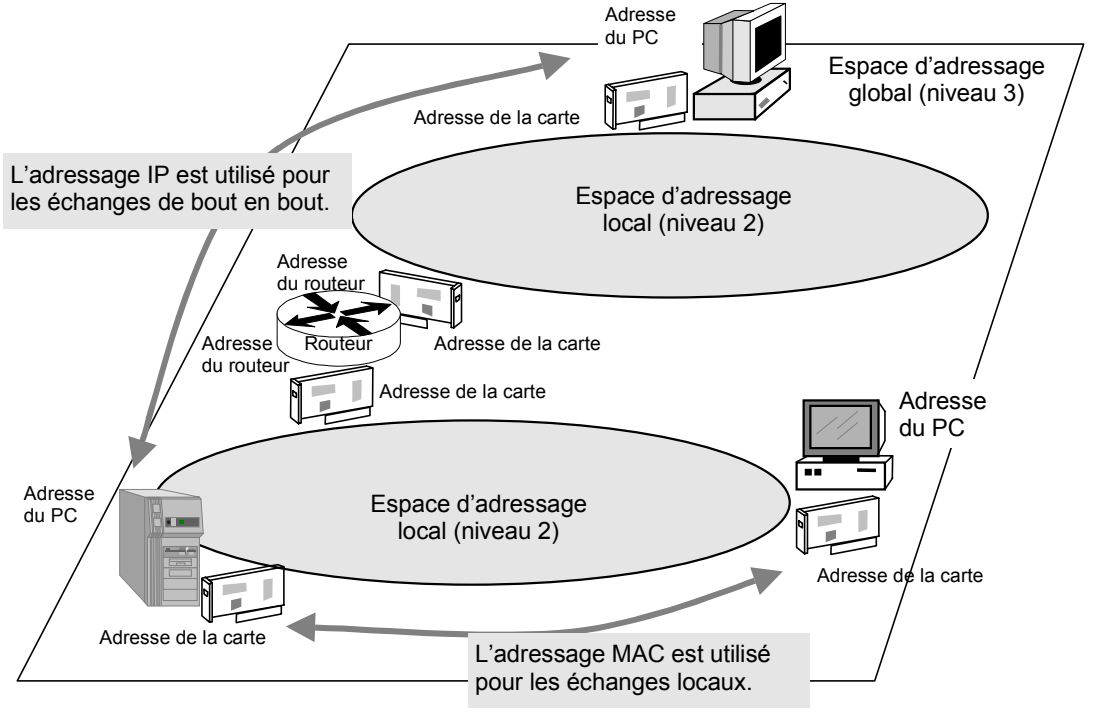
L'adressage

Toutes les couches réseau, de la couche physique à l'application en passant par les couches liaison, réseau et transport, utilisent des adresses afin d'identifier l'émetteur et le destinataire. Chaque couche utilise un système d'adressage spécifique qui répond à un besoin précis.

L'adressage de niveau 2 est géographiquement limité à un réseau local ou à une liaison point à point d'un réseau étendu.

L'adressage de la couche 3 permet d'identifier les stations à un niveau supérieur. Il assure la continuité entre des réseaux physiques qui utilisent différents systèmes d'adressage.

Figure 7-7.
Le rôle de l'adressage.

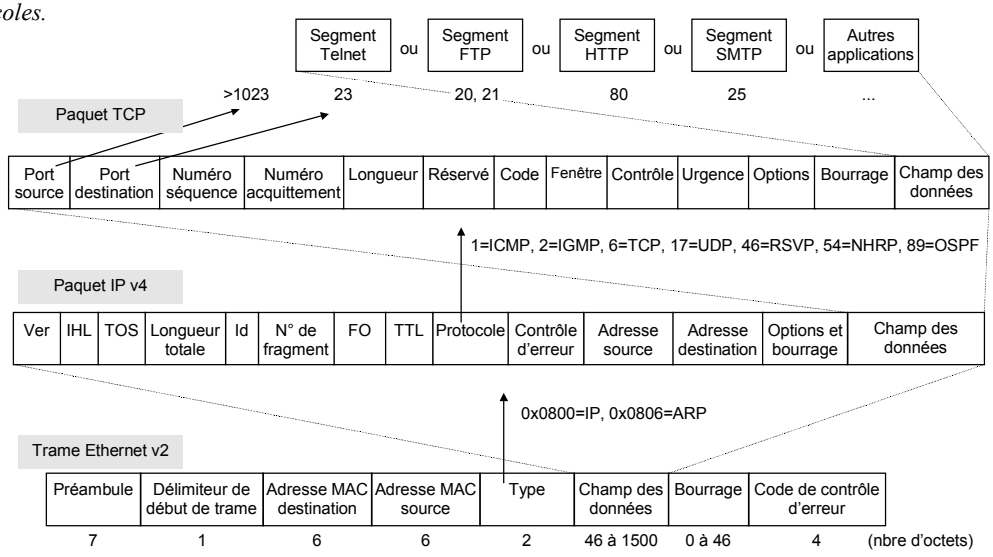


Enfin l'adressage de niveau 4 permet d'identifier les applications qu'utilisent les services de la couche transport.

Le multiplexage

Chaque couche réseau dispose d'un champ pour identifier le type de protocole encapsulé dans le champ de données. Ethernet identifie ainsi qu'il transporte un paquet IP, IP identifie qu'il transporte des données TCP, et TCP identifie l'application qui a rempli son champ de données.

Figure 7-8.
Le principe de l'encapsulation des protocoles.



Les champs " Type ", " Protocole " et " Port " permettent à chaque couche de savoir à quelle couche supérieure remettre les données reçues. La RFC 1700 recense ainsi toutes les valeurs affectées aux protocoles de la famille TCP/IP ou à ceux qui utilisent IP.

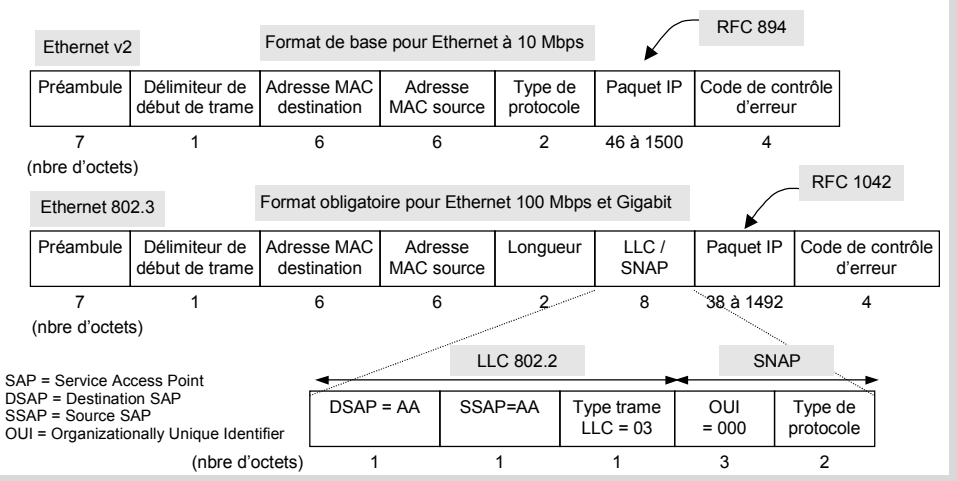
Il est ainsi possible d'envisager toutes les combinaisons d'encapsulation, telles que celle spécifiée par le protocole STUN (*Serial Tunneling*) qui permet de transporter dans un paquet IP une trame SDLC (*Synchronous Dala Link Control*) qui est un protocole de niveau 2 utilisé dans les réseaux SNA d'IBM. On pourra ainsi trouver l'encapsulation : " SDLC → TCP → IP → SNAP → LLC → Ethernet 802.3 ".

En théorie, tout protocole peut donc être encapsulé dans n'importe quel autre protocole. Dans la pratique, on utilise cette facilité pour répondre à une contrainte particulière, telle que le transport des flux SNA dans un réseau IP.

L'ENCAPSULATION D'IP DANS ETHERNET (RFC 894 ET 1042)

Il existe deux formats de trames : **Ethernet v2**, également appelée DIX (du nom des constructeurs Digital, Internet et Xerox), et **Ethernet IEE 802.3**. Il y a donc deux façons d'envoyer un paquet IP sur Ethernet : directement dans une trame Ethernet v2 (RFC 894) ou *via* un en-tête LLC/SNAP dans une trame 802.3 (RFC 1042).

Si la valeur du champ " Type de protocole/Longueur " est supérieure à 1 500 (correspondant au nombre maximal d'octets pour le champ contenant le paquet IP), il s'agit d'une trame Ethernet v2 (le champ a alors la signification " Type de protocole "). Dans le cas contraire, il s'agit d'une trame Ethernet 802.3 (le champ a alors la signification " Longueur ").



Il existe différents types de **trames LLC** impliquant différents modes de fonctionnement. Pour IP, seule la trame de type *Unnumbered Information* (type 03) est utilisée (trame simple sans acquittement). Elle est également utilisée pour transporter IP dans ATM (*Classical IP*), Token-Ring et FDDI.

La **couche SNAP** (*Sub Network Access Protocol*) est nécessaire car la trame LLC (*Logical Link Control*) ne contient pas de champ équivalent au champ " Type " de la trame Ethernet v2. Le SAP (*Service Access Point*) utilisé pour transporter SNAP est 170 (0xAA). On retrouve ce principe d'adaptation avec d'autres protocoles comme Frame-Relay ou ATM (voir chapitre 10).

Dans le dernier champ de l'en-tête SNAP, on retrouve enfin le **type de protocole** utilisé dont les valeurs sont identiques à celles du champ de même nom de la trame Ethernet v2 (**0x0800 pour IP**, 0806 pour ARP, etc.).
Il est à noter que le **MTU** (*Maximum Transfert Unit*), c'est-à-dire les données utiles transportées dans la trame Ethernet, est plus important avec Ethernet v2, l'encapsulation 802.3 faisant perdre 8 octets.

Dans le cas d'une navigation sur le web, l'empilement des protocole est : " segment HTTP → TCP (port 80) → IP (protocole 6) ", puis toutes sortes de réseaux de transport, tels que PPP, Frame-Relay, ATM, etc.

Sur votre réseau, l'encapsulation sera : " IP → SNAP (protocole 2048) → LLC (SAP 170) → Ethernet 802.3 ".

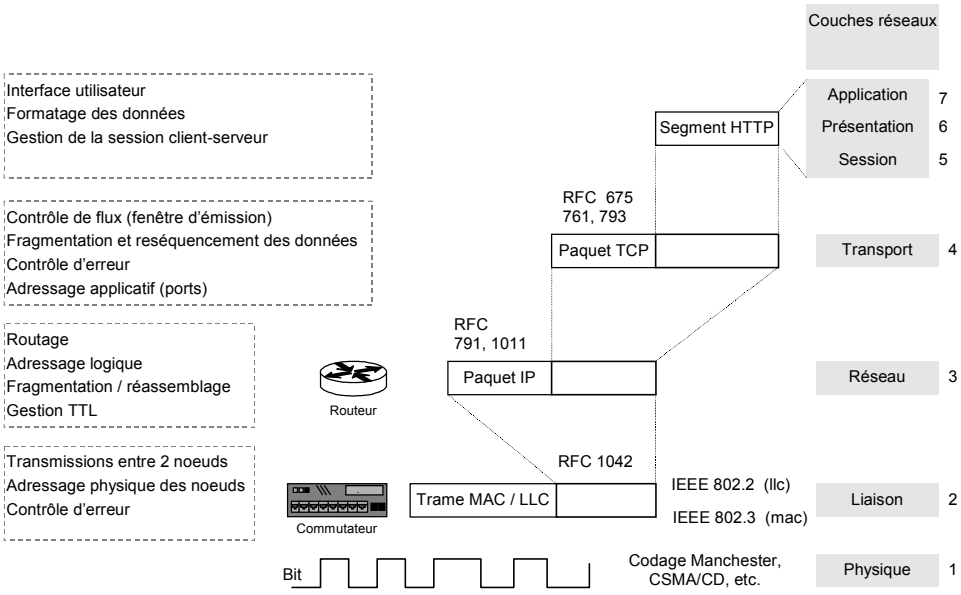


Figure 7-9.
Le rôle des couches réseau.

Ce modèle en couches simplifie la programmation des protocoles en leur assignant des rôles précis, et offre plus de souplesse par le jeu des encapsulations possibles.

Comment une station envoie-t-elle une trame Ethernet à une autre ?

Un réseau local tel qu'Ethernet est constitué de concentrateurs et de commutateurs, deux équipements au comportement bien différent.

Caractéristique	Ethernet partagé	Ethernet commuté
Équipement	Concentrateur	Commutateur
Segment	Un segment partagé par tous les ports	Un segment par port
Architecture matérielle	Composants électroniques ; un ou plusieurs bus Ethernet	ASIC et processeur RISC, matrice de commutation
Architecture logicielle	Petit logiciel pour des options de configuration et pour l'administration SNMP	Logiciel plus complexe, mais traitement essentiellement matériel
Algorithme de routage	Aucun	Spanning tree
Traitement des trames	Aucun traitement (transparent pour les trames Ethernet)	Filtrage et transmission (*forward*) des trames ; apprentissage des adresses MAC
Couche réseau	Niveau 1 = couche physique	Niveaux 1 et 2 = couche physique et liaison
Fonction	Connecter plusieurs PC au sein d'un segment	Interconnecter plusieurs segments

Échange de trames sur un segment Ethernet

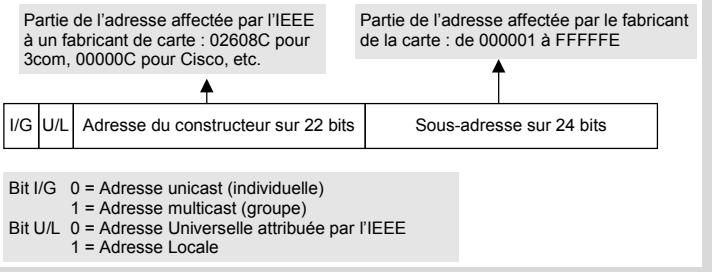
Commençons par étudier le fonctionnement au sein d'un segment Ethernet partagé (la trame circule sur un seul segment).

La norme Ethernet spécifie l'utilisation d'adresses physiques liées aux cartes réseau : les adresses MAC.

La carte recevant une trame Ethernet ne la prendra en compte que si l'adresse MAC de destination de la trame est identique à celle qui est inscrite dans sa mémoire. La seule exception à cette règle concerne les adresses de broadcast et, éventuellement, les adresses multicast. En résumé, les cartes sont programmées pour accepter les trames qui leur sont destinées, plus toutes les trames de broadcast ainsi que les trames multicast qui ont été configurées.

LES ADRESSES MAC (IEEE 802.3)

Une adresse MAC est constituée de 6 octets, par exemple 02:60:8C:EB:25:D2.



Chaque carte intègre dans sa mémoire une adresse MAC qui est unique dans le monde. L'IEEE (*Institute of Electrical and Electronics Engineers*) affecte les trois premiers octets à un fabricant de carte qui a la charge d'attribuer les trois derniers octets aux cartes qu'il fabrique. L'adresse FF:FF:FF:FF:FF:FF est l'adresse de broadcast.

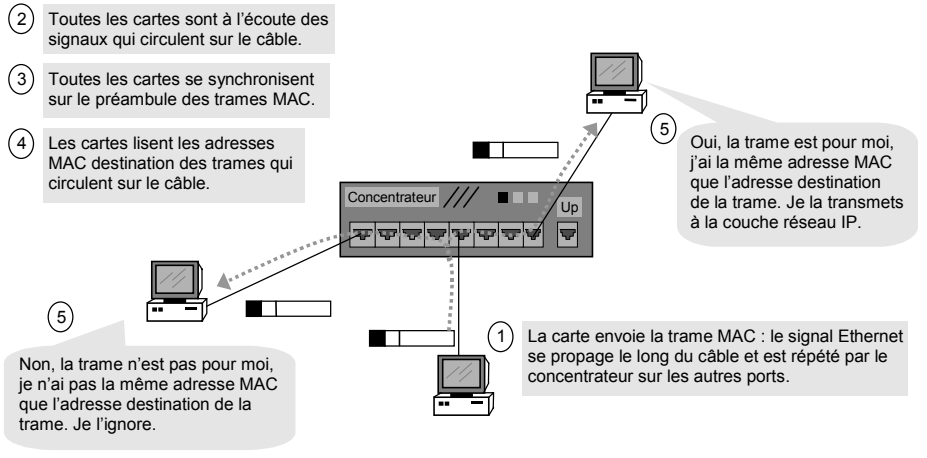


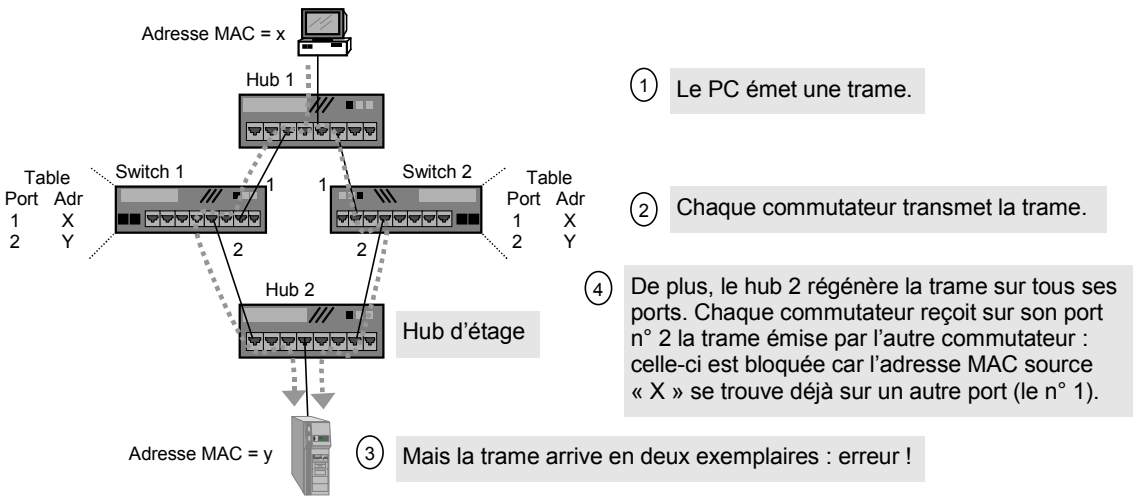
Figure 7-10. *Échange des trames Ethernet.*

Les trames qui sont acceptées sont remises au protocole de niveau 3, qui correspond à l'identifiant trouvé dans le champ " Type ", soit 0×0800 pour IP. Celles qui ne sont pas destinées à la station sont ignorées.

Échange de trames entre différents segments Ethernet

On l'a vu, le commutateur est un équipement permettant de segmenter le réseau Ethernet et d'interconnecter ces différents segments. Pour décider si la trame doit passer d'un segment à l'autre, il se base sur son adresse MAC qu'il compare avec celles se trouvant dans ses tables d'adresses en mémoire.

On peut envisager différentes situations dans lesquelles il existe plusieurs chemins possibles entre deux stations.



Il faut noter que, si le serveur n'a jamais émis de trame, son adresse MAC " Y " n'est pas encore connue des commutateurs. Ces derniers transmettront alors la trame sur tous les autres ports, dont le port n° 2. De plus, la même situation se produit pour toutes les trames de broadcast et de multicast.

Pour éviter ces problèmes, il faut qu'une des deux routes soit interdite. C'est là qu'intervient le **spanning tree**. Le but de ce protocole est de définir une route unique vers un commutateur désigné racine en se basant sur des coûts et des priorités.

LE POINT SUR LE SPANNING TREE (IEEE 802.1D)

L'algorithme du spanning tree consiste à construire un arbre définissant un chemin unique entre un commutateur et sa racine.

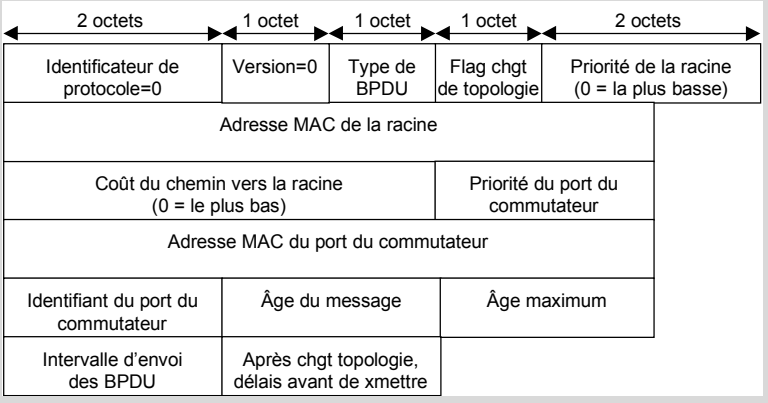
Lors de la construction de l'arbre (suite à un changement de topologie), chaque commutateur émet un **BPDU** (*Bridge Protocol Data Unit*) de configuration sur tous ses ports. Inversement, il retransmet tous les BPDU (éventuellement en les modifiant) qui lui arrivent, et ainsi de suite jusqu'à ce que les BPDU échangés contiennent tous la même valeur.

La première étape de ce processus consiste à élire un **commutateur racine** : c'est celui dont la **priorité** est la plus basse ou, en cas d'égalité, celui dont l'adresse MAC est la plus basse.

Ensuite, chaque commutateur détermine le **port racine** — celui par lequel un BPDU émis par la racine arrive. S'il y en a plusieurs, le port choisi est celui qui a le **coût** de chemin vers la racine le plus bas. Le coût est déterminé par la somme des coûts des ports situés entre le commutateur et la racine. En cas d'égalité, le port choisi est celui qui a la priorité la plus basse ; en cas de nouvelle égalité, c'est celui qui a l'adresse MAC la plus basse.

Enfin, sur chaque segment Ethernet, le commutateur dont le port racine a le coût de chemin vers la racine le plus bas est élu **commutateur désigné**. En cas d'égalité, c'est celui qui a la priorité la plus basse et, en cas de nouvelle égalité, celui qui a l'adresse MAC la plus basse.

En définitive, sur chaque segment Ethernet, un seul chemin vers le commutateur racine sera calculé. Les commutateurs désactivent tous leurs ports qui ne sont ni racines ni désignés.



Afin de détecter les changements de topologie (apparition ou disparition d'un commutateur), la racine envoie régulièrement (toutes les deux secondes par défaut) un BPDU d'annonce (comprenant seulement les trois premiers octets) sur tous ses ports. Les commutateurs transmettent ce BPDU sur leurs ports désignés.
Les BPDU sont envoyés dans des trames Ethernet multicast 01:80:C2:00:00:10.

Compte tenu des processus d'élection, il est important de bien paramétrer les coûts (exprimés en nombre de sauts et/ou dépendant du débit du port) ainsi que les priorités. Les valeurs par défaut (fixées en usine) peuvent en effet aboutir à choisir des chemins qui ne sont pas les meilleurs. Prenons le cas de notre réseau local redondant.

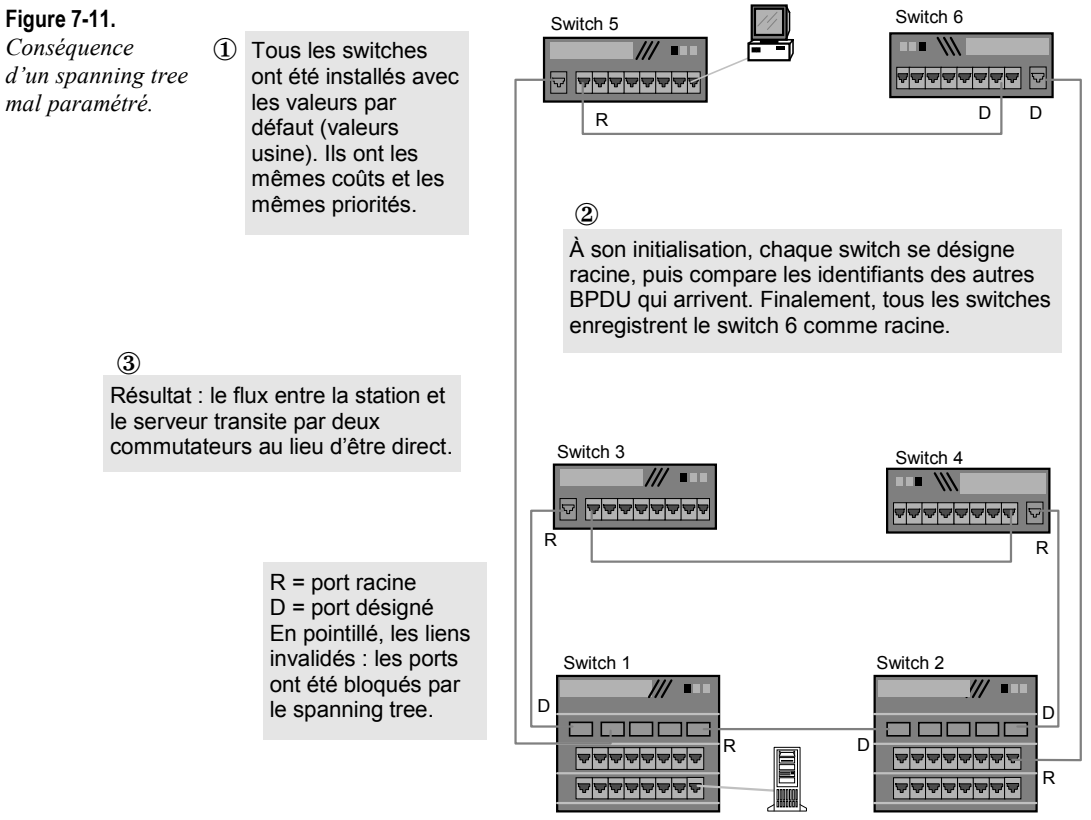


Figure 7-11.
Conséquence d'un spanning tree mal paramétré.

Tous les commutateurs ont la même priorité (par exemple 32 768) et le même coût sur chaque port de même débit (par exemple 19 pour les ports à 100 Mbit/s). Le switch 6 a été désigné racine parce que son identifiant (priorité + adresse MAC) était le plus bas. Le même processus de sélection a déterminé les routes menant vers la racine uniquement en se fondant sur les valeurs des adresses MAC, puisque toutes les autres valeurs (priorité et coût) sont identiques.

Résultat, certains flux ne sont pas optimisés et peuvent dégrader les performances.

Reprenons les différentes phases de calcul de l'arbre spanning tree. Les commutateurs désignent la racine. Afin d'optimiser les flux, il est préférable que ce soient les commutateurs fédérateurs qui assurent ce rôle. Leur priorité doit donc être abaissée par rapport aux commutateurs d'étage. Étant donné qu'il s'agit de Catalyst 5000, la commande est la suivante :

```
Console> (enable)set spantree priority 10000
VLAN 1 bridge priority set to 10000.
```

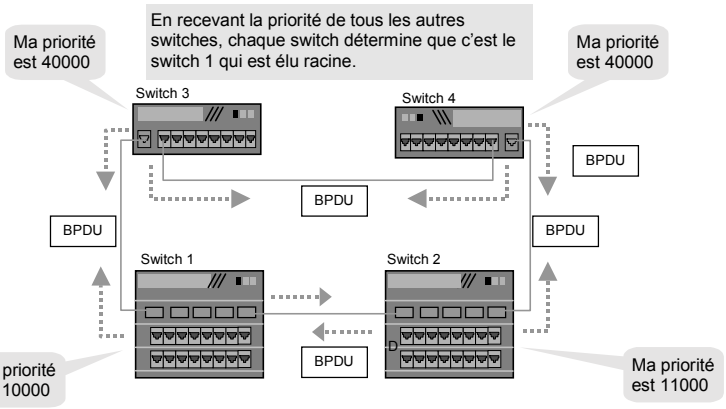


Figure 7-12.
Élection du commutateur racine.

Chaque commutateur choisit ensuite son port racine, celui dont le coût de chemin vers la racine est le plus bas. Sur un Catalyst, le coût de chaque port dépend de son débit : 4 pour 1 Gbit/s, 19 pour 100 Mbit/s, et 100 pour 10 Mbit/s. La commande suivante permet de changer la valeur par défaut :

```
Console> (enable)set spantree portcost 1/1 4
```

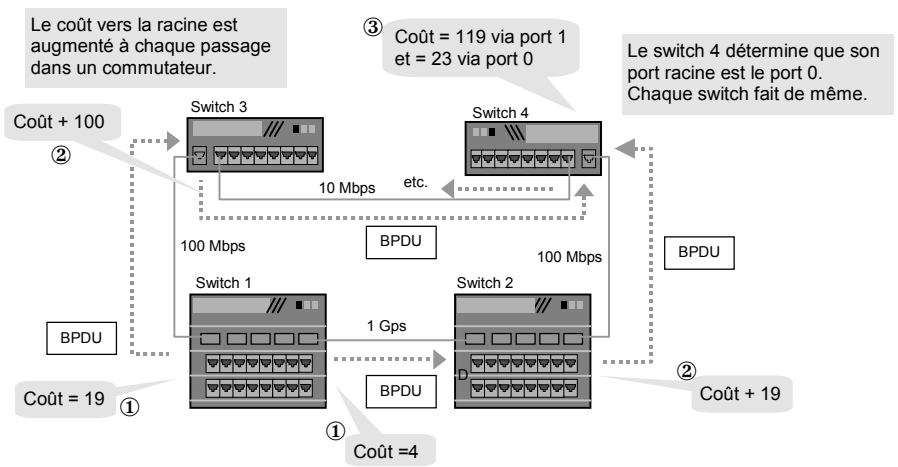


Figure 7-13.
Choix des ports racines.

Sur chaque segment Ethernet, le commutateur désigné est celui dont le port racine a le coût le plus bas. En cas d'égalité, la priorité détermine ce coût. Étant donné que tous les ports ayant un même débit ont le même coût et la même priorité par défaut, le choix s'effectuera en fonction de l'adresse MAC. Pour éviter les mauvaises surprises, il est possible d'abaisser la priorité d'un port pour être sûr qu'il soit désigné en cas de routes multiples :

```
set spantree portpri 1/1 32
```

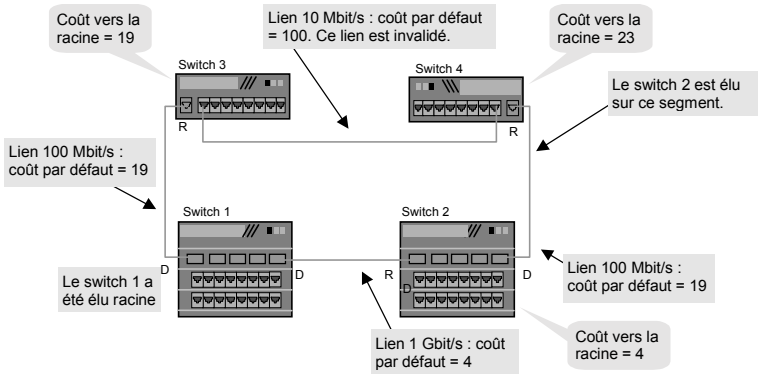


Figure 7-14.
Élection des commutateurs désignés.

En définitive, les chemins redondants ne sont pas utilisés, et le partage de la charge entre plusieurs routes n'est pas possible.

Les mêmes BPDU sont envoyés sur tous les ports, même là où il n'y a qu'un PC connecté. Le spanning tree peut donc être désactivé sur ces ports, ce qui présente l'avantage de diminuer (un peu) le trafic et d'éviter que des ajouts sauvages de commutateurs (qui seraient connectés sur ces ports) ne viennent perturber votre réseau.

```
set spantree disable
```

Le commutateur racine émet régulièrement (toutes les deux secondes par défaut) des BPDU pour maintenir l'état du spanning tree. Si le réseau est stable (peu d'incidents et de changements), il est possible d'augmenter cette valeur afin de diminuer le trafic

```
set spantree hello 5
```

On peut s'assurer que, sur le switch 1, le spanning tree s'est stabilisé dans une bonne configuration.

```
Console> (enable) show spantree
VLAN 1
Spanning tree enabled
Designated Root             00-1f-00-40-0b-eb-25-d2
Designated Root Priority    45
Designated Root Cost        0
Designated Root Port        1/1
Root Max Age   20 sec    Hello Time 2  sec   Forward Delay 20 sec
Bridge ID MAC ADDR          00-40-0b-eb-25-d2
Bridge ID Priority          45
Bridge Max Age 20 sec    Hello Time 2  sec   Forward Delay 20 sec
```

```
Port      Vlan  Port-State     Cost   Priority  Fast-Start
--------  ----  -------------  -----  --------  ----------
1/1       1     forwarding         4        32   disabled
1/2       1     forwarding        19        32   disabled
2/1       1     forwarding        19        32   disabled
2/2       1     not-connected     19        32   disabled
2/3       1     not-connected     19        32   disabled
3/1       1     not-connected    100        32   disabled
3/2       1     forwarding       100        32   disabled
```

Le processus de création de l'arbre spanning tree peut durer plusieurs dizaines de secondes. Ce temps est, en réalité, proportionnel au nombre de commutateurs.

Pendant cette phase, aucun commutateur ne traite de trame au cours des 15 premières secondes (valeur par défaut) ; le réseau s'arrête donc de fonctionner chaque fois qu'un commutateur est allumé ou éteint quelque part dans le réseau.

La phase d'apprentissage est encore plus longue lorsque tous les commutateurs s'initialisent en même temps (suite à une panne de courant, par exemple). En effet, les BPDU sont reçus par plusieurs ports dont l'un peut être élu racine, puis invalidé par la suite si un commutateur situé en aval a invalidé sa route. Le temps de stabilisation de l'arbre peut ainsi atteindre plusieurs minutes.

Dans certains cas, notamment sur les commutateurs fédérateurs, il peut être intéressant de diminuer ce temps, surtout si l'architecture réseau est conçue sans aucune boucle.

```
set spantree fwddelay 5
```

Inversement, si ce temps est trop court par rapport au délai de construction de l'arbre, des trames peuvent commencer à circuler et potentiellement être dupliquées dans le cas de routes multiples. Il vaut alors mieux augmenter le paramètre " forward delay " au-delà des 15 secondes par défaut.

La meilleure solution consiste à activer plus rapidement les ports qui ne sont pas concernés par le *spanning tree*, c'est-à-dire ceux sur lesquels sont connectés à une seule station.

```
set spantree portfast 1/2 enable
```

On peut remarquer que l'échange de BPDU et l'élection d'un commutateur désigné impliquent que chaque port du commutateur soit identifié par une adresse MAC (comme une carte réseau).

Comment une station envoie-t-elle un paquet IP à une autre ?

Une carte réseau ne se préoccupe que des adresses MAC pour envoyer et recevoir des données. En revanche, une application telle que Telnet ne connaît que l'adresse IP qui est purement logique : une pile IP recevant un paquet IP ne le prendra en compte que si l'adresse de destination du paquet correspond à l'adresse IP qui a été paramétrée dans le PC. Dans le cas contraire, il sera ignoré.

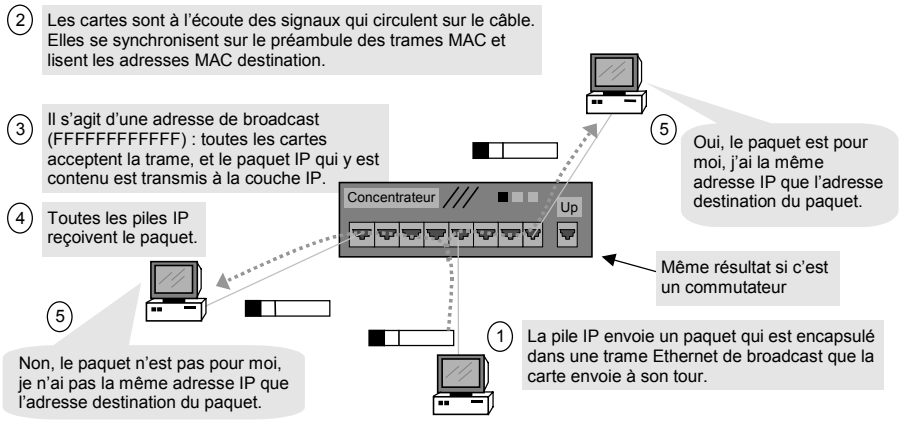


Figure 7-15. *Échange de paquets IP.*

Par ailleurs, l'adresse MAC de la station changera si la carte réseau est changée (en cas de panne, par exemple). De même, son adresse IP peut être modifiée à tout moment à l'aide des outils de configuration Windows (en cas de déménagement, par exemple).

L'exemple précédent montrait un paquet IP envoyé dans une trame de broadcast. Ce moyen d'opérer est pratique mais très consommateur de bande passante puisque la trame est propagée à travers tout le réseau. Sauf quand cela est nécessaire, un paquet IP est envoyé dans une trame unicast, c'est-à-dire directement au PC concerné. Mais comment connaître l'adresse MAC de la carte du PC destinataire alors que vous ne connaissez que l'adresse IP de sa pile IP ?

Cela est, par exemple, le cas lorsque vous lancez la commande suivante, qui permet de vous connecter à un serveur Unix :

```
Telnet 192.50.10.1
```

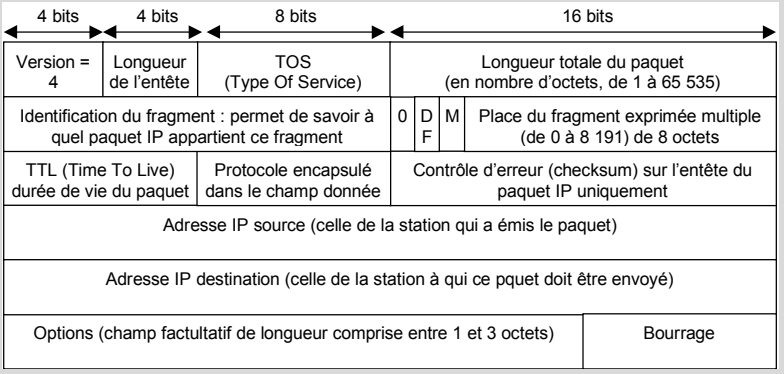
L'application Telnet va demander à la couche TCP d'ouvrir une connexion avec l'adresse IP indiquée, et va transmettre son paquet à la couche IP (avec l'adresse de destination indiquée). Cette dernière va encapsuler le paquet TCP dans un paquet IP, puis l'envoyer à la carte. Mais la carte ne sait pas quoi faire d'une adresse IP ; elle ne sait gérer que des adresses MAC : une trame Ethernet ne contient qu'une adresse MAC qui permet aux autres cartes de la prendre ou non en compte.

La solution repose sur un mécanisme qui réalise la correspondance entre l'adresse MAC du PC destinataire et son adresse IP.

On pourrait utiliser une table de correspondance statique Adresse MAC ↔ Adresse IP. Mais cela serait fastidieux, car il faudrait relever les adresses MAC des stations ainsi que les adresse IP, et paramétrer la table sur tous les PC. Cela est inimaginable étant donné le nombre important de PC et les nombreux changements d'adresses qui interviennent. On perdrait en plus l'avantage de dissocier l'adresse physique de l'adresse logique. En outre, un PC peut être configuré avec plusieurs adresses IP.

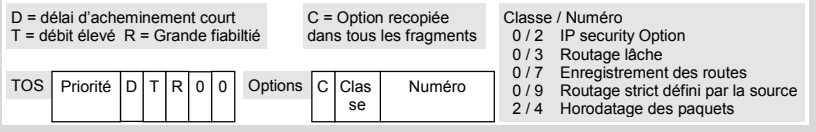
LE POINT SUR IP V4 (RFC 791)

IP (*Internet Protocol*) est un protocole de niveau 3 (couche réseau) qui découpe les réseaux locaux en réseaux logiques indépendamment de leur implémentation physique. Ce protocole permet donc d'envoyer des données à travers **un réseau virtuel** reposant sur des réseaux physiques de différente nature (Ethernet et PPP, par exemple). Pour ce faire, IP utilise un **adressage logique** différent de l'adressage physique (MAC, PPP ou autre).



Cette couche se contente de **router** (c'est-à-dire acheminer) le paquet à travers un réseau IP : les paquets peuvent être perdus (pas de garantie d'acheminement), contenir des erreurs (sauf sur l'en-tête, qui est contrôlé) ou arriver dans le désordre. IP **fragmente** les paquets dont la taille excède celle des trames (le MTU, *Maximum Transfer Unit)*. Les fragments sont routés indépendamment les uns des autres comme autant de paquets, mais IP **assemble** dans le bon ordre les fragments d'un même paquet original. Si le bit " DF " est positionné à 1, la fragmentation est interdite. Le bit " M " positionné à 0 indique que ce paquet est le dernier fragment d'une série lorsque le bit " DF " est positionné à 0.

Le champ **TTL** est décrémenté de 1 chaque fois que le paquet passe par un routeur. Si la valeur atteint 0, le routeur détruit le paquet. Ce mécanisme évite aux paquets de rester trop longtemps sur le réseau, soit parce qu'ils tournent en boucle (suite à une erreur de routage), soit parce qu'ils traversent trop de routeurs. La valeur initiale du TTL est fixée par la station émettrice (de 32 à 128, en général).



Le champ **TOS** permet de décrire la qualité de service souhaitée. La signification de ce champ est abordée au chapitre 14.

La résolution d'adresse

La seule solution est donc une résolution d'adresse automatique, c'est-à-dire un mécanisme permettant de trouver l'adresse MAC en connaissant uniquement l'adresse IP. C'est le rôle du protocole **ARP** (*Address Resolution Protocol*) lié à la couche IP. Ce protocole gère une table de correspondance dynamique Adresse MAC ↔ Adresse IP, appelée **cache ARP**. Vous pouvez en visualiser le contenu à l'aide de la commande Windows suivante :

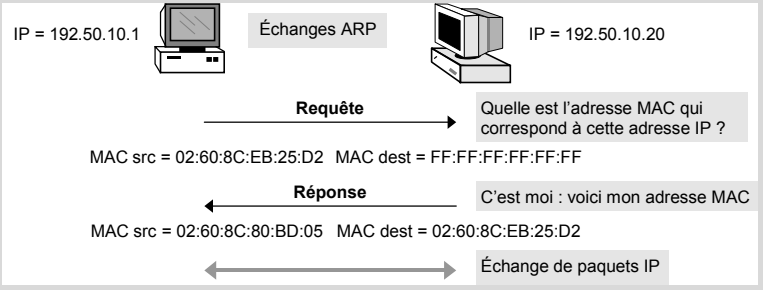
```
arp –a
00:40:0b:4b:25:d2    190.50.1.253
```

LE POINT SUR ARP (RFC 826)

Pour obtenir l'adresse MAC d'une station ne connaissant que son adresse IP, la pile TCP/IP émet une requête ARP (Address Resolution Protocol) dans une trame Ethernet de broadcast dont le champ « Type » contient la valeur **0x0806**.

Chaque pile IP recevant un tel paquet compare alors son adresse avec celle figurant dans le champ « Adresse protocole destination ».

S'il y a correspondance, la couche ARP envoie un paquet de réponse en remplissant le champ « Adresse physique destination » avec l'adresse MAC de sa carte. Dans le cas contraire, le paquet est ignoré.



Donc, seule la station dont l'adresse IP correspond à celle demandée par la requête envoie en réponse un paquet contenant sa propre adresse MAC.

La résolution inverse, c'est-à-dire l'obtention de l'adresse IP à partir de l'adresse MAC, est réalisée par le protocole **RARP** (*Reverse ARP* – RFC 903).

Si vous n'avez pas communiqué récemment avec un autre PC, la table sera vide : les entrées sont, en effet, effacées au bout d'un certain temps. Sous Windows, une entrée ARP (adresse MAC / adresse IP) est supprimée au bout de deux minutes si le PC n'a pas dialogué avec la station cible (selon le mécanisme TTL, *Time To Live*). Dans tous les cas, l'entrée reste au maximum dix minutes en mémoire, puis elle est supprimée.

Si l'adresse IP recherchée n'est pas dans le cache, ARP va alors envoyer un paquet de requête encapsulé dans une trame Ethernet de broadcast. Cette dernière va donc être lue par toutes les cartes réseau.

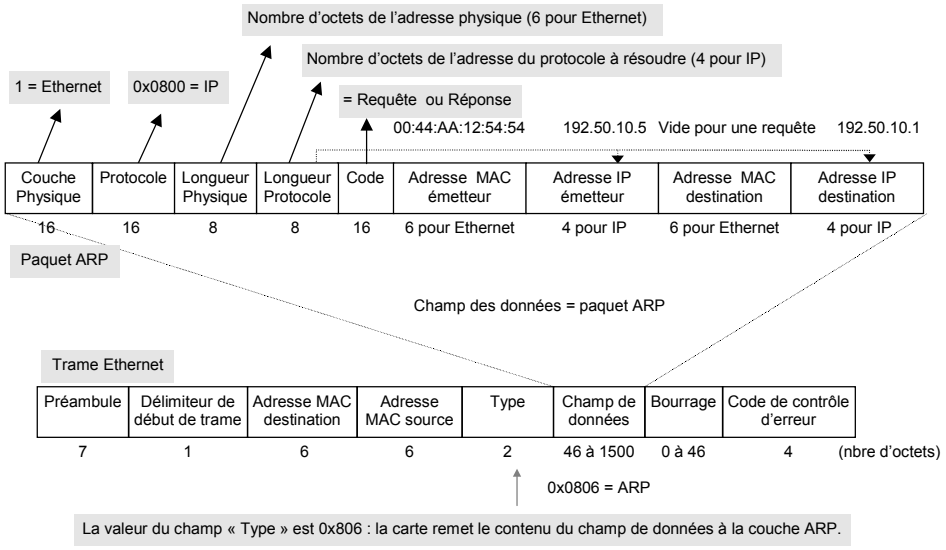


Figure 7-16. *Format d'un paquet ARP dans une trame Ethernet.*

Seule la station configurée avec l'adresse IP recherchée va répondre en renvoyant son adresse MAC.

Une fois l'adresse résolue, le paquet IP peut être envoyé dans une trame MAC unicast dont l'adresse de destination est celle de la station cible.

Comment une application envoie-t-elle des données ?

Une application utilise les services de la couche transport avec qui elle échange des données à travers une interface de programmation livrée avec la pile TCP/IP. Sous Unix, il s'agit des Sockets ; sous Windows de Winsock.

La couche transport est soit **TCP** (*Transport Control Protocol*) soit **UDP** (*User Datagram Protocol*), qui est une version allégée de TCP.

Le protocole TCP agit en **mode connecté**, ce qui implique que le client demande l'ouverture d'une connexion préalablement à tout échange. Par exemple, lorsque vous entrez la commande Windows " Telnet 192.50.10.1 ", le programme client Telnet demande à TCP d'ouvrir une connexion à un serveur Telnet qui est en attente, c'est-à-dire à l'écoute du **port** TCP 23.

Inversement, UDP agit en mode **non connecté**, ce qui permet à deux machines d'échanger des données à tout moment, sans entrer dans une phase de connexion. Par exemple, lorsque vous voulez vous connecter à un serveur de fichiers Windows NT, votre PC émet une demande de résolution de nom à destination d'un serveur WINS qui est à l'écoute sur le **port** UDP 137.

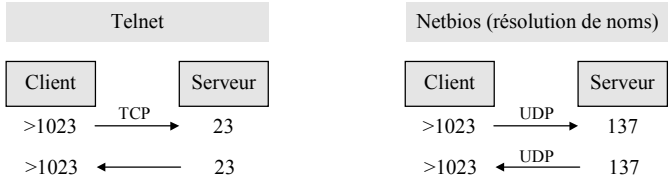
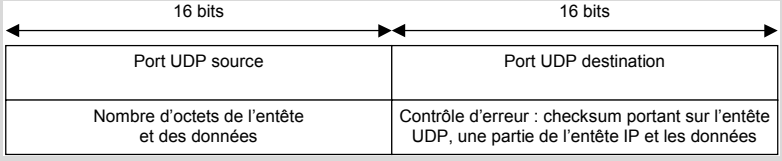


Figure 7-17.
Utilisation des ports TCP et UDP.

La pile TCP/UDP du client choisit généralement un port source supérieur à 1023, et incrémente cette valeur à chaque nouvelle session ouverte simultanément à d'autres déjà actives.

LE POINT SUR UDP (RFC 768)

UDP (*User Datagram Protocol*) permet simplement à une application d'avoir accès au réseau IP. Ce protocole n'offre aucune garantie d'acheminement, aucun mécanisme de reprise sur erreur, ni de contrôle de flux, et ne vérifie pas la duplication des paquets. Tous ces contrôles doivent être opérés par les autres couches réseau. En revanche, les paquets remis à l'application le sont sans erreur.



Les champs **port source** et **port destination** servent à identifier une application (par exemple, 23 pour Telnet). Ces valeurs sont réservées et enregistrées par l'IANA (*well known port* – RFC 1700).

Les clients et les serveurs manipulent des noms (une machine Unix, un serveur de fichiers NT) et s'échangent des données à travers des ports qui leur sont réservés par l'IANA (voir chapitre 3). Sur l'Internet, le service DNS permet de convertir les noms en adresses IP (voir chapitre 17), tandis que, dans le monde Microsoft, on utilise encore le service WINS pour convertir des noms **Netbios** en adresses IP.

On peut considérer que ce protocole est situé au niveau de la couche 5 (couche session) : il permet, en effet, d'établir et de gérer des sessions entre applications. Dans le monde Internet, les applications comme Telnet, votre navigateur web, FTP, etc.) gèrent elles-mêmes tous les mécanismes situés au-dessus de la couche transport, c'est-à-dire TCP et UDP.

À l'origine, Netbios circulait nativement dans des trames Ethernet, mais de nos jours, il est encapsulé dans IP (RFC 1001 et 1002).

Par exemple, le partage de fichiers et la messagerie Exchange utilisent le protocole Netbios sur le port TCP 139. Par ailleurs, les serveurs WINS s'échangent des données sur le port TCP 42.

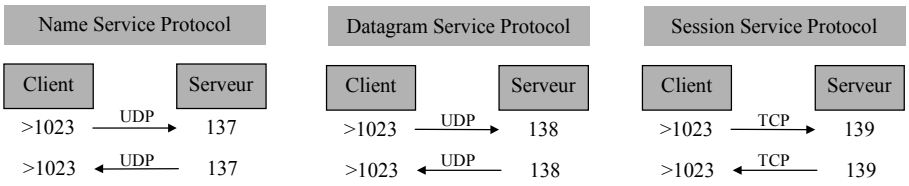
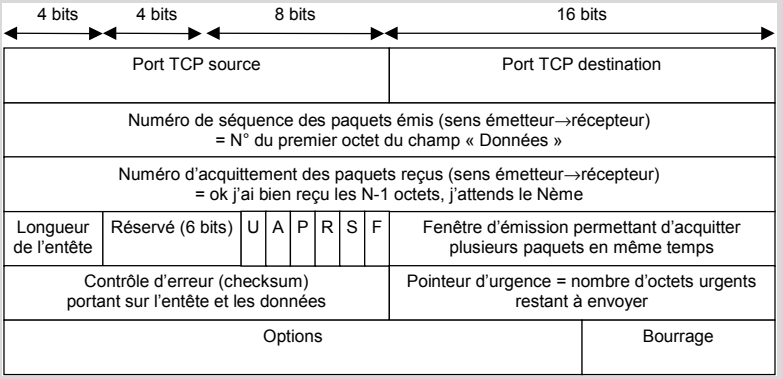


Figure 7-18.
Netbios sur IP.

LE POINT SUR TCP (RFC 793)

TCP (*Transport Control Protocol*) est un protocole de niveau 4 (couche transport) qui permet à deux applications (un client et un serveur) d'échanger des données en leur masquant les mécanismes réseau. Les paquets TCP sont transportés dans des paquets IP de type 6 (champ protocole = 6).

TCP offre un service de **bout en bout** (entre deux entités, quelle que soit leur localisation) en **mode connecté** (un client doit se connecter à une application serveur). Il utilise pour cela un adressage applicatif basé sur des **ports TCP**. Chaque application est identifiée par un numéro de port réservé (*well known port*). Généralement, le client choisit un numéro de port aléatoire supérieur à 1023 comme port source. Le serveur lui répond sur ce port.



Les bits de contrôle U, A, P, R, S et F ont la signification suivante :
U = Urgent. Indique que les données doivent être remises sans délai à l'application.
A = Ack. Acquittement d'une demande de connexion ou de fermeture.
P = Push. Indique à la couche TCP d'envoyer et de remettre les données sans attendre le remplissage des tampons d'émission et de réception.
R = Reset. Ferme la connexion TCP suite à un problème.
S = Synchhronize. Le numéro de séquence est réinitialisé à une valeur aléatoire.
F = Fin. Demande de déconnexion.

Des options peuvent être négociées entre entités TCP (champ " Option "), par exemple la taille maximale des segments transportés.

La couche TCP assure le **contrôle d'erreur** et le **séquencement** des paquets (les paquets sont remis dans le même ordre que lors de leur émission). La taille de la **fenêtre d'émission** indique le nombre de paquets pouvant être acquittés en même temps. Elle permet également de demander la retransmission à partir du premier paquet en erreur (manquant ou erroné).

La couche TCP mesure le temps écoulé entre l'émission d'un paquet et la réception de l'accusé de réception correspondant, et calcule ainsi une moyenne glissante du temps de réponse (R*ound Trip Time*). Elle utilise l'algorithme de Karn pour déduire la valeur de ses temporisateurs. Par exemple, plus le temps de réponse est long, plus TCP attendra longtemps l'accusé de réception avant de retransmettre. De même, TCP estime le nombre de paquets perdus : plus celui-ci augmente, plus la fenêtre d'émission est réduite. Ces mécanismes permettent à TCP de **contrôler le flux** de données en fonction de l'état du réseau (perte de paquets et débits) et donc d'éviter une surcharge du réseau par un nombre croissant de retransmissions devenues inutiles.