

Notion de DNS

Table des matières

1. Introduction.....	2
2. Réseaux IP.....	2
3. Le routage.....	5
4. Domain Name System.....	7
4.1. Introduction.....	7
4.2. La structure du système de nommage.....	8
4.3. La résolution de noms.....	9
4.4. Le courrier électronique.....	9
4.5. Les serveurs de noms.....	9

En janvier 2003, le nombre de machines directement accessibles sur le réseau était de 180 000 000 selon l'ISC. Depuis on ne compte plus tant la croissance est importante... Pour la France l'AFNIC propose également quelques statistiques... Il n'existe pas de bottin général du réseau, par contre Bill Cheswick des Bell labs l'a cartographié, et [le résultat est fascinant](#) !



1. Introduction

Un réseau est un ensemble de dispositifs matériels et logiciels permettant à 2 machines ou plus de communiquer. Pour mettre en évidence les concepts importants, on peut utiliser l'analogie avec la vie quotidienne:

Imaginez que vous (M. Dupond) ayez un client (M. Schmidt) au téléphone. Celui-ci vous soumet un problème délicat: vous attirez l'attention de votre collègue (M. Jones) et lui griffonnez quelques mots sur un bout de papier. M. Jones réfléchit un instant et vous griffonne sa réponse.

MM. Dupond, Schmidt et Jones représentent les machines connectées. Ces personnes sont reliées par 2 réseaux ("Liaison téléphonique" et "Liaison visuelle") auxquels ils sont reliés par 2 types d'interfaces ("Combiné téléphonique" et "Papier + Crayon"). M. Dupond, qui possède les 2 types d'interfaces, permet d'établir une communication entre M. Schmidt et M. Jones: il sert de passerelle entre les 2 réseaux. Enfin, chaque personne parle pendant un temps puis attend une réponse de son interlocuteur : ils communiquent en échangeant des paquets d'informations.

A la lumière de cet exemple, on peut donc formuler ainsi les définitions des mots importants :

Réseau Support permettant les échanges. Ce terme générique peut aussi bien désigner un câble coaxial reliant 2 machines qu'une ligne spécialisée (dont l'autre extrémité est reliée à une installation complexe, mais qui est vue comme un support ordinaire du point de vue d'une interface).

Le terme de réseau désigne aussi la totalité de l'installation (machines, interfaces et câblage). Le contexte permet de distinguer facilement le sens dans lequel il doit être compris.

Interface Dispositif assurant la connexion de la machine à un réseau. Les interfaces utilisées avec les réseaux informatiques sont les cartes réseaux, les modems et les ports parallèles.

Routeur Machine assurant l'interconnexion de plusieurs réseaux. On parle aussi de passerelle (en anglais : "router" et "gateway"). Il peut s'agir d'une machine du réseau dans laquelle on a installé plusieurs interfaces, ou d'un appareil spécialisé n'assurant que cette fonction.

Paquet Unité de transport d'information.

Protocole Ensemble de règles régissant les échanges d'informations.

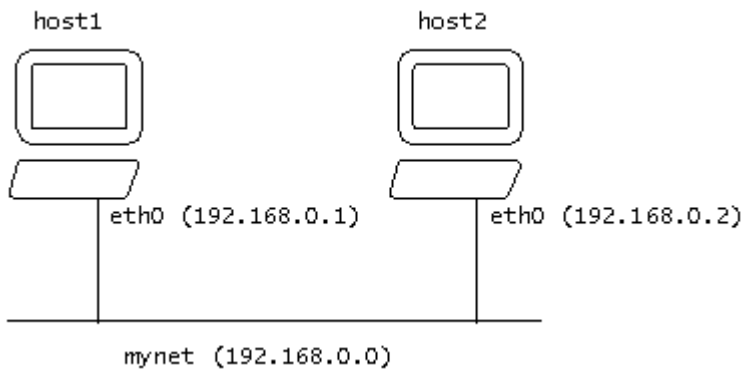
Adresse Identification des éléments intervenant dans la communication (interfaces et réseaux). L'analogie postale est particulièrement bonne: le nom de la rue correspond à l'adresse de réseau et le numéro de la maison à l'adresse de l'interface.

2. Réseaux IP

Les réseaux IP sont caractérisés par leur indépendance par rapport au matériel et par la possibilité d'établir une communication entre 2 machines situées des réseaux différents (on dit que c'est un protocole "routable"). Ces avantages découlent du mode d'adressage choisi : on affecte à chaque interface une adresse logique découlant de l'adresse du réseau auquel elle est connectée.

Puisqu'il s'agit d'une adresse logique, on peut facilement faire évoluer le matériel (il suffit de donner à la nouvelle machine l'adresse de l'ancienne), et puisqu'elle découle de l'adresse du réseau auquel elle est reliée, il est facile de détecter si la machine à laquelle on veut envoyer des données est sur le même réseau ou non (auquel cas on doit passer par un routeur).

Prenons un exemple simple :



host1 et host2 sont les noms des machines, eth0 le nom des interfaces, et mynet le nom du réseau.

On a donc commencé par choisir une adresse de réseau : 192.168.0.0. Ensuite, on affecte une adresse logique à chacune des interfaces qui y est reliée, l'adresse de l'interface découlant de celle du réseau : 192.168.0.1, 192.168.0.2, etc.

Les adresses IP sont des nombres codés sur 32 bits. Par commodité, on les représente en décimal sous la forme de 4 nombres (de 0 à 255 chacun) séparés par des point ("dotted quad" en anglais).

Nous avons vu que l'adresse d'une interface découle de celle du réseau. La relation entre les 2 consiste à "couper" l'adresse en 2 parties : les bits situés à gauche du point de coupure constituent la partie fixe (l'adresse de réseau) et ceux situés à droite sont disponibles pour numéroter les interfaces. L'adresse du réseau a donc tous les bits de la partie variable à 0. L'adresse dont tous les bits de la partie variable sont à 1 est réservée.

D'un point de vue arithmétique, on peut donc considérer que l'adresse de réseau peut être déterminée à partir de l'adresse d'une interface en faisant un ET logique entre l'adresse de l'interface et un nombre dont tous les bits de la partie fixe sont à 1 et ceux de la partie variable sont à zéro. Ce nombre est appelé masque de réseau (netmask en anglais). Le netmask est représenté sous forme de dotted quad, comme les adresses.

Selon la valeur de l'adresse IP, on distingue plusieurs classes de réseaux (correspondant à la position du point de coupure). Ces classes sont simplement des conventions destinées à faciliter l'attribution des plages d'adresses.

- Classe A : le bit de poids le plus fort est à 0. Le netmask correspondant est 255.0.0.0. Les réseaux 0.0.0.0 et 127.0.0.0 étant réservés, la classe A correspond donc à 126 réseaux de 16 millions de machines.
- Classe B : le bit de poids le plus fort est à 1, le suivant à 0. Son netmask est 255.255.0.0. Elle comporte donc environ 16000 réseaux de plus de 65534 machines.
- Classe C : les 2 bits de poids le plus fort sont à 1, le suivant à 0. Le netmask est 255.255.255.0 et on a 2 millions de réseaux de 254 machines.

Il existe d'autres classes spéciales ou expérimentales sur lesquelles nous ne nous étendrons pas. Enfin, il est possible de créer des sous-réseaux à partir d'une plage d'adresses donnée en utilisant un netmask comportant plus de bits à 1 que le netmask conventionnel.

Remarque : Pour indiquer une adresse de réseau et son netmask, on utilise souvent une notation consistant à faire suivre l'adresse d'un slash et du nombre de bits à un dans le netmask. Exemples :

- Adresse 192.168.12.0 - netmask 255.255.255.0 => 192.168.12.0/24
- Adresse 10.0.0.0 - netmask 255.0.0.0 => 10.0.0.0/8
- Adresse 10.0.0.0 - netmask 255.192.0.0 => 10.0.0.0/10
- Adresse 172.16.128.0 - netmask 255.255.128.0 => 172.16.128.0/17

Dans le cas d'un réseau local privé, on peut choisir n'importe quelle adresse. Dans le cas d'un réseau local connecté à Internet, votre fournisseur de connectivité IP vous indique la plage d'adresses qui vous a été allouée.

Certains numéros de réseaux ont été réservés pour les réseaux locaux privés (RFC 1597):

- Classe A : 10.0.0.0 (1 réseau)
- Classe B : 172.16.0.0 à 172.31.0.0 (16 réseaux)
- Classe C : 192.168.0.0 à 192.168.255.0 (256 réseaux)

Pour en finir avec les adresses réservées, signalons encore 2 cas: l'adresse de loopback et les adresses de broadcast.

L'adresse de loopback (127.0.0.1) correspond à une interface fictive présente sur toutes les machines, l'interface de loopback (lo ou lo0). Elle permet à la machine de s'envoyer à elle-même des paquets : on peut ainsi utiliser des applications réseau sans disposer d'interface physique, ou sans qu'elle soit reliée au réseau. On peut par exemple installer les programmes client et serveur d'une application dans la même machine pour aller faire une démonstration chez un (futur) client, ou encore travailler chez soi à la conception d'un site Internet, etc.

L'adresse de broadcast associée à un réseau correspond à une partie machine dont tous les bits sont à 1. Elle sert en particulier à déterminer quelle adresse physique correspond à une adresse IP donnée: une requête envoyée à l'adresse de broadcast est reçue par toutes les machines, mais seule celle dont l'interface correspond à l'adresse demandée répond.

Communiquer, c'est échanger des informations. Les règles régissant ces échanges sont appelées protocoles. Les principaux protocoles rencontrés avec les réseaux IP sont les suivants (liste loin d'être exhaustive).

Nom	Description
IP (Internet Protocol)	Échange de paquets entre nœuds
ICMP (Internet Control Message Protocol)	Transmission de messages d'erreur et de contrôle entre machines et passerelles
ARP (Address Resolution Protocol)	Correspondance entre adresse IP et adresse physique
RARP (Reverse Address Resolution Protocol)	Correspondance entre adresse physique et adresse IP
TCP (Transmission Control Protocol)	Trasmission de données en mode connecté (mode "Stream")

UDP (User Datagram Protocol)	Trasmission de données sans connexion (mode "Datagram")
FTP (File Transfer Protocol)	Services de haut niveau de transfert de fichiers
HTTP (HyperText Transfer Protocol)	Serveurs Web
NNTP (Network News Transfer Protocol)	Serveurs de News
SMTP (Simple Mail Transfer Protocol)	Transmission de courrier électronique
POP (Post Office Protocol)	Gestion de boîtes aux lettres électroniques
Telnet	Émulation de terminal

Cette liste permet de voir que les protocoles s'adressent à différents niveaux: un programme de transfert de fichiers fournira une interface utilisateur au protocole FTP, qui s'appuie lui même sur TCP, lequel utilise IP, ce dernier exploitant ARP et RARP. C'est pour cette raison qu'on parle souvent de la "pile TCP/IP", ou de la "suite de protocoles TCP/IP".

Les spécifications de ces protocoles sont appelées RFC (Request For Comments). On peut les trouver sur tous les grands sites FTP, notamment les sites universitaires, ceux des organismes de recherche et ceux des sociétés liées à l'internet. Exemples: ftp.lip6.fr, ftp.inria.fr, ftp.nic.fr.

Supposons maintenant qu'une machine soit utilisée pour faire plusieurs choses : serveurs Web, FTP, SMTP et POP. Cette machine n'a qu'une interface, donc qu'une adresse IP. Le problème qui se pose est de trouver un moyen d'indiquer à cette machine le programme auquel transmettre le paquet qu'on lui envoie.

La solution à ce problème est la notion de port: on affecte un numéro à chacun des services offerts par la machine (80 pour le serveur HTTP, 21 pour le serveur FTP, 25 pour le serveur SMTP et 110 pour le serveur POP).

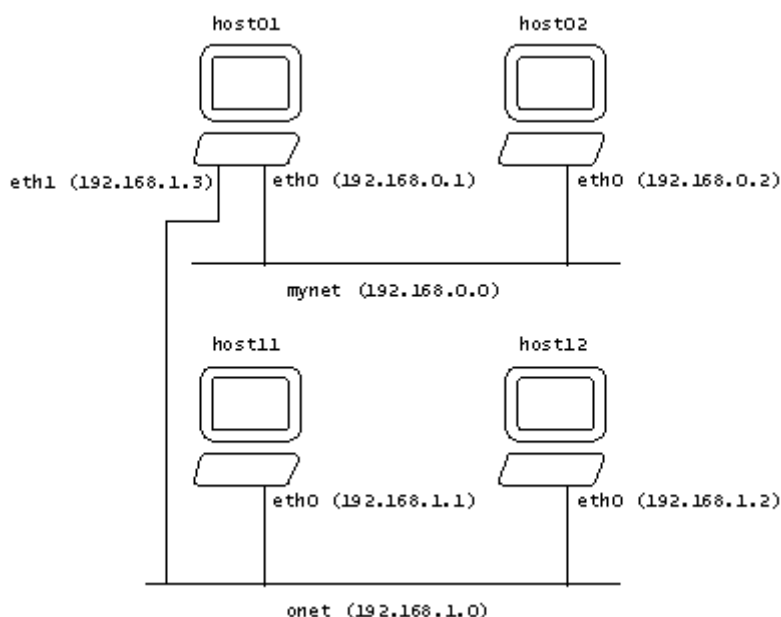
Notez qu'on peut transmettre des données à un port selon 2 modes : connecté (TCP) ou non (UDP). Le mode connecté (ou mode Stream) garantit que les données seront reçues en totalité et dans l'ordre où elles ont été envoyées. Le mode non-connecté (ou mode Datagram) permet d'envoyer un message isolé sans garantie qu'il soit reçu.

Cette dernière définition peut surprendre, mais ce mode est très utile : au cours d'un transfert de fichier (donc mode Stream), par exemple, on peut avoir besoin de transmettre un message d'alerte. On enverra alors cette information sous la forme d'un datagramme sur le même port (puisque'il est destiné à la même application). Le programme de réception peut détecter l'arrivée de ces données "hors bande" et appeler la procédure adéquate sans interférer avec le transfert en cours (sauf s'il s'agit d'une demande d'interruption...). Si par malchance le message d'erreur n'était pas reçu, il pourrait être envoyé une seconde fois : l'absence de garantie d'acheminement n'est donc pas un handicap.

3. Le routage

Pour permettre à 2 machines situées sur des réseaux différents, il est nécessaire d'établir une liaison entre ces 2 réseaux. La machine "host01" remplit cette fonction de passerelle entre les 2 réseaux : elle dispose de 2 interfaces reliées chacune à un réseau. Il ne reste qu'à indiquer à la machine

comment utiliser ces interfaces : c'est le rôle de la table de routage.



Chaque paquet émis possède une adresse de départ et une adresse d'arrivée. Lorsqu'une application envoie un paquet, la fonction système responsable de son acheminement parcourt la table de routage de la machine pour déterminer quelle interface elle devra utiliser.

Voici sous forme simplifiée les tables de routage des machines host01, host02 et host11.

Machine host01			
Adresse	Masque	Interface	Passerelle
127.0.0.0	255.0.0.0	lo0	néant
192.168.0.0	255.255.255.0	eth0	néant
192.168.1.0	255.255.255.0	eth1	néant

Machine host02			
Adresse	Masque	Interface	Passerelle
127.0.0.0	255.0.0.0	lo0	néant
192.168.0.0	255.255.255.0	eth0	néant
0.0.0.0	0.0.0.0	eth0	192.168.0.1

Machine host11			
Adresse	Masque	Interface	Passerelle

127.0.0.0	255.0.0.0	lo0	néant
192.168.1.0	255.255.255.0	eth0	néant
0.0.0.0	0.0.0.0	eth0	192.168.1.3

La ligne des tables de routage de host02 et host11 dont l'adresse de destination est 0.0.0.0 correspond à la route par défaut, c'est à dire à l'interface à utiliser quand aucune des autres entrées de la table ne convient.

Supposons maintenant que host02 veuille envoyer un paquet à host11. L'adresse d'arrivée est donc 192.168.1.1. host02 parcourt chaque entrée de sa table de routage et vérifie si l'adresse d'arrivée remplit les conditions indiquées.

Sur la première ligne, il applique le masque 255.0.0.0 à 192.168.1.1 et trouve donc 192.0.0.0, alors que cette ligne est donnée pour 127.0.0.0. On passe donc à la suivante: 192.168.1.1 et 255.255.255.0 donne 192.168.1.0, ce qui ne correspond pas au 192.168.0.0 attendu. Il ne reste plus que la route par défaut: le paquet sera donc transmis à host01 (192.168.0.1).

Lorsque host01 reçoit ce paquet, il applique la même technique. Sur la première ligne, le résultat est identique au cas précédent. Sur la 2ème ligne, 192.168.1.1 et 255.255.255.0 donne 192.168.1.0, ce qui ne correspond pas au 192.168.0.0 attendu. Par contre, la condition de la 3ème ligne est satisfaite: le paquet sera donc transmis sur eth1 et reçu par host11.

4. Domain Name System

4.1. Introduction

Jusqu'à maintenant, nous ne nous sommes occupé que des adresses des machines, mais dans la réalité on utilise leurs noms : pour la plupart des gens, `www.yahoo.fr` est beaucoup plus facile à retenir que `195.67.49.44`.

Toute machine peut utiliser un fichier local contenant la correspondance entre adresses IP et noms de machines. Le problème avec cette approche est qu'elle n'est valable que pour un réseau local qui ne change pas très souvent (car il faut alors mettre à jour les tables de toutes les machines!).

L'objet du Domain Name System est de fournir un moyen pour transformer les noms de machines en adresses IP à l'échelle planétaire. Le principe consiste à organiser les noms des machines de façon hiérarchique.

Ainsi, le nom `www.yahoo.fr` comporte 3 niveaux, lus de droite à gauche. Chaque niveau gère la totalité des niveaux immédiatement inférieurs. Un certain nombre de serveurs gèrent les domaines de premier niveau (ici: `fr`), ceux-ci déléguant à leur tour la gestion des niveaux inférieurs.

En résumé, les serveurs "racine" gèrent les domaines de plus haut niveau (`com`, `fr`, `uk`, etc), les Network Information Centers nationaux gérant tous les niveaux inférieurs (ex. le NIC France gère `ibp.fr`, `gouv.fr`, `yahoo.fr`, etc). Yahoo! France gère tous les noms se terminant par `yahoo.fr`, et ainsi de suite.

Lorsqu'une machine veut transformer un nom en adresse IP, elle fait appel à un serveur de noms qui se chargera d'exploiter toute l'arborescence de ses confrères pour obtenir le renseignement. Les serveurs de noms utilisent un "cache" pour éviter de déclencher une recherche à chaque demande.

Le système de délégation permet de résoudre le problème des mises à jour (à la durée de vie des informations copiées dans les caches près) et de laisser une totale liberté de gestion à chaque niveau.

Chaque serveur en charge d'un domaine doit être secondé par au moins un autre serveur. Ce sont les DNS secondaires, qui recopient à intervalles réguliers les données du serveur primaire et peuvent traiter les demandes en cas d'indisponibilité du serveur primaire.

4.2. La structure du système de nommage

Le système de nommage a une structure hiérarchique: ainsi, `www.univ-lehavre.fr` désigne le service Web (`www`) de l'université du Havre (`univ-lehavre`) en France (`fr`). Cette université possédant plusieurs machines, la France plusieurs universités et le monde plusieurs pays, on reconnaît là une structure arborescente classique.

Dans ce système, tout nœud non-terminal (c'est à dire tout nom ne désignant pas une machine) est appelé **zone**. `fr` est donc une zone, de même que `univ-lehavre.fr`. La distinction machine/zone est importante puisque ces 2 objets n'ont pas du tout les mêmes propriétés: une zone est caractérisée par des serveurs de noms, de courrier électronique et d'autres détails, alors qu'une machine est caractérisée par son adresse IP.

Le principe de fonctionnement du DNS est la délégation: chaque zone est gérée par une autorité: la zone `fr` est gérée par le NIC (Network Information Center) France, `univ-lehavre.fr` est gérée par l'université du Havre. Gérer une zone signifie tenir à jour une base de données décrivant les propriétés et le contenu de cette zone et déléguer la gestion des zones de niveau immédiatement inférieur à d'autres autorités.

Imaginons que vous vouliez connecter votre société, Grossboite SA, à Internet. En dehors des problèmes de connectivité, il vous faudrait déposer auprès du NIC le nom de zone `grossboite.fr` et lui fournir l'adresse IP d'une machine gérant la base de données de cette zone (un **serveur de nom**). Le NIC modifiera sa base de données de façon à créer cette zone et à indiquer que votre serveur de noms est responsable de sa gestion. En pratique, votre fournisseur de connectivité IP peut se charger de cette démarche.

Si par la suite votre filiale de Lyon veut se connecter, vous pourrez à votre tour créer une zone `lyon.grossboite.fr` dans votre base de données et déléguer sa gestion au serveur de noms de cette filiale. Ainsi, ses machines seront accessibles par leurs noms depuis l'Internet.

Autre notion importante: le nom canonique. Chaque machine porte un nom unique. La machine hébergeant le service Web de l'université du Havre s'appelle `dell`: ce nom permet de la désigner sans ambiguïté à partir d'une autre machine de la faculté des sciences et techniques. Pour toute autre machine connectée à l'Internet, elle n'est visible que sous le nom de `dell.fst.univ-lehavre.fr`, son **nom canonique**.

Cependant, pour une personne étrangère au service informatique de l'université, il est impossible de deviner que `dell` fait tourner le serveur Web. On est donc amené à la rendre visible sous un autre nom (`www.univ-lehavre.fr`): un **alias**. L'utilisation d'alias permet en outre de changer l'organisation physique du réseau sans changer la façon dont il est vu de l'extérieur (ex. commencer avec une seule machine pour le Web, le courrier et le FTP, puis déplacer le FTP sur une autre machine pour éviter l'asphyxie, etc).

Dernière notion: pour désigner une machine située sur le même réseau, on utilise souvent la partie de son nom précédant le point le plus à gauche (ex. `dell`), un dispositif particulier se chargeant

d'ajouter automatiquement la partie manquante (ex. fst.univ-lehavre.fr). Cette pratique est appelée **adressage relatif**, par opposition à l'**adressage absolu** consistant à donner le nom complet. Notez qu'un nom absolu n'est pas forcément canonique (ex. www.univ-lehavre.fr est un nom absolu, mais c'est un alias de dell.fst.univ-lehavre.fr).

4.3. La résolution de noms

Toute machine utilisant TCP/IP comporte un dispositif permettant la traduction de noms en adresses IP: le resolver. Quand vous fournissez à votre machine une liste d'adresses de serveurs de noms et un nom de domaine, vous configurez en fait le resolver.

Quand vous tapez une URL dans votre "brouteur" Internet, celui-ci en extrait le nom de la machine (entre "http://" et le "/" suivant) et la soumet au resolver qui lui retourne l'adresse IP correspondante. Le browser peut ensuite se connecter sur cette machine.

Pour ce faire, le resolver s'adresse au serveur de noms (le premier de la liste qui soit réceptif, si vous en avez indiqué plusieurs) et lui transmet le nom à résoudre. Le serveur de noms retourne la réponse directement s'il la connaît, ou à défaut indique où le resolver pourra la trouver, auquel cas le resolver recommence son manège avec son nouvel interlocuteur.

Pour obtenir le nom correspondant à une adresse IP, le resolver construira une requête sur une zone spéciale, in-addr.arpa, et la transmettra au serveur suivant le même principe. Par exemple, pour trouver le nom correspondant à 192.252.19.4, le resolver essaiera de résoudre 4.19.252.192.in-addr.arpa. L'adresse IP est renversée (pour respecter la même logique que les noms de domaines) et ajoutée devant in-addr.arpa.

Pour information, on appelle **requête directe** la transformation d'un nom en adresse IP et **requête inverse** la transformation d'une adresse IP en nom. Enfin, le resolver permet également d'obtenir toutes les informations d'un type donné.

4.4. Le courrier électronique

Le routage du courrier électronique est réalisé à partir des informations fournies par les serveurs de noms. Quand vous envoyez un message, votre serveur SMTP analyse l'adresse de chaque destinataire: il en extrait la partie à droite du signe @ et demande au serveur de noms quels sont les mail exchangers (MX) pour cette zone.

Un mail exchanger est un autre serveur SMTP capable de traiter le courrier à destination d'une zone. Ce traitement consiste soit à le transmettre au "facteur" (Mail Delivery Agent) qui le placera dans la bonne boîte aux lettres, soit à relayer le message vers le bon serveur si celui-ci ne peut être atteint par le serveur d'origine.

Ainsi, le serveur d'origine essayera de transmettre le message aux différents MX l'un après l'autre, par ordre décroissant de pertinence, jusqu'à ce que l'un d'eux l'accepte.

4.5. Les serveurs de noms

On distingue 2 types de serveurs de noms: les serveurs primaires et secondaires. Une zone est gérée par un serveur primaire, qui contient la base de données originale, et au moins un serveur secondaire, qui en maintient une copie.

En dehors du cas où primaire et secondaire sont branchés sur la même triplète et sur le même câble réseau (!), il est peu probable que tous deux défaillent en même temps. Ce système permet donc

d'assurer la continuité du service de noms pour une zone donnée.

La fonction du serveur de noms est de fournir à tous ceux qui le demandent les informations contenues dans sa base de données qui peut par ailleurs concerner plusieurs zones). Les informations concernant une zone sont regroupées dans un fichier (le "fichier de zone") sous la forme de **Resource Records** (RR) de différents types, selon la nature des informations. Les principaux types de RR sont:

- **SOA** (Start of authority): Indique le nom du serveur primaire, l'adresse électronique du responsable de la zone, le numéro de version du fichier de zone et différents délais (vérification du besoin de rafraîchissement (pour un serveur secondaire), délai avant réessai en cas d'échec de rafraîchissement, péremption des données en cas d'impossibilité de rafraîchissement, durée de vie minimale des RR).
- **NS** (Name server): Indique le nom canonique d'un serveur de noms. Il **DOIT** y avoir un NS par serveur (primaire ou secondaire) pour la zone.
- **MX** (Mail exchanger): Indique un serveur acceptant le courrier à destination de la zone ainsi que son degré de préférence (nombre arbitraire. Plus il est élevé par rapport aux autres MX et moins le serveur est approprié). Il **DOIT** y avoir au moins un MX (= on doit toujours pouvoir envoyer du courrier électronique, au minimum au zone-contact).
- **A** (Address): Indique l'adresse d'une machine.
- **PTR** (Pointer): Pointe sur un autre nom. Utilisé pour fournir le nom correspondant à une adresse IP: il fait alors correspondre un nom de in-addr.arpa à un nom de la zone.
- **CNAME** (Canonical name): Indique le nom canonique correspondant à un alias.

Un fichier de définition de zone doit comporter au minimum: un SOA, 2 NS (un seul suffit, mais pour des raisons administratives, on doit avoir au moins un DNS secondaire) et un MX. Chaque machine donnera lieu à un A.

Les requêtes DNS prenant beaucoup de temps (pour le client), le serveur de noms joint à ses réponses toute précision pertinente en sa possession susceptible de diminuer le nombre de requêtes nécessaires. Ainsi, si vous demandez la liste des MX records, le serveur pourra vous donner en prime les adresses IP des serveurs, ce qui évitera au client d'envoyer d'autres requêtes pour résoudre les noms de ces machines.