Chapitre 3

Les services annuaires
(DNS, DHCP et LDAP)

Dr. H. Zerrouki
UABBT, Faculté de Technologie, Département de Télécommunications

Un service d'annuaire est un service réseau qui identifie toutes les ressources d'un réseau et met ces informations à la disposition des utilisateurs ainsi que des applications. Les services d'annuaires sont importants, car ils fournissent un moyen cohérent de nommer, décrire, localiser, administrer et sécuriser les informations relatives à ces ressources et d'y accéder. Lorsqu'un utilisateur recherche un dossier partagé sur le réseau, le service d'annuaire identifie la ressource et fournit l'information à l'utilisateur.

III.1 INTRODUCTION

Les particuliers et les entreprises ont de plus en plus recours aux réseaux pour accéder à des applications distribuées et à des ressources partagées (sites web, serveurs d'applications, serveurs de fichiers, etc.).

Ces applications et ces ressources doivent interagir avec des ordinateurs situés dans le même réseau local, à travers l'intranet de l'entreprise, ou plus généralement au travers de l'Internet. Cela nécessite a priori la connaissance des adresses de ces différentes machines. Or, dans la très grande majorité des cas, on n'utilise jamais les adresses réelles des machines ; on utilise des noms.

Prenons des exemples simples. L'accès à un site web se fera par l'intermédiaire d'un nom désignant le site. L'accès à une imprimante se fera également par l'intermédiaire d'un nom désignant l'imprimante. Ces informations vont être gérées dans une base de données spéciale appelée annuaire. L'annuaire va permettre de transformer le nom du site ou le nom de l'imprimante en une adresse physique permettant aux protocoles de communication d'accéder aux équipements concernés.

De nombreux outils d'annuaires ont donc vu le jour au fil des années, offrant des services divers et variés ; certains ont périclité, d'autres sont devenus immédiatement des standards incontournables, tel **DNS** (Domain Name System).

Depuis quelques années maintenant, est apparu un nouveau standard, lui-même en passe de devenir absolument indispensable connu sous le sigle **LDAP** (Lightweight Directory Access Protocol). Ce standard ne remplacera pas DNS, ce n'est pas sa vocation, mais il permet d'unifier certains besoins tels que ceux d'annuaires de type pages blanches, d'annuaires de type **NIS** (Network Information Service), d'authentification, etc.

III.2 LES ANNUAIRES

III.2.1 Qu'est-ce qu'un annuaire ?

Un annuaire électronique peut être vu comme une base de données spécialisée, dont la fonction première est de retourner un ou plusieurs attributs d'un objet grâce à des fonctions de recherche multicritères.

Les objets peuvent être de nature très diverse. Par exemple, un objet de l'annuaire peut représenter une personne et les attributs de cet objet seront alors son nom, son prénom, son numéro de téléphone, etc. Donnons un autre exemple (déjà évoqué plus haut) : un objet représentera une imprimante et les attributs de l'objet seront alors les différents noms de cette imprimante, son adresse réseau, sa situation géographique, etc.

La figure ci-dessous illustre l'implantation d'un service d'annuaires type. Les clients se connectent au service d'annuaire pour interroger la base de données du service. Certains services d'annuaires peuvent échanger des informations avec d'autres.

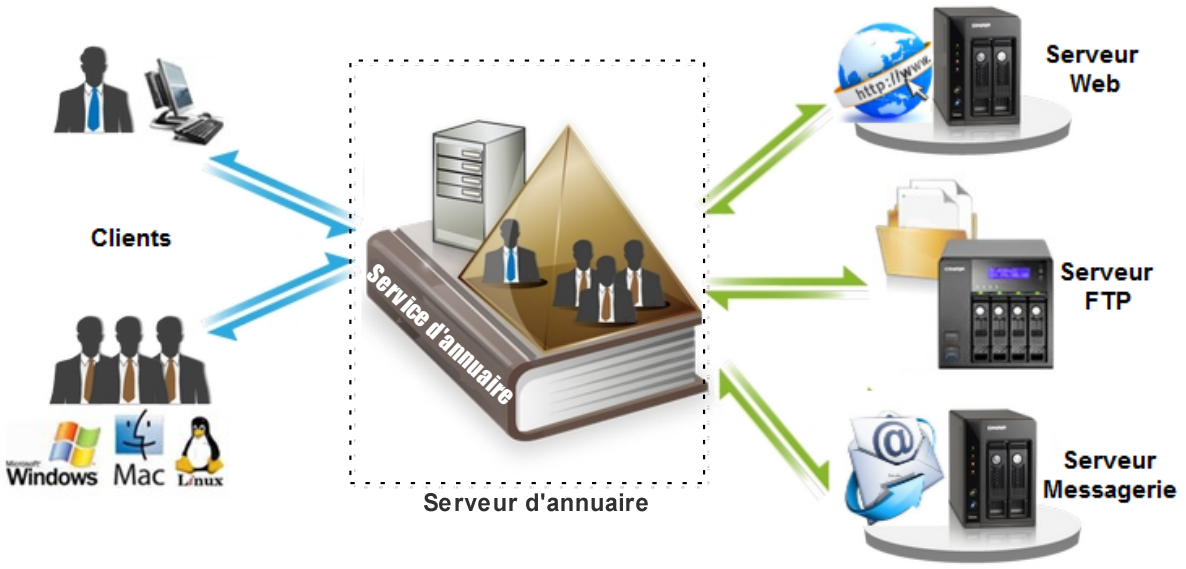


Figure III.1 – *Exemple d'implantation d'un service d'annuaires.*

Un annuaire électronique va centraliser des informations et les rendre disponibles, via le réseau, à des applications, des systèmes d'exploitation ou des utilisateurs. Il va généralement s'appuyer sur les éléments suivants :

- **Un protocole :** échange des données proprement dit et indication des opérations à effectuer sur ces dernières.
- **Un modèle fonctionnel :** description de la nature des opérations que l'on peut effectuer, comme par exemple une recherche, ou une modification.
- **Un modèle de nommage :** identification des données ; organisation des différentes entrées de l'annuaire.
- **Un modèle d'information :** nature des données pouvant être enregistrées (des chaînes de caractères, des nombres, des numéros de téléphone...).
- **Un modèle de sécurité :** description des services de sécurité permettant d'assurer par exemple le chiffrement des données transférées ou bien l'authentification du client vis-à-vis du serveur.
- **Un modèle de distribution :** création et gestion de serveurs secondaires dans un but de sauvegarde ou de répartition de charge, création et gestion de liens spéciaux (*referrals*, méta-annuaires) pointant vers des annuaires responsables d'une partie des données de l'entreprise ou vers des annuaires complètement différents.

III.2.1 Différences avec une base de données

Bien qu'un annuaire soit comparable à une base de données pour un grand nombre de fonctionnalités, il en diffère en de nombreux points.

1. Un annuaire est très performant en consultation (c'est-à-dire en lecture ou en recherche ; la lecture n'étant qu'une recherche particulière). Par contre, un annuaire n'est pas très adapté pour des mises à jour fréquentes (autrement dit en écriture). Les données contenues dans un annuaire sont en effet beaucoup plus pérennes, et il est donc totalement inutile d'optimiser les fonctions de mise à jour.
   Un annuaire doit, à l'opposé, supporter un nombre important de consultations simultanées. L'exemple le plus évident est l'annuaire électronique téléphonique. L'optimisation de la fonction « lecture » est donc à privilégier dans ce contexte.

2. Une base de données doit, par contre, généralement supporter des applications qui la remettent constamment à jour. Cela signifie que la fonctionnalité « écriture » dans une base de données est importante et doit par conséquent être optimisée. Parmi les exemples les plus classiques de bases de données, on trouve :

   - un système de réservation de billets d'avion ;
   - un système de gestion des stocks d'une grande surface de distribution ;
   - un gestionnaire de comptes bancaires.

3. Dans le cas d'un annuaire, par contre, l'accès en écriture est généralement réservé aux administrateurs de l'annuaire ou bien aux propriétaires des informations. Il n'est donc pas nécessaire que la fonction « écriture » soit optimisée ; ce type d'opération sera beaucoup plus épisodique que la lecture.

4. Un annuaire ne supporte pas bien les transactions. Une transaction est caractérisée par différentes propriétés que l'on peut résumer grâce à l'acronyme ACID : Atomicité, Cohérence, Isolation et Durabilité.

   a. **Atomicité :** une transaction doit être entièrement effectuée ou pas du tout (par exemple, lors de la mise à jour de n éléments d'une table).
   b. **Cohérence :** la cohérence entre tables d'une même base doit être respectée, même en cas d'incident.
   c. **Isolation :** pendant une transaction complexe, les autres transactions voient la totalité des données à modifier dans l'état antérieur au démarrage de la transaction jusqu'à son achèvement. Les résultats partiels ne sont donc pas accessibles, exceptées par les opérations elles-mêmes.
   d. **Durabilité :** lorsque la transaction est achevée, les modifications sont définitives, même en cas d'incident.

Le plus souvent, l'isolation n'est pas du tout garantie dans le cas d'un annuaire. On peut, par exemple, lancer une opération de modification composée de plusieurs requêtes, suivie d'une opération de recherche sur les mêmes données. Les résultats de la recherche correspondront aux données en cours de modification, et non pas avant la modification.

Ceci étant posé, il est évident que pour des raisons de facilité de mise en œuvre, les différents produits existant sur le marché vont le plus souvent faire appel à des outils de base de données pour matérialiser la base de l'annuaire proprement dit, plutôt que de mettre en œuvre un outil spécifique pour gérer cette dernière.

Partie I : Domain Name System (DNS)

Dans le monde de l'Internet, les machines du réseau sont identifiées par des adresses IP. Néanmoins, ces adresses ne sont pas très agréables à manipuler, c'est pourquoi, on utilise les noms. L'objectif a alors été de permettre la résolution des noms de domaines qui consiste à assurer la conversion entre les noms d'hôtes et les adresses IP. La solution actuelle est l'utilisation des DNS (Domain Name System).

III.3 SYSTEME DE NOM DE DOMAINE (DNS)

III.3.1 Historique

Jusqu'en 1984, sur la suite des protocoles TCP/IP, la transcription de noms d'hôtes en adresses Internet s'appuyait sur une table de correspondance maintenue par le Network Information Center (NIC), et ce dans un fichier `.txt`, lequel était transmis par FTP à tous les hôtes. Il n'était à l'époque pas compliqué de stocker les adresses puisque le nombre de machines était très réduit. Par ailleurs, avec la croissance exponentielle d'Internet il a fallu trouver une autre solution, car les problèmes se sont multipliés :

- ***La mise à jour des fichiers :*** En effet il fallait retransmettre le fichier de mise à jour à tous les hôtes, ce qui encombrait fortement la bande passante du NIC.

- ***L'autonomie des organismes :*** Avec l'évolution de l'Internet, les architectures ont été transformées, ainsi des organismes locaux ont eu la possibilité de créer leur propres noms et adresses, et ils étaient alors obligés d'attendre que le NIC prenne en compte leurs nouvelles adresses avant que les sites ne puissent être visibles par tous sur Internet. Le souhait était alors que chacun puisse gérer ses adresses avec une certaine autonomie.

Tous ces problèmes ont fait émerger des idées sur l'espace des noms et sa gestion. Les propositions ont été diverses, mais l'une des tendances émergentes a été celle d'un espace de noms hiérarchisé, et dont le principe hiérarchique s'appuierait autant que possible sur la structure des organismes eux-mêmes, et où les noms utiliseraient le caractère "." pour marquer la frontière entre deux niveaux hiérarchiques.

En 1983-1984, *Paul Mockapetris* et *John Postel* proposent et développent une solution qui utilise des structures de base de données distribuée : les Domain Name System devenue obsolète. Les spécifications des DNS ont été établies en 1987.

III.3.2 Définition de DNS

DNS (*Domain Name System*, système de noms de domaine) est une base de données distribuée hiérarchisée qui contient les mappages de noms d'hôtes DNS à des adresses IP. Le système DNS est utilisé dans les réseaux TCP/IP tels qu'Internet pour localiser des ordinateurs et des services à l'aide de noms conviviaux.

Lorsqu'un utilisateur entre un nom DNS dans une application, les services DNS peuvent résoudre ce nom en une autre information qui lui est associée, par exemple une adresse IP.

DNS permet également de découvrir des services réseau comme des serveurs de messagerie et des contrôleurs de domaine dans le service d'annuaire Active Directory.

III.3.3 Fonction de DNS

DNS est à la base du système de noms Internet, mais aussi du système de noms de domaine Active Directory d'une organisation. Il prend en charge l'accès aux ressources à l'aide de noms alphanumériques. Sans DNS, vous devriez trouver les adresses IP des ressources pour accéder à ces ressources. Comme les adresses IP des ressources peuvent changer, il serait difficile d'en tenir à jour une liste exacte. Au lieu de cela, DNS permet aux utilisateurs de faire appel à des noms alphanumériques, lesquels restent assez stables dans une organisation.

Avec DNS, les noms d'hôtes résident dans une base de données qui peut être distribuée entre plusieurs serveurs, ce qui diminue la charge de chaque serveur et permet d'administrer le système de noms par partitions. DNS prend en charge des noms hiérarchiques et permet d'inscrire divers types de données en plus du mappage de noms d'hôtes à adresse IP qui est utilisé dans les fichiers Hosts.

Comme la base de données DNS est distribuée, sa taille est illimitée et l'ajout de serveurs ne dégrade guère ses performances.

L'illustration suivante représente une utilisation élémentaire de DNS qui consiste à trouver l'adresse IP d'un ordinateur à partir de son nom.

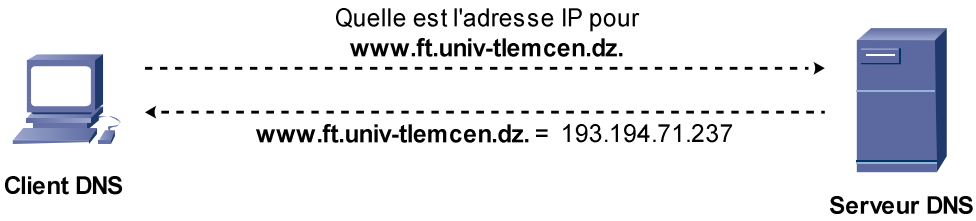


Figure III.2 – *Exemple d'une utilisation élémentaire de DNS.*

Dans cet exemple, un ordinateur client interroge un serveur DNS pour lui demander l'adresse IP d'un troisième ordinateur configuré pour utiliser le nom de domaine DNS **www.ft.univ-tlemcen.dz**. Le serveur DNS étant en mesure de répondre à cette requête en interrogeant sa base de données locale, il renvoie une réponse qui fournit l'information demandée, c'est-à-dire un enregistrement de ressource A (adresse d'hôte) contenant l'adresse IP correspondant à **www.ft.univ-tlemcen.dz**.

Cet exemple illustre une requête DNS simple entre un client unique et un serveur DNS. En pratique, les requêtes DNS sont souvent plus complexes que celle-ci et comprennent des étapes supplémentaires.

III.4 ESPACE DE NOMS DE DOMAINES

III.4.1 Noms de domaines DNS

Le système de nom de domaine (DNS, Domain Name System) a été initialement défini dans les RFC (*Request For Comments*) 1034 et 1035. Ces documents spécifient les éléments communs à toutes les implémentations des logiciels DNS, qui comprennent entre autres :

- **Un espace de noms de domaines DNS**, qui définit une structure hiérarchique des domaines permettant d'organiser les noms.
- **Des enregistrements de ressources**, qui mappent les noms de domaines DNS sur un type spécifique d'informations de ressources et sont utilisés lorsque le nom est inscrit ou résolu dans l'espace de noms.
- **Des serveurs DNS**, qui stockent les requêtes de noms portant sur des enregistrements de ressources et y répondent.

- **Des clients DNS**, également appelés solveurs, qui demandent aux serveurs de rechercher et de convertir les noms en un type d'enregistrement de ressource spécifié dans la requête.

III.4.2 Présentation de l'espace de noms de domaines DNS

L'espace de noms de domaines DNS, illustré dans la figure ci-dessous, repose sur le concept d'arborescence des domaines nommés. Chaque niveau de l'arborescence représente une branche ou une feuille de cet arbre. Une branche est un niveau dans lequel plusieurs noms sont utilisés pour identifier un ensemble de ressources nommées. Une feuille représente un nom unique utilisé une seule fois à ce niveau pour identifier une ressource spécifique.

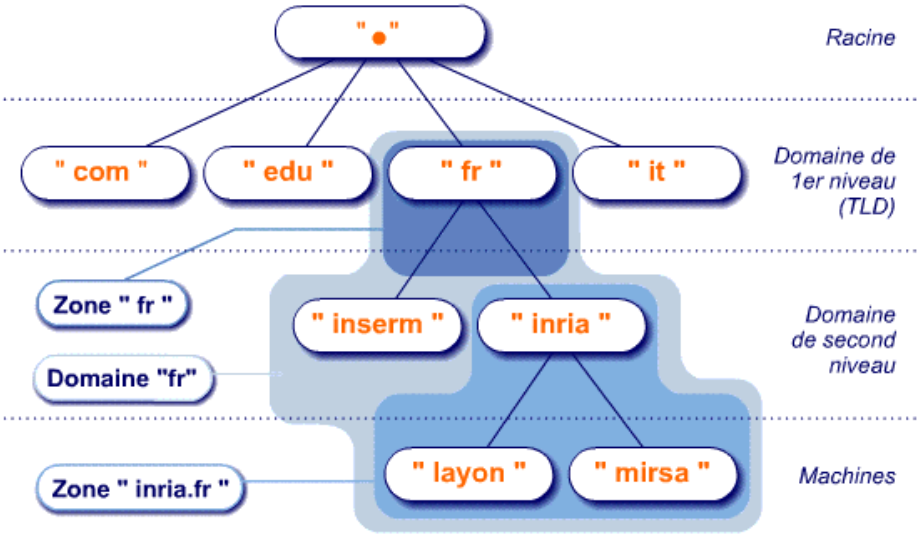


Figure III.3 – *Structure arborescente de l'espace de noms de domaines DNS.*

La figure précédente montre de quelle manière les serveurs racine Internet confèrent à Microsoft l'autorité sur sa propre partie de l'arborescence des espaces de noms de domaines sur Internet. Les clients et les serveurs DNS utilisent des requêtes pour convertir les noms de l'arborescence en types spécifiques d'informations de ressources. Ces informations sont fournies par les serveurs DNS dans les réponses aux requêtes des clients DNS, qui extraient ensuite ces informations et les transmettent à un programme demandeur pour résoudre le nom faisant l'objet de la requête.

Dans le processus de résolution d'un nom, n'oubliez pas que les serveurs DNS fonctionnent souvent comme des clients DNS qui interrogent d'autres serveurs afin de résoudre complètement une requête de nom.

III.4.3 Structure arborescente de l'espace de noms

L'espace de nom est une arborescence de nœuds. Elle permet l'indexation sur les noms de domaines.

La racine de cet arbre est noté « **.** », nommé *dot*. Les nœuds situés à la profondeur 1 de l'arbre sont appelés les domaines *Top-Level Domain* (ou *root-level*) : les TLDs. On distingue à ce niveau deux groupes :

- un groupe dit de ***forward mapping*** qui met en correspondance les noms d'hôtes avec les adresses IP. Ce groupe est représenté par des fichiers de données appelés fichier de ***forward zone***.
- un groupe de ***reverse mapping*** qui associe les adresses IP aux noms de machine. Ce groupe est représenté par des fichiers de ***reverse zone***.

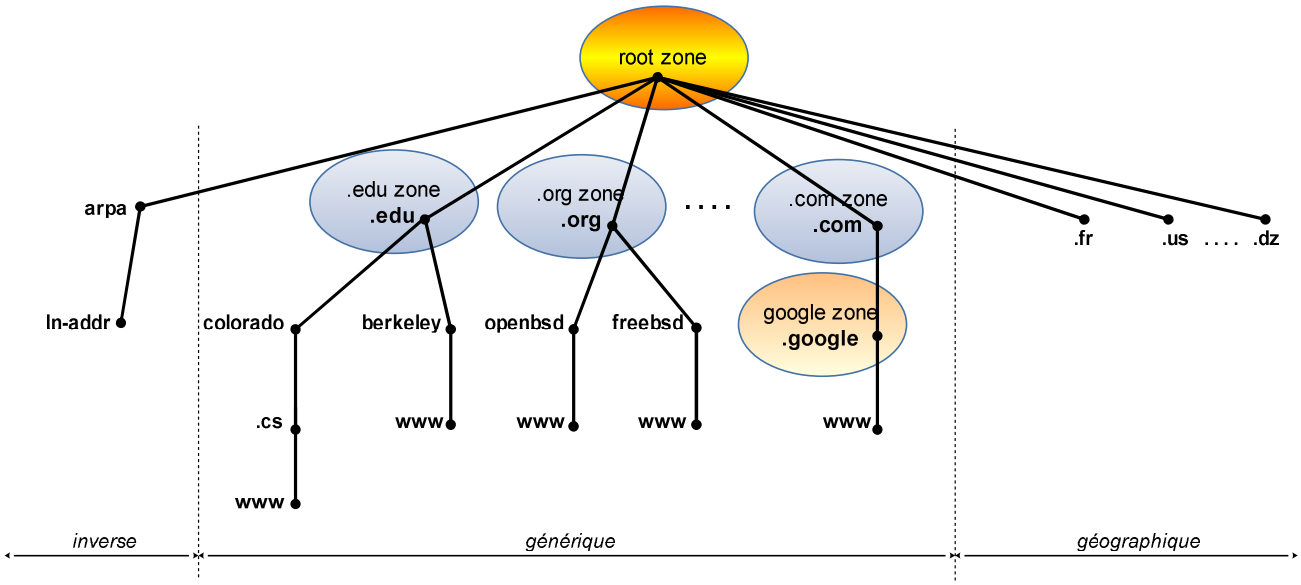


Figure III.4 – *Hiérarchie du DNS, domaines et zones.*

Le DNS utilise la gestion hiérarchique des noms, et pour des raisons essentiellement historiques, On distingue deux domaines pour le classement des noms.

a) *Les domaines géographiques*

En général les domaines géographiques sont directement maintenus par des organismes nationaux, l'AFNIC pour : USA (**us**), Royaume Uni (**uk**), la France (**fr**), Chine (**cn**) et l'Algérie (**dz**) par exemple.

b) *Les domaines génériques*

Cette liste est définie par la Rfc 1591 - Domain Name System Structure and Delegation

.com - Commerciaux
.edu - Organismes d'éducation américaine
.net - Organismes de gestion de réseaux
.org - Organismes non-commerciaux
.int - Organismes internationaux
.gov - Organismes gouvernementaux USA
.mil - Organismes militaires USA
.arpa - Transition ARPAnet-> Internet + traduction inverse

Chaque nœud est étiqueté par un composant du nom de domaine, avec les règles suivantes :

- Un nom de domaine est limité à 63 caractères au maximum.
  - *a priori* insensible à la casse, cependant les implémentations ont tendance à tenir compte de la casse.
  - constitués de caractères alphanumériques et du **-** (tiret [*hyphen*]).

Les nœuds d'un même niveau doivent être uniques (sauf au niveau des feuilles), ce qui garantit un espace sans collision. Enfin, les feuilles (nœuds terminaux) sont elles aussi étiquetées par des noms de domaines (bien que souvent qualifiés de nom d'hôte).

Un nom de domaine totalement qualifié (FQDNs : *Fully-Qualified Domain Names*) est la concaténation des nœuds constituant un chemin dans l'arbre, d'origine une feuille et d'extrémité la racine. Chaque composant est séparé de l'autre par un point (« **.** »).

III.4.3 Mode d'organisation de l'espace de noms de domaines DNS

Tout nom de domaine DNS utilisé dans l'arborescence est du point de vue technique un domaine. Cependant, les noms sont généralement identifiés de cinq manières, selon leur niveau et leur mode d'utilisation courant. Par exemple, le nom de domaine DNS inscrit au nom de Microsoft (microsoft.com) est appelé un domaine de second niveau. En effet, ce nom est formé de deux parties (appelées étiquettes) qui indiquent que le domaine est situé deux niveaux en dessous de la racine ou du premier niveau de l'arborescence. La plupart des noms de domaines DNS comporte deux étiquettes ou plus, chacune indiquant un niveau de l'arborescence. Les points sont utilisés pour séparer les étiquettes.

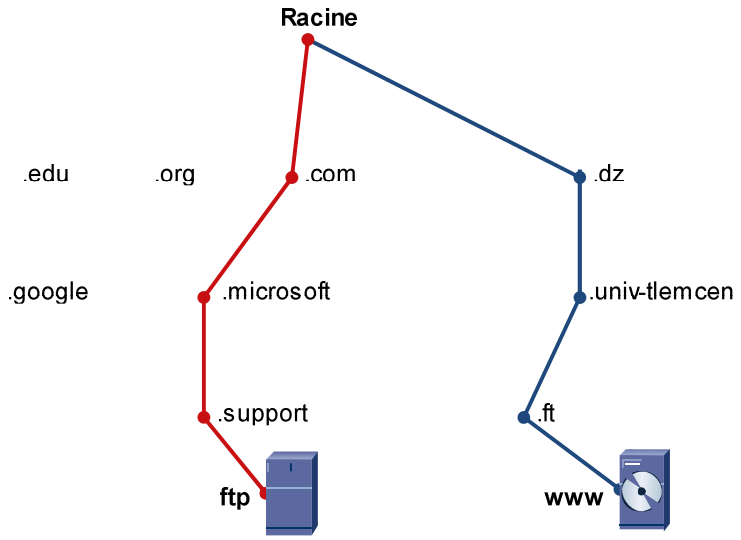


Figure III.5 – *Exemple de mode d'organisation de l'espace de noms.*

Outre les domaines de second niveau, d'autres termes utilisés pour décrire les noms de domaines DNS par leur fonction dans l'espace de noms sont décrits comme suite :

III.4.3.1 La racine du domaine

Il s'agit de la cime de l'arborescence, représentant un niveau non nommé. Elle est parfois affichée sous la forme de deux guillemets vides (""), indiquant une valeur nulle. Lorsqu'elle est utilisée dans un nom de domaine DNS, elle est indiquée par un point à droite (**.**) nommé *dot*, pour indiquer que le nom est situé à la racine ou niveau supérieur de la hiérarchie du domaine.

Le nom de domaine DNS est alors considéré comme complet et désigne un emplacement exact de l'arborescence des noms. Les noms exprimés de cette manière sont appelés des noms de domaines complets (FQDN, Fully Qualified Domain Names). Un point (**.**) utilisé seul ou à la fin d'un nom, tel que dans « **ft.univ-tlemcen.dz.** ».

III.4.3.2 Domaine de premier niveau

Nom de deux ou trois lettres utilisé pour indiquer un pays/une région ou le type d'organisation utilisant un nom. « **.dz** », qui indique un nom inscrit au nom d'une entreprise pour une utilisation commerciale sur Internet.

III.4.3.3 Domaine de second niveau

Noms de longueur variable inscrits au nom d'un individu ou d'une organisation pour une utilisation sur Internet. Ces noms sont toujours associés à un domaine de premier niveau approprié, selon le type d'organisation ou l'emplacement géographique dans lequel un nom est utilisé. « **univ-tlemcen.dz** », qui est le nom de domaine de second niveau inscrit au nom de Microsoft par le Registre des noms de domaines DNS d'Internet.

III.4.3.3.4 Sous-domaine

Noms supplémentaires pouvant être créés par une organisation et associés au nom de domaine de second niveau inscrit. Ils comprennent les noms ajoutés pour développer l'arborescence DNS des noms dans une organisation et la diviser en services ou en emplacements géographiques. « **ft.univ-tlemcen.dz.** », qui est un sous-domaine fictif défini par Microsoft pour être utilisé dans les exemples de noms de la documentation.

III.4.3.5 Nom d'hôte ou de ressource

Noms qui représentent une feuille de l'arborescence DNS des noms et identifient une ressource spécifique. Généralement, l'étiquette la plus à gauche d'un nom de domaine DNS identifie un ordinateur spécifique du réseau. Par exemple, si un nom situé à ce niveau est utilisé dans un enregistrement de ressource (RR) hôte (A), il doit être utilisé pour rechercher l'adresse IP de l'ordinateur en fonction de son nom d'hôte. « **www.ft.univ-tlemcen.dz.** », où la première étiquette (« www ») est le nom d'hôte DNS d'un ordinateur spécifique du réseau.

III.4.4 Type de serveurs et autorités

Par le découpage en zone on a donc trois types de serveurs de noms.

III.4.4.1 Le serveur primaire

Le serveur primaire est serveur d'autorité sur sa zone : il tient à jour un fichier appelé "fichier de zone", qui établit les correspondances entre les noms et les adresses IP des hosts de sa zone. Chaque domaine possède un et un seul serveur primaire.

III.4.4.2 Le serveur secondaire

Un serveur de nom secondaire obtient les données de zone via le réseau, à partir d'un autre serveur de nom qui détient l'autorité pour la zone considérée. L'obtention des informations de zone via le réseau est appelé transfert de zone. Il est capable de répondre aux requêtes de noms IP (partage de charge), et de secourir le serveur primaire en cas de panne. Le nombre de serveurs secondaires par zone n'est pas limité. Ainsi il y a une redondance de l'information. Le minimum imposé est un serveur secondaire et le pré requis mais pas obligatoire est de le situer sur un segment différent du serveur primaire.

Un serveur qui effectue un transfert de zone vers un autre serveur est appelé serveur maître. Un serveur maître peut être un serveur primaire ou un serveur secondaire. Un serveur secondaire peut disposer d'une liste de serveurs maîtres (jusqu'à dix serveurs maîtres). Le serveur secondaire contacte successivement les serveurs de cette liste, jusqu'à ce qu'il ait pu réaliser son transfert de zone.

III.4.4.3 Le serveur cache

Le serveur cache ne constitue sa base d'information qu'à partir des réponses des serveurs de noms. Il inscrit les correspondances nom / adresse IP dans un cache avec une durée de validité limitée (TTL) ; il n'a aucune autorité sur le domaine : il n'est pas responsable de la mise à jour des informations contenues dans son cache, mais il est capable de répondre aux requêtes des clients Dns.

De plus on peut distinguer les serveurs racine : ils connaissent les serveurs de nom ayant autorité sur tous les domaines racine. Les serveurs racine connaissent au moins les serveurs de noms pouvant résoudre le premier niveau (.com, .edu, .fr, etc.) C'est une pierre angulaire du système DNS : si les serveurs racine sont inopérationnels, il n'y a plus de communication sur l'Internet, d'où multiplicité des serveurs racines (actuellement il y en a 14). Chaque serveur racine reçoit environ 100 000 requêtes par heure.

Un serveur de nom, en terme de physique, peux très bien jouer le rôle de plusieurs de ces fonctions. On trouvera par exemple, beaucoup d'entreprise qui héberge leurs domaines sur le serveur DNS primaire servant aussi de cache pour les requêtes sortantes des utilisateurs interne.

III.4.5 La diffusion des modifications " Le transfert de zones "

Pour chaque zone DNS, le serveur servant de référence est le DNS maître ou DNS primaire. Les DNS esclaves ou secondaires servant cette zone vont récupérer les informations du DNS maître. Cette récupération d'information est appelée transfert de zone. Seuls les DNS secondaires ont besoin d'être autorisés à effectuer cette opération, mais assez souvent aucune restriction n'est présente. Ceci permettant à n'importe qui de se connecter via **`nslookup`** et d'utiliser l'argument **`ls -d`** permettant l'affichage du contenu d'une zone.

Lorsque des changements apparaissent sur une zone, il faut que tous les serveurs qui gèrent cette zone en soient informés. Les changements sont effectués sur le serveur principal, le plus souvent en éditant un fichier. Après avoir édité le fichier, l'administrateur signale au serveur qu'une mise à jour a été effectuée, le plus souvent au moyen d'un signal (SIGINT). Les serveurs secondaires interrogent régulièrement le serveur principal pour savoir si les données ont changé depuis la dernière mise à jour. Ils utilisent un numéro constitué de la date au format américain: année, mois, jour; version du jour, il est donc toujours incrémenté.

III.4.6 Les pannes

Lorsqu'un serveur primaire est indisponible, le serveur secondaire ne reçoit pas de réponse à ses interrogations sur le numéro de version du fichier de zone. Il continue ses tentatives jusqu'à expiration de la validité des enregistrements de son fichier de zone ('Expire Time'). Lorsqu'un serveur primaire redevient disponible, aucun mécanisme de synchronisation entre le fichier de zone des serveurs secondaires et celui du serveur primaire n'a été normalisé.

III.5 RECHERCHE DE RESSOURCES

III.5.1 Les Résolveurs

Les "résolveurs" sont des programmes qui interfacent les applications utilisateur aux serveurs de noms de domaines. En effet, ce n'est pas l'utilisateur qui effectue les requêtes directement. Dans le cas le plus simple, un résolveur reçoit une requête provenant d'une application (ex., Applications de courrier électronique, Telnet, FTP) sous la forme d'un appel d'une fonction de bibliothèque, d'un appel système etc., et renvoie une information sous une forme compatible avec la représentation locale de données du système.

Le résolveur est situé sur la même machine que l'application recourant à ses services, mais devra par contre consulter des serveurs de noms de domaines sur d'autres hôtes. Comme un résolveur peut avoir besoin de contacter plusieurs serveurs de noms, ou obtenir les informations directement à partir

de son cache local, le temps de réponse d'un résolveur peut varier selon de grandes proportions, depuis quelques millisecondes à plusieurs secondes.

L'une des raisons les plus importantes qui justifient l'existence des résolveurs est d'éliminer le temps d'acheminement de l'information depuis le réseau, et de décharger simultanément les serveurs de noms, en répondant à partir des données cachées en local. Il en résulte qu'un cache partagé entre plusieurs processus, utilisateurs, machines, etc., sera incomparablement plus efficace qu'une cache non partagé.

III.5.2 Les Requêtes DNS

La principale activité d'un serveur de noms est de répondre aux requêtes standards. La requête et sa réponse sont toutes deux véhiculées par un message standardisé. La requête contient des champs QTYPE, QCLASS, et QNAME, qui décrivent le(s) type(s) et les classes de l'information souhaitée, et quel nom de domaine cette information concerne. Les requêtes sont des messages envoyés aux serveurs de noms en vue de consulter les données stockées par le serveur. Par exemple avec Internet, on peut utiliser aussi bien UDP que TCP pour envoyer ces requêtes.

III.5.2.1 Structure des requêtes

Parmi les champs fixes on trouve 4 bits très importants appelé code d'opération (OPCODE). Le code d'opération permet de donner des informations sur la nature du message (requête, réponse, ...). Les quatre possibilités sont :

Question, Contient la question (nom d'hôte ou de domaine sur lequel on cherche des renseignements et type de renseignements recherchés).

Answer, Contient les RRs (Ressource Records) qui répondent à la question.

Authority, Contient des RRs qui indiquent des serveurs ayant une connaissance complète de cette partie du réseau.

Additional, Contient des RRs supplémentaires pouvant être utiles pour exploiter les informations contenues dans les autres sections.

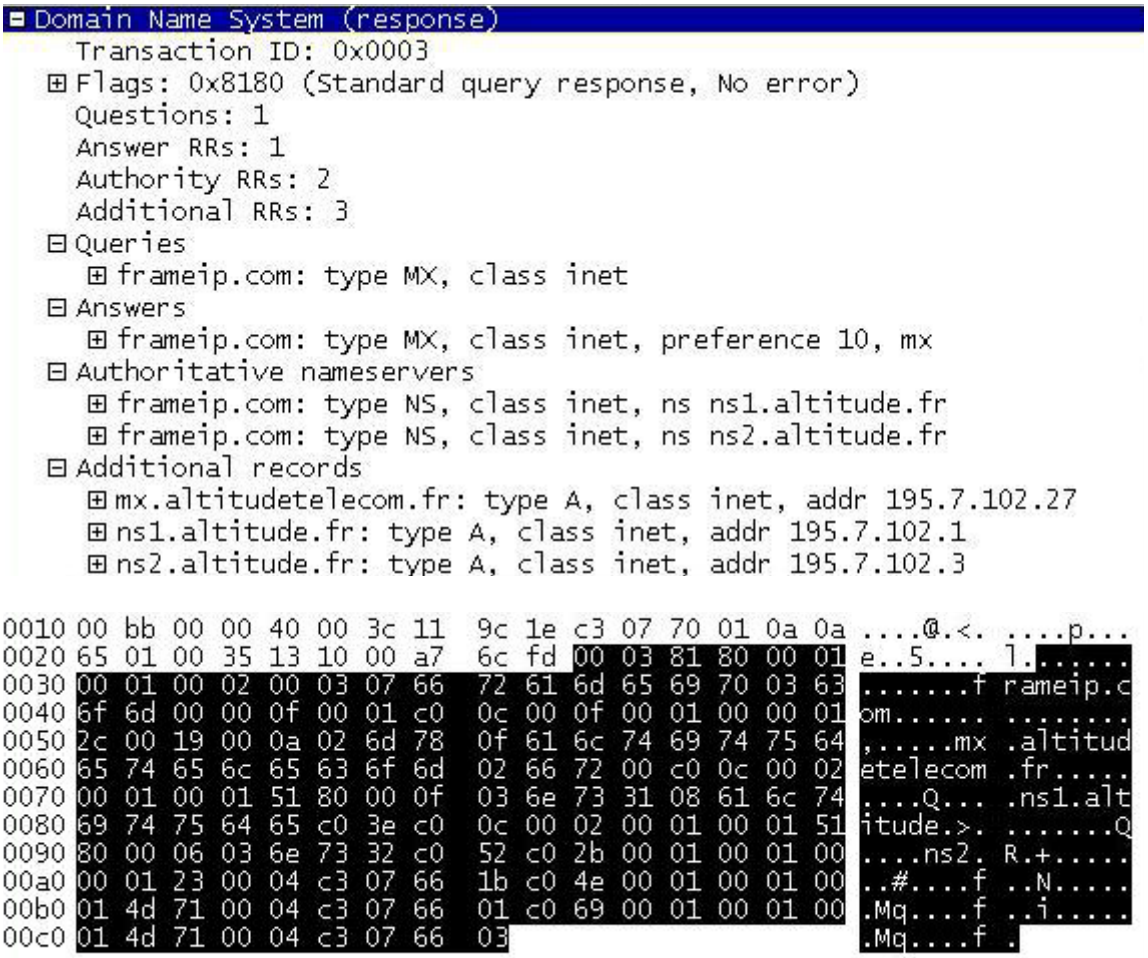
Voici un exemple de requête où l'on souhaite connaître le nom du serveur de courrier s'occupant de frameip.com :

Header	OPCODE=SQUERY
Question	QNAME=ISI.EDU., QCLASS=IN, QTYPE=MX
Answer	vide
Authotity	vide
Additionnal	vide

```
⊟Domain Name System (query)
    Transaction ID: 0x0003
  ⊞ Flags: 0x0100 (Standard query)
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
0000 00 00 0c 8a 6b 30 00 00  00 00 00 01 08 00 45 00 ....k0.. ......E.
0010 00 39 d8 03 00 00 80 11  c0 9c 0a 0a 65 01 c3 07 .9...... ....e...
0020 70 01 13 10 00 35 00 25  75 d9 00 03 01 00 00 01 p....5.% u.......
0030 00 00 00 00 00 00 07 66  72 61 6d 65 69 70 03 63 .......f rameip.c
0040 6f 6d 00 00 0f 00 01                             om.....
```

La réponse obtenue est :

Header	OPCODE=SQUERY, RESPONSE, AA
Question	QNAME=ISI.EDU., QCLASS=IN, QTYPE=MX
Answer	ISI.EDU MX 10 VENERA.ISI.EDU MX 10 VAXA.ISI.EDU
Authotity	vide
Additionnal	VENERA.ISI.EDU A 128.9.0.32 A 10.1.0.52 VAXA.ISI.EDU A 10.2.0.27 A 128.9.0.33



III.5.2.2 Le mode Itératif

Ce mode est le plus simple du point de vue du serveur. Les serveurs répondent directement à la requête sur la base seule de ses informations locales. La réponse peut contenir la réponse demandée, ou bien donne la référence d'un autre serveur qui sera "plus susceptible " de disposer de l'information demandée. Il est important que tous les serveurs de noms puissent implémenter ce mode itératif et désactive la fonction de récursivité.

Les avantages d'une résolution itérative :

- Dans le cas d'une implémentation simplifiée d'un résolveur qui ne sait exploiter d'autres réponses qu'une réponse directe à la question.
- Dans le cas d'une requête qui doit passer à travers d'autres protocoles ou autres "frontières" et doit pouvoir être envoyée à un serveur jouant le rôle d'intermédiaire.
- Dans le cas d'un réseau dans lequel intervient une politique de cache commun plutôt qu'un cache individuel par client.

Le service non-récursif est approprié si le résolveur est capable de façon autonome de poursuivre sa recherche et est capable d'exploiter l'information supplémentaire qui lui est envoyée pour l'aider à résoudre son problème.

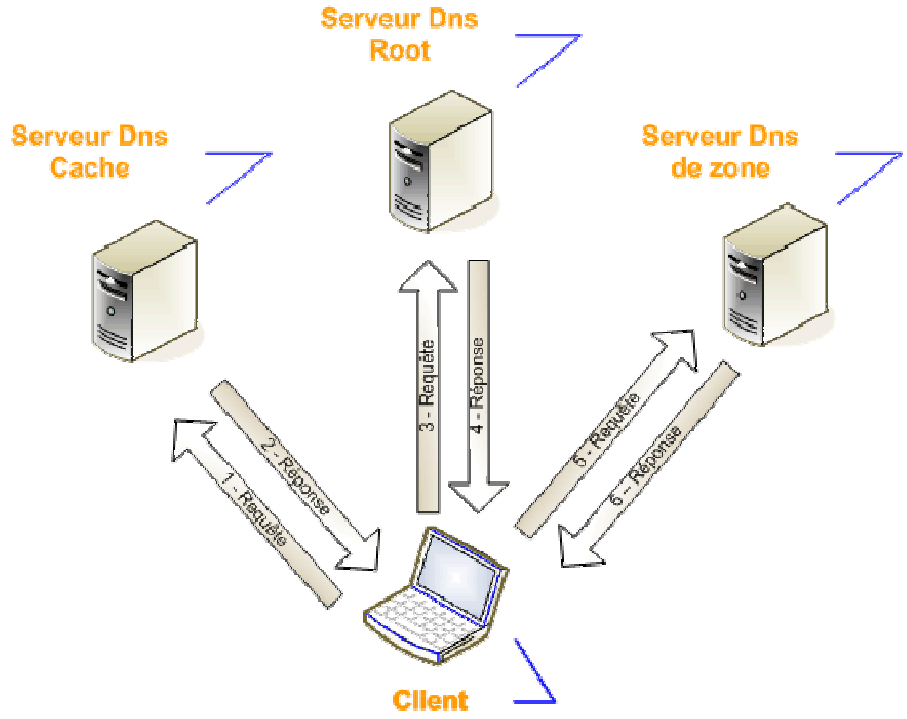


Figure III.6 – *Le mode Itératif.*

III.5.2.3 Le mode Récursif

Le mode récursif une fois est plus simple du point de vue du client. Dans ce mode, le premier serveur prend le rôle de résolveur.

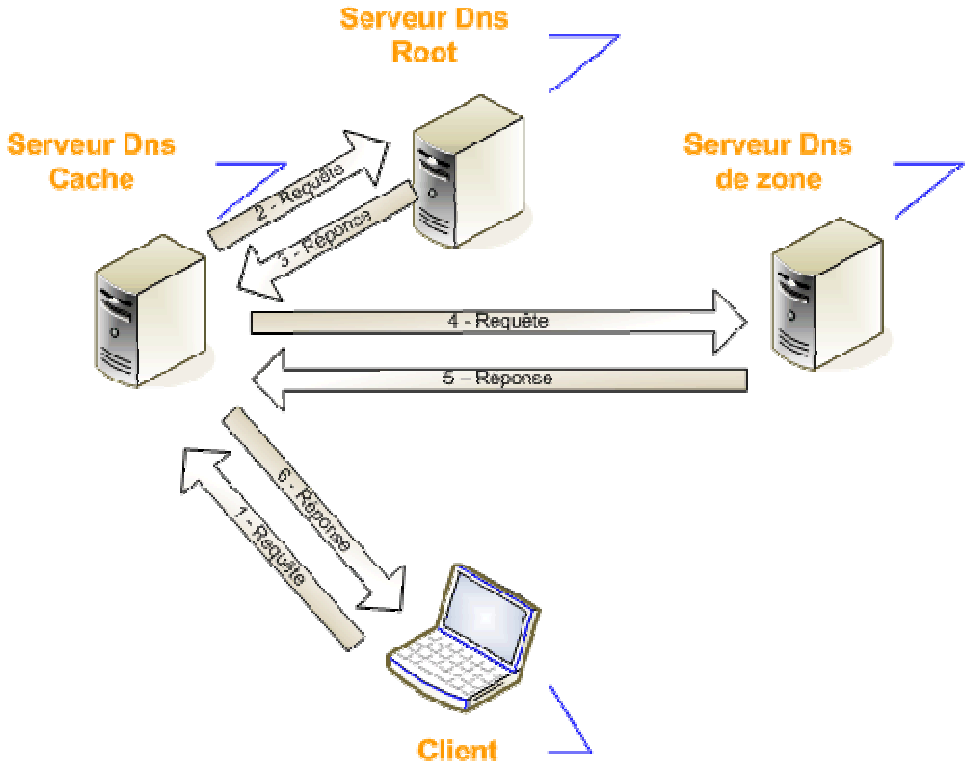


Figure III.7 – *Le mode Récursif.*

L'utilisation du mode récursif est limitée aux cas qui résultent d'un accord négocié entre le client et le serveur. Cet accord est négocié par l'utilisation de deux bits particuliers des messages de requête et de réponse :

Le bit **RA** (Récursion Admissible), est marqué ou non par le serveur dans toutes les réponses. Ce bit est marqué si le serveur accepte à priori de fournir le service récursif au client, que ce dernier l'ait demandé ou non. Autrement dit, le bit RA signale la disponibilité du service plutôt que son utilisation.

Les requêtes disposent d'un bit **RD** (pour "Récursion Désirée"). Ce bit indique que le requérant désire utiliser le service récursif pour cette requête. Les clients peuvent demander le service récursif à n'importe quel serveur de noms, bien que ce service ne puisse leur être fourni que par les serveurs qui auront déjà marqué leur bit RA, ou des serveurs qui auront donné leur accord pour ce service par une négociation propriétaire ou tout autre moyen hors du champ du protocole DNS.

Le mode récursif est mis en œuvre lorsqu'une requête arrive avec un bit RD marqué sur un serveur annonçant disposer de ce service, le client peut vérifier si le mode récursif a été utilisé en constatant que les deux bits Ra et Rd ont été marqués dans la réponse.

Notez que le serveur de noms ne doit pas utiliser le service récursif s'il n'a pas été explicitement demandé par un bit RD, car cela interfère avec la maintenance des serveurs de noms et de leurs bases de données. Lorsque le service récursif est demandé et est disponible, la réponse récursive à une requête doit être l'une des suivantes :

- La réponse à la requête, éventuellement préfacée par un ou plusieurs RR CNAME qui indiquent les alias trouvés pendant la recherche de la réponse.
- Une erreur de nom indiquant que le nom demandé n'existe pas. Celle-ci peut inclure des RR CNAME qui indiquent que la requête originale pointait l'alias d'un nom qui n'existe pas.
- Une indication d'erreur temporaire.

Si le service récursif n'est pas requis, ou n'est pas disponible, la réponse non-récursive devra être l'une des suivantes :

- Une réponse d'erreur "autorisée" indiquant que le nom n'existe pas.
- Une indication temporaire d'erreur.
- Une combinaison :
  - Des RR qui répondent à la question, avec indication si les données sont extraites d'une zone ou d'un cache.
  - D'une référence à un serveur de noms qui gère une zone plus "proche" du nom demandé que le serveur qui a été contacté.

Les RR que le serveur de nom pense être utile au requérant pour continuer sa recherche.

III.5.2.4 Exemple de résolution de noms

Nous allons voir avec un exemple comment se fait le parcours de l'arborescence pour la résolution de noms. On prend par exemple l'adresse suivante : www.univ-tlemcen.dz Il faut alors :

- Trouver le NS de la racine
- Interroger pour trouver le NS des .dz
- Poser la question finale au NS de univ-tlemcen.dz qui identifiera l'entrée www

III.5.3 Les Requêtes inverses

III.5.3.1 Fonctionnement

Dans le cas d'une requête inverse, le solveur envoie une demande à un serveur de noms afin que celui-ci renvoie le nom d'hôte associé à une adresse IP connue. C'est utile surtout pour des questions de sécurité, pour savoir avec qui on échange. La mise en place de la résolution inverse est un peu plus compliquée, car l'adressage par nom est basé sur la notion de domaine qui souvent n'a rien à voir avec la structure des adresses IP. Par conséquent, seule une recherche approfondie portant sur tous les domaines peut garantir l'obtention d'une réponse exacte. Deux moyens existent pour convertir une adresse IP en nom d'hôte : l'usage de requêtes Dns inversées (Au sens Opcode=Iquery où Iquery = 1) ou les requêtes Dns de type PTR (Classe IN et Opcode=Query).

En effet, dans le premier cas, on envoie un message Dns contenant une réponse et on demande toutes les questions pouvant conduire à cette réponse, alors que les requêtes PTR posent la question de façon explicite : Qui est l'adresse a.b.c.d ?

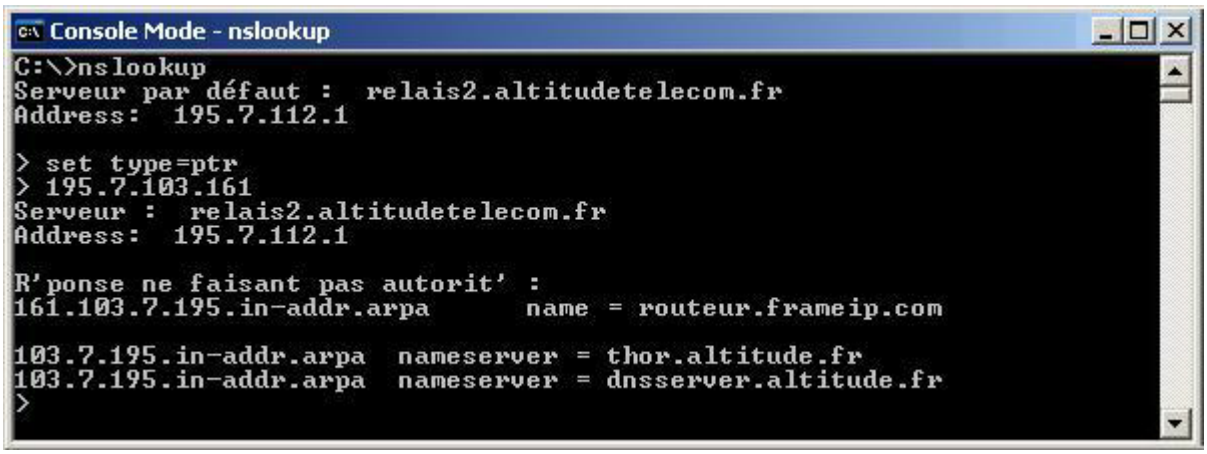
Une requête DNS inversée a la particularité d'avoir le champ Question vide, et de contenir une entrée dans le champ Answer. Pour que le serveur DNS comprenne le sens de la requête, le champ Opcode des en-têtes du message DNS doit être à la valeur Iquery. Voici une représentation :

```
Header       OPCODE=IQUERY, ID=997
Question     "EMPTY"
Answer       "ANYNAME" A IN 10.1.0.52
Authotity    "EMPTY"
Additionnal  "EMPTY"
```

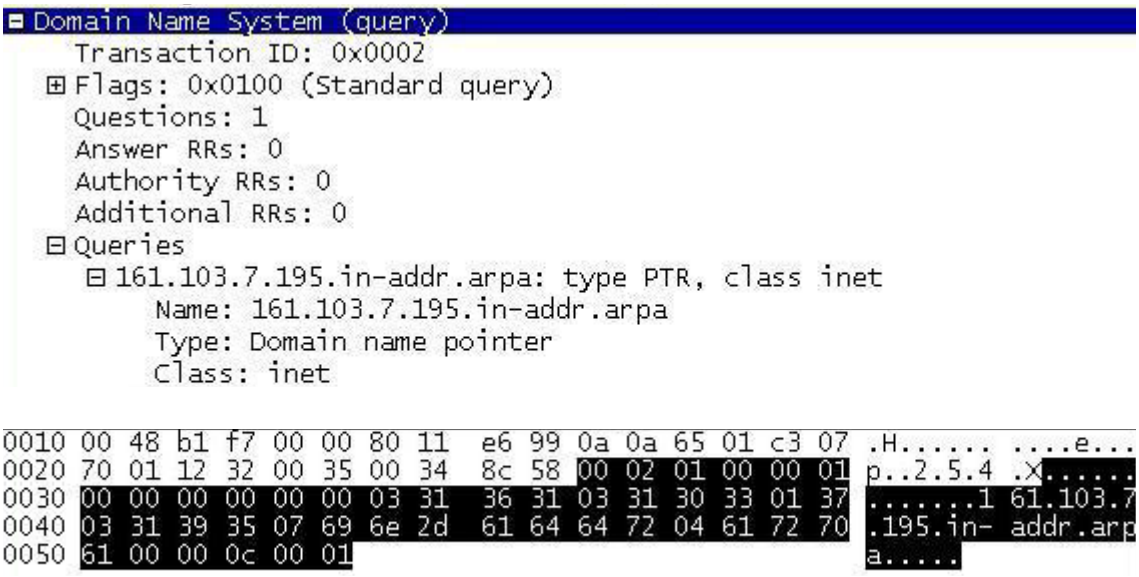
Pour répondre aux requêtes inverses en évitant des recherches exhaustives dans tous les domaines, un domaine spécial appelé **in-addr.arpa** a été créé. Une fois le domaine in-addr.arpa construit, des enregistrements de ressources spéciaux sont ajoutés pour associer les adresses IP aux noms d'hôte qui leur correspondent. Il s'agit des enregistrements pointeurs (PTR), ou enregistrements de références.

Par exemple pour connaître le nom de la machine dont l'adresse est 137.194.206.1, on envoie une requête dont la question contient QNAME=1.206.194.137.IN-ADDR.ARPA.

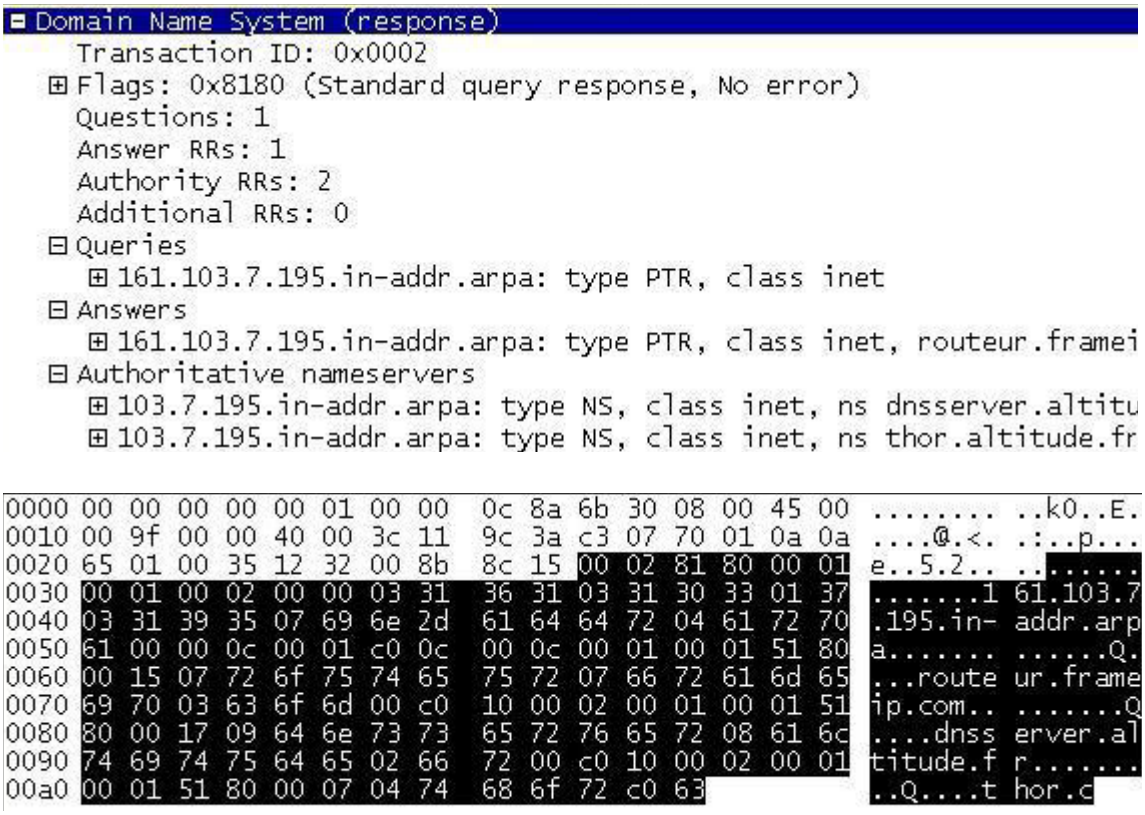
Exp. : Ligne de commande permettant d'établir la requête.



Capture du datagramme Query



Capture du datagramme Answer :



III.6 FORMAT D'UN MESSAGE DNS

III.6.1 Le transport

a) Utilisation d'UDP

Le port serveur utilisé pour l'envoi des datagrammes en UDP est 53. Les datagrammes DNS en UDP sont limités à 512 octets (valeur représentant les données sans l'entête UDP et IP). Les datagrammes plus longs doivent être tronqués à l'aide du champ TC.

L'utilisation d'UDP n'est pas recommandée pour les transferts de zone, mais uniquement pour les requêtes standards.

b) Utilisation de TCP

Le port serveur utilisé pour l'envoi des datagrammes en TCP est 53. Le datagramme inclus alors un champ de deux octets nommé "longueur", il permet de spécifier la longueur totale des données indépendamment de la fragmentation. La longueur est calculée sans les 2 octets de ce même champ.

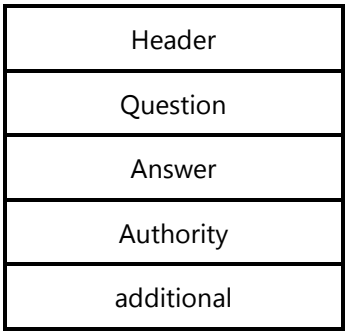


Figure III.8 – *Format d'un message DNS*

Nous décrivons l'entête dans la figure III.9. Les quatre sections suivantes ne contiennent pas nécessairement de données.

- Pour les requêtes seules la section ***Question*** en possède.
- Les réponses aux requêtes de type A, PTR, CNAME figurent dans la section ***Answers***
- Les réponses aux requêtes de type NS figurent dans la section ***Authority***
- Enfin, les autres informations sont dans la section ***Additional***.

III.6.2 L'entête d'un message DNS

Voici la structure de l'entête DNS basé sur 12 octets.

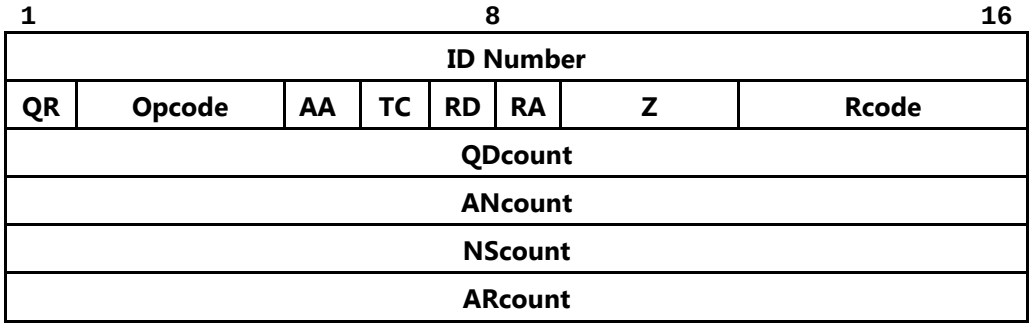


Figure III.9 – *Format de l'entête d'un message DNS.*

ID : Codé sur 16 bits, doit être recopié lors de la réponse permettant à l'application de départ de pouvoir identifier le datagramme de retour.

QR : Sur un 1 bit, ce champ permet d'indiquer s'il s'agit d'une requête (0) ou d'une réponse (1).

Opcode : Sur 4 bits, ce champ perme de spécifier le type de requête :

0 - Requête standard (Query)
1 - Requête inverse (Iquery)
2 - Status d'une requête serveur (Status)
3-15 - Réservé pour des utilisations futurs

AA : Le flag Aa, sur un bit, signifie "Authoritative Answer". Il indique une réponse d'une entité autoritaire.

TC : Le champ Tc , sur un bit, indique que ce message a été tronqué.

RD : Le flag Rd, sur un bit, permet de demander la récursivité en le mettant à 1.

RA : Le flag Ra, sur un bit, indique que la récursivité est autorisée.

Z : Le flag Z, sur trois bits, est réservé pour une utilisation futur. Il doit être placé à 0 dans tout les cas. Désormais, cela est divisé en 3 bits : 1 bit pour Z, 1 bit pour AA (Authentificated Answer) qui indique si la réponse et authentifiée, et 1 bit NAD (Non-Authenticated Data) qui indique si les données sont non-authentifiées.

Rcode : Le champ Rcode, basé sur 4 bits, indique le type de réponse.

0 - Pas d'erreur
1 - Erreur de format dans la requête
2 - Problème sur serveur
3 - Le nom n'existe pas
4 - Non implémenté
5 - Refus
6-15 - Réservés

QDcount : Codé sur 16 bits, il spécifie le nombre d'entrée dans la section "Question".

ANcount : Codé sur 16 bits, il spécifie le nombre d'entrée dans la section "Réponse".

NScount : Codé sur 16 bits, il spécifie le nombre d'entrée dans la section "Autorité".

ARcount : Codé sur 16 bits, il spécifie le nombre d'entrée dans la section "Additionnel".

III.6.3 Les RR " Ressource Record"

La base de données des serveurs de noms (fichier de domaine et fichiers de résolution inverse) est constituée "d'enregistrements de ressources", "Ressource Records" (RRs). Ces enregistrements sont répartis en classes. La seule classe d'enregistrement usuellement employée est la classe Internet (IN). L'ensemble d'informations de ressources associé à un nom particulier est composé de quatre enregistrements de ressources séparés (RR).

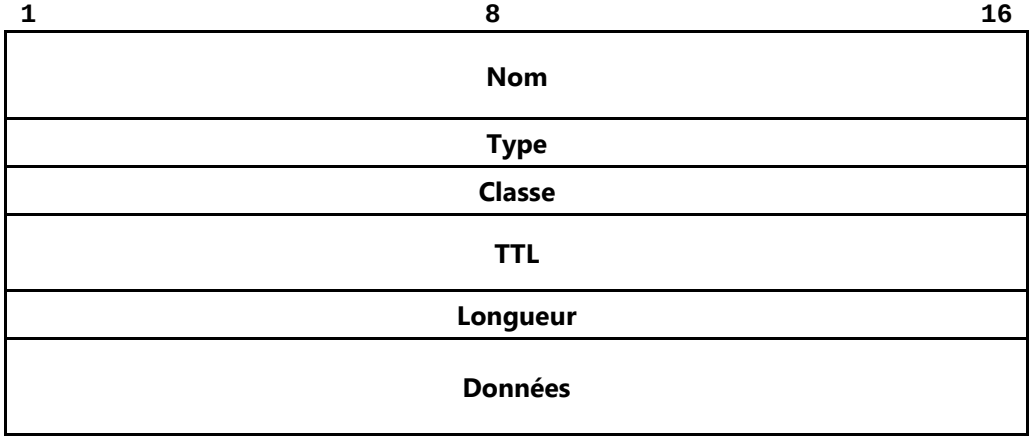


Figure III.10 – *Format de l'entête RR (N octets).*

Voici les différents champs d'un RR :

Nom : identifie l'entité, un hôte ou bien un domaine, que l'enregistrement décrit. Il doit commencer en première colonne. Les noms sont soit absolus, soit relatifs. Les noms absolus se terminent par un point

[*dot*] ; ils sont complets. En interne les programmes ne manipulent que des noms absolus, si un nom ne se termine par un point il est complété avec le domaine courant et terminé par un point. Cela permet d'utiliser les noms courts mais *attention aux erreurs* !

Type : Ce champ type, codé sur 16 bits, spécifie quels types de données sont utilisés dans le RR. Ils sont cependant classifiables en 4 groupes :

- *Zone* : identification du/des domaine(s) et NS associés.
- *Basic* : les correspondances noms/adresses et routage de mail.
- *Security* : ajout de l'authentification et signature des fichiers de zone.
- *Option* : informations supplémentaires sur les hôtes et les domaines.

Classe : Une valeur encodée sur 16 bits identifiant une famille de protocoles ou une instance d'un protocole. Voici les classes de protocole possible :

- In 01 Internet
- Cs 02 Class Csnet (obselete)
- Ch 03 Chaos (chaosnet est un ancien réseau qui historiquement a eu une grosse influence sur le développement de l'Internet, on peut considérer à l'heure actuelle qu'il n'est plus utilisé)
- Hs 04 Hesiod

TTL : C'est la durée de vie des RRs (32 bits, en secondes), utilisée par les solveurs de noms lorsqu'ils ont un cache des RRs pour connaître la durée de validité des informations du cache.

Longueur : Sur 16 bits, ce champ indique la longueur des données suivantes.

Données : Données identifiant la ressource, ce que l'on met dans ce champ dépend évidemment du type de ressources que l'on décrit.

A : Pour la classe IN, une adresse IP sur 32 bits. Pour la classe CH, un nom de domaine suivi d'une adresse octale Chaotique sur 16 bits.
Cname : un nom de domaine.
MX : une valeur de préférence sur 16 bits (la plus basse possible) suivie d'un nom d'hôte souhaitant servir d'échangeur de courrier pour le domaine de l'owner.
PTR : Une adresse IP sous forme d'un nom.
NS : Un nom d'hôte.
SOA : Plusieurs champs.

Voici un exemple montrant les différents champs saisies par Ethereal :

```
⊟ Answers
  ⊟ www.frameip.com: type A, class inet, addr 195.7.102.13
      Name: www.frameip.com
      Type: Host address
      Class: inet
      Time to live: 1 hour
      Data length: 4
      Addr: 195.7.102.13

0020  65 01 00 35 0b c0 00 39  c4 75 ba d7 81 80 00 01  e..5...9 .u......
0030  00 01 00 00 00 00 03 77  77 77 07 66 72 61 6d 65  .......w ww.frame
0040  69 70 03 63 6f 6d 00 00  01 00 01 c0 0c 00 01 00  ip.com.. ........
0050  01 00 00 0e 10 00 04 c3  07 66 0d                 ........ .f.
```