TP 2 - DNS

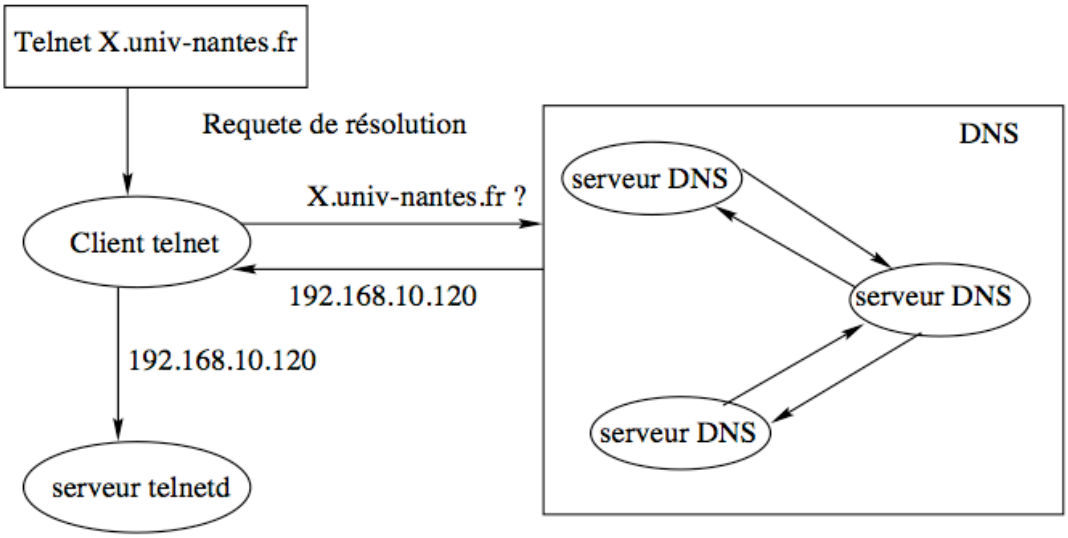
ANNEXE 1 : Présentation de DNS

1. Introduction

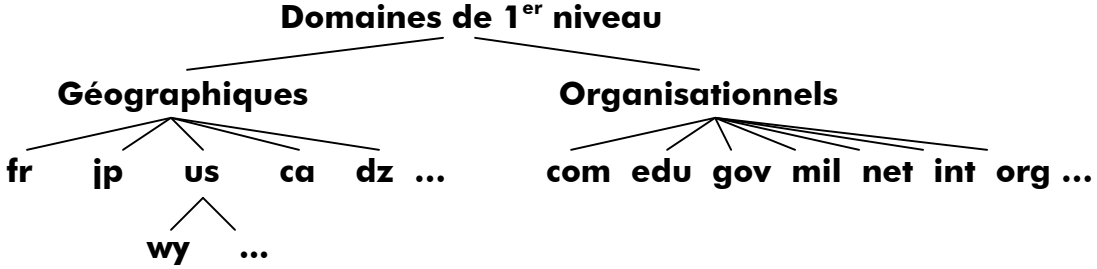
Internet est constitué de plusieurs dizaines de milliers de réseaux et eux-mêmes constitués de sous réseaux. La technologie de base TCP/IP permet l'accès aux machines par leur adresse IP. Or, il est pratiquement devenu impossible aux humains de connaître les adresses IP des machines auxquelles ils veulent accéder.
Le système DNS permet d'identifier une machine par un (des) nom(s) représentatif(s) de la machine et du (des) réseau(x) sur le(les)quel(s) elle se trouve. Il est mis en oeuvre par une base de données distribuée au niveau mondial.
Les noms sont gérés par un organisme mondial : **IANA** (*Internet Assigned Numbers Authority*), anciennement **interNIC** (*International Network Information Center*), repris par l'**ICANN** (*Internet Corporation for Assigned Names and Numbers*). Cet organisme délègue vers d'autres organismes tels que **AFNIC** (ex **NIC France**) pour la France, qui à son tour accrédite des **Bureaux d'enregistrement** (ex : opérateur réseau, hébergeur de sites, société de services…).

2. Principe

Le DNS est basé sur un système de base de données distribuée gérée de façon globale. Il garantit la diffusion d'information concernant une nouvelle machine au reste du réseau.
Par exemple, pour une session `telnet`, le logiciel client interroge un serveur de nom afin de traduire le nom du domaine auquel veut accéder l'application (résolution d'adresses). Si la réponse à la requête envoyée au serveur DNS n'est pas disponible, il transmet la requête à un autre serveur jusqu'à ce que l'association nom de domaine/adresse IP soit réalisée. Le serveur de nom retourne l'adresse IP au logiciel client. Celui ci contacte alors le serveur (`telnetd`) comme si l'utilisateur avait spécifié une adresse IP : `telnet @IP`. Le serveur local conserve la réponse (cache) pour une future requête.
Le schéma suivant montre une illustration de ce fonctionnement :



L'organisation de l'espace nom de domaine est similaire à un système de gestion de fichiers. Il est représenté sous forme d'arbre où chaque noeud est identifié par un nom et la racine est appelée `root`, identifiée par un point « . ».
Directement sous le domaine racine se trouve les domaines de 1er niveau.
Une partie de l'espace nom de domaine est appelée zone pour laquelle le serveur de nom a une autorité administrative.
Les domaines de 1er niveau sont organisés en deux catégories : les domaines géographiques et les domaines organisationnels :



`com` : organismes commerciaux
`edu` : établissements scolaires, universitaires ou de recherche américains
`gov` : agences gouvernementales américaines
`mil` : organismes militaires américaines
`net` : organismes spécialisés dans les réseaux
`int` : organismes gouvernementaux internationaux
`org` : autres (ex : organismes à but non lucratifs)

`arpa` est un nom de domaine réservé à la résolution de nom inversé.

3. Création de domaines et sous domaines

Le **IANA** est l'autorité compétente pour allouer les domaines de 1er niveau.
On appelle ***registre*** l'organisation qui maintient la base de données des domaines de 2nd niveau. On utilise souvent les termes ***Network Information Center*** ou **NIC** pour désigner cette organisation. L'**AFNIC** (Association française pour le nommage Internet en coopération) est chargé de gérer le domaine `.fr` et **VeriSign Global Registry Services** est l'entreprise chargée de gérer les domaines `.com` et `.net`.
Le NIC est l'autorité compétente pour allouer un domaine. Une fois que l'autorisation est donnée par celle-ci, le demandeur a le droit de créer des sous domaines et des noms de machines sans consulter le NIC. Le NIC ajoute alors des pointeurs dans le domaine de haut niveau vers les nouveaux serveurs de noms de domaines.

4. L'implantation du DNS

Dans la plupart des systèmes unix, l'implantation du DNS utilise le programme **BIND** (*Berkeley Internet Name Domain*). Ce programme qui joue le rôle de serveur DNS est constitué de deux composants (client/serveur) :

- Le résolveur : processus client qui crée la requête
- Le serveur de noms `named` (ou `bind9`) : est le démon qui traite la requête.

4.1. Fonctionnement du résolveur

Le résolveur demande aux serveurs de noms les informations concernant un domaine et est implanté sous forme de bibliothèque : la machine exécutant le résolveur donne l'adresse du (des) serveur (s) de noms, le résolveur interprète les réponses et retourne l'information au logiciel appelant.

Un seul fichier sert à la configuration du résolveur : `/etc/resolv.conf`.
Il contient essentiellement :

`nameserver1 @IP` (nom du serveur auquel est adressée la requête et son @IP)
`nameserver2 @IP` (nom du serveur suivant si aucune réponse de nameserver1)
`domain nom` (nom de domaine qui sera rajouté à la requête)
`search domaine` (la liste de domaine qui sera rajouté à la requête)

On utilisera soit `domain` soit `search`.

Exemple du résolveur sur la machine *quad* du CIE :

```
nameserver 127.0.0.1
nameserver 192.168.11.22
nameserver  193.52.109.10
search ensinfo.sciences.univ-nantes.fr
       ipv6.univ-nantes.fr
```

4.2 Configuration du serveur de noms

Le serveur de noms s'exécute sous la forme d'un processus appelé `named` ou `bind9`. On distingue trois types de serveurs de noms :

- **Primaire** : possède toutes les informations concernant le domaine
- **Secondaire** : transfert toute la BD du domaine à partir du primaire. Met régulièrement à jour les informations concernant un domaine depuis un fichier particulier appelé fichier de zone.
- **Cache** : conserve toutes les réponses de toutes les requêtes effectuées auprès d'autres serveurs de noms. Lorsqu'une machine est rajoutée au réseau, seul le serveur primaire est mis à jour.

Plusieurs fichiers servent à configurer le serveur de noms (exemple sous `fedora`) :

- `named.conf` : définit les pointeurs vers les BD des domaines de ce serveur
- `named.ca` : définit les pointeurs sur les serveurs du domaine racine
- `localhost.zone` : définit la résolution des adresses locales
- `named.local` : définit la résolution inverse des adresses locales
- Un fichier de zone qui fait correspondre nom de machine et @IP doit être également créé. Un autre fichier qui fait correspondre @IP et nom de machine est également créé. Il s'agit respectivement des fichiers `mazone.dns` et `mazone.rev` dans l'exemple suivant.

4.2.1 Le fichier `named.conf`

Le premier paragraphe de ce fichier commence par un champ `options` qui inclut essentiellement le répertoire principal de l'installation (`directory`).
On distingue deux types de serveurs de noms : primaire (`master`) et secondaire (`slave`). Leur configuration se fait tel que le montrent les exemples suivants :

```
zone "mazone.fr" {
type master; file "mazone.dns";
};
```

où :

`mazone.fr` est le nom de domaine pour lequel le serveur sera primaire,
`mazone.dns` désigne le fichier `/var/named/mazone.dns` où seront stockés les enregistrements de la zone.

```
zone "zone-ami.fr" {
type slave; file "zone-ami.dns";
masters { 192.169.100.10; };
};
```

où :

`zone-ami.fr` est le nom de domaine pour lequel le serveur sera secondaire,
`zone-ami.dns` désigne le fichier `/var/named/zone-ami.dns` où sera écrite la zone à l'issue du premier transfert depuis le serveur DNS primaire,
`192.169.100.10` est l'adresse IP du serveur primaire de la zone.

4.2.2 Les autres fichiers

À l'exception du fichier `named.conf`, tous les autres fichiers ont le même format de base et utilisent le même type d'enregistrement de base de données.

On citera en particulier les six plus importants enregistrements :

- **`SOA`** : *Start Of Authority*, est le premier enregistrement de chaque fichier. Il permet de préciser le nom de l'hôte sur lequel a été crée ce fichier, l'adresse de la personne qui maintient ce fichier et une liste de nombre qui servent pour le DNS secondaire :
  - **`serial`** : numéro (un nombre entier) de version du fichier d'information de zones. Ce numéro est utilisé par les DNS secondaires pour savoir si le fichier d'informations de zone du DNS primaire a été changé. Il doit être incrémenté à chaque modification du fichier. Généralement sous la forme AAAAMMJJXX (date à l'envers suivi d'un n°XX).
  - **`refresh`** : intervalle de temps en secondes durant lequel le DNS secondaire attend avant de vérifier (et éventuellement mettre à jour) l'enregistrement SOA du DNS primaire. Il vaut généralement 86400 secondes (une journée).
  - **`retry`** : intervalle de temps en secondes durant lequel le DNS secondaire attend avant de réessayer une requête vers le DNS primaire si celui ci n'est pas accessible. Cette valeur devrait être de quelques minutes.
  - **`expire`** : intervalle de temps en secondes durant lequel le DNS secondaire attend avant de rejeter les informations de zones s'il n'a pu contacter le DNS primaire. Cette valeur devrait être de plusieurs jours voire plusieurs mois.

- **NS** : *Name Server*, permet d'indiquer le serveur de nom pour un domaine donné.
- **A** : *Address*, permet d'associer un hôte donné à un numéro IP. Cet enregistrement est réservé aux fichiers `localhost.zone` et `mazone.dns`.
- **PTR** : *PoinTeR*, permet d'associer un numéro IP donné à un hôte. Cet enregistrement est réservé aux fichiers de type named.rev.
- **CNAME** : *Canonical NAME*, permet de définir des alias.
- **MX** : *Mail eXchange*, identifie la machine à qui on doit transmettre le courrier électronique destiné à un nom de domaine donnée.

Le format d'enregistrement des ressources DNS est **:**

`[nom] [ttl] IN type données.`

- Le **nom** identifie l'objet que la ressource référence.
- Le **ttl** (*time to live*) est la durée maximale pendant laquelle les informations de cet enregistrement pourront être conservées dans le cache d'un système distant ; généralement, il est laissé vide.
- **IN** permet d'identifier l'enregistrement comme un enregistrement ressource DNS.
- Le **type** identifie le type d'enregistrement (liste ci-dessus).
- Enfin la **donnée** correspond à l'information spécifique à ce type.

4.3 Lancement du serveur de noms

Après avoir construit le fichier `named.conf` et tous les autres fichiers, lancer `bind9` avec la commande : `/etc/init.d/bind9 start`

Vérifier les messages d'erreurs dans le fichier de messages sous : `/var/log/syslog` avec la commande : `tail –30 /var/log/syslog`

TP 2 - DNS
ANNEXE 2 : Exemples de fichiers de configuration de DNS

Fichier principal : /etc/bind/named.conf

C'est le fichier de configuration de Bind (serveur de nom).
Sous Linux (Debian) il contient :

```
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind/README.Debian for information on the //
structure of BIND configuration files in Debian for BIND versions 8.2.1
// and later, *BEFORE* you customize this configuration file.
//

options { directory "/etc/bind";

      // If there is a firewall between you and nameservers you want
      // to talk to, you might need to uncomment the query-source
      // directive below. Previous versions of BIND always asked
      // questions using port 53, but BIND 8.1 and later use an unprivileged
      // port by default.

      // query-source address * port 53;

      // If your ISP provided one or more IP addresses for stable
      // nameservers, you probably want to use them as forwarders.
      // Uncomment the following block, and insert the addresses replacing
      // the all-0's placeholder.

      // forwarders {
      // 0.0.0.0;
      // };
};

// reduce log verbosity on issues outside our control logging {
      category lame-servers { null; };
      category cname { null; };
};

// prime the server with knowledge of the root servers zone "." {
      type hint;
      file "db.root";
};

// be authoritative for the localhost forward and reverse zones, and for
// broadcast zones as per RFC 1912

zone "localhost" {
      type master;
      file "db.local";
// fichier.dns
};
```

```
zone "127.in-addr.arpa" {
     type master;
     file "db.127";
};

zone "0.in-addr.arpa" {
     type master;
     file "db.0";
};

zone "255.in-addr.arpa" {
     type master;
     file "db.255";
};

// add entries for other zones below here

// Creation d'une zone : "mazone.fr"

zone "mazone.fr" {
     type master;
     file "mazone.dns";
};

// pour permettre la résolution inverse

zone "xxx.xxx.xxx.in-addr.arpa" {
     type master;
     file "mazone.rev";
};

// On remplace xxx.xxx.xxx par l'inverse de l'adresse IP du réseau
// exemple :
// Pour le réseau 192.168.10.0 on a : 10.168.192.in-addr.arpa
```

Exemple de fichier /etc/bind/nomfichier.dns

C'est le fichier de configuration de Bind (serveur de nom).

Ce fichier (exemple sous Linux/Debian) contient les informations permettant de retrouver à partir de son adresse IP, le nom d'un hôte.

```
$TTL 3D
@       IN  SOA nserveur.mazone.fr. administrateur.mazone.fr. (
                199609206 ; serial, todays date + todays serial #
                8H ; refresh, seconds
                2H ; retry, seconds
                4W ; expire, seconds
                1D ) ; minimum, seconds
            NS nserveur.mazone.fr.
            NS st-11.mazone.fr.
            MX 10 st-11.mazone.fr. ; Primary Mail Exchanger

localhost   A   127.0.0.1
nserveur    A   192.168.10.1
```

```
serv2        A    192.168.10.2
serv3        A    192.168.10.3


; les alias
ftp      CNAME    st-11.mazone.fr.
mail     CNAME    st-11.mazone.fr.


;
; les stations de ma zone et leur adresse
;
st-11    A    192.168.10.200
         MX   10   st-11.mazone.fr. ; serveur de mail

st-12    A    192.168.10.201
MX       10   st-12.mazone.fr. ; serveur de mail

st-13    A    192.168.10.202
st-14    A    192.168.10.203
```

Exemple de fichier /etc/bind/nomfichier.rev

Ce fichier (exemple sous Linux/Debian) contient les informations permettant de retrouver le nom d'un hôte à partir de son adresse IP.

```
$TTL 3D
@        IN   SOA nserveur.mazone.fr. administrateur.mazone.fr. (
                  199609206 ; Serial
                  28800 ; Refresh
                  7200 ; Retry
                  604800 ; Expire
                  86400 ) ; Minimum TTL
             NS nserveur.mazone.fr.
             NS st-11.mazone.fr.
;
;        les serveurs
;
1        PTR      nserveur.mazone.fr.
2        PTR      serv2
3        PTR      serv3
;
;        les stations
;
200      PTR      st-11.mazone.fr.
201      PTR      st-12.mazone.fr.
202      PTR      st-13.mazone.fr.
203      PTR      st-14.mazone.fr.

}
```