

ROYAUME DU MAROC

مكتب التكوين المهني وإنعاش الشغل

Office de la Formation Professionnelle et de la Promotion du Travail

# Installation du service DNS sous Gnu/Linux

[www.ofppt.info](http://www.ofppt.info)



OFPPT

DIRECTION RECHERCHE ET INGENIERIE DE FORMATION  
SECTEUR NTIC

## Sommaire

1.	Qu'est ce que le service de résolution de noms de domaine .....	3
1.1.	Présentation des concepts .....	4
1.1.1.	Notion de domaine, de zone et de délégation.....	4
1.1.2.	le domaine in-addr.arpa.....	7
1.1.3.	Fichiers, structure et contenus .....	8
1.1.4.	Principaux types d'enregistrements .....	8
1.1.5.	Structure des enregistrements.....	9
1.1.6.	La délégation.....	10
1.1.7.	Serveur primaire et serveur secondaire.....	10
1.1.8.	Le cache .....	11
2.	Installation et configuration d'un serveur DNS .....	11
2.1	Fichiers déjà installés.....	11
2.2	rndc, le fichier de configuration, le fichier de clé.....	12
2.2.A	Procédure de configuration du serveur .....	13
2.2.B	Configurer les fichiers .....	13
2.2.C	Configuration du DNS manuellement .....	13
2.2.D	Le fichier named.conf .....	14
2.2.E	Le fichier db.foo.org .....	14
2.2.F	Le fichier db.foo.org.rev .....	14
2.3	Procédure de tests .....	14
2.3.A	Vérifier la résolution de noms : .....	14
3.	Dépannage et outils.....	15
3.1	Les erreurs de chargement de bind .....	15
3.2	nslookup, dig .....	16
3.2.A	Le cache du DNS.....	18
3.2.B	Les journaux .....	18
4.	Remarques.....	19

<b>www.ofppt.info</b>	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	1 - 27

## Système de fichiers réseau sous Gnu/Linux

5.	Annexes.....	19
5.1	Annexe 1 - extraits de fichiers de configuration.....	19
5.2	Annexe 2 - Serveur primaire et serveur secondaire .....	24
5.3	Annexe 3 - Mise en place d'une délégation de zone .....	25
5.4	Annexe 3 - Outils de diagnostic et de contrôle .....	25

<b>www.ofppt.info</b>	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	2 - 27

## 1. Qu'est ce que le service de résolution de noms de domaine

Le service de résolution de noms d'hôtes DNS (Domain Name Services), permet d'adresser un hôte par un nom, plutôt que de l'adresser par une adresse IP. Quelle est la structure d'un nom d'hôte?

Exemple :            Nom\_d\_hôte            ou bien            Nom\_d\_hôte.NomDomaine  
                      ns1                    ou bien            ns1.foo.org

Le nom de domaine identifie une organisation dans l'internet, comme, par exemple, yahoo.com, wanadoo.fr, eu.org. Dans les exemples, nous utiliserons un domaine que l'on considère fictif : « foo.org ». Chaque organisation dispose d'un ou plusieurs réseaux. Ces réseaux sont composés de noeuds, ces noeuds (postes, serveurs, routeurs, imprimantes) pouvant être adressés.

Par exemple, la commande `ping ns1.foo.org`, permet d'adresser la machine qui porte le nom d'hôte ns1, dans le domaine (organisation) foo.org.

Quelle différence entre la résolution de noms d'hôtes avec un serveur DNS et les fichiers `hosts` ?

Avec les fichiers `hosts`, chaque machine dispose de sa propre base de données de noms. Sur des réseaux importants, cette base de données dupliquée n'est pas simple à maintenir.

Avec un service de résolution de noms, la base de données est localisée sur un serveur. Un client qui désire adresser un hôte regarde dans son cache local, s'il en connaît l'adresse. S'il ne la connaît pas il va interroger le serveur de noms.

Tous les grands réseaux sous TCP/IP, et internet fonctionnent (schématiquement) sur ce principe.

Avec un serveur DNS, un administrateur n'a plus qu'une seule base de données à maintenir. Il suffit qu'il indique sur chaque hôte, quelle est l'adresse de ce serveur. Ici il y a 2 cas de figures possibles :

- soit les hôtes (clients) sont des clients DHCP (Dynamic Host Configuration Protocol), cette solution est particulière et n'est pas abordée ici. Cette technique est l'objet d'un autre chapitre.
- soit les clients disposent d'une adresse IP statique. La configuration des clients est détaillée dans ce document.

Normalement un service DNS nécessite au minimum deux serveurs afin d'assurer un minimum de redondance. Les bases de données des services sont synchronisées. La configuration d'un serveur de noms secondaire sera expliquée. Nous verrons également en TP le fonctionnement de la réplication des bases de données (bases d'enregistrements de ressources). On peut parler de bases de données réparties et synchronisées.

<b>www.ofppt.info</b>	Document	Millésime	Page
	Systeme de fichiers reseau sous Gnu/Linux	août 14	3 - 27

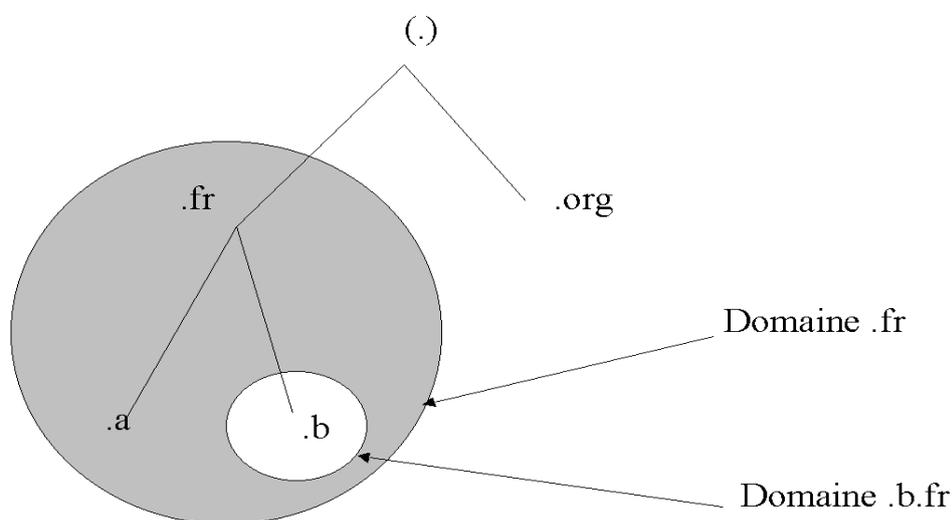
## 1.1. Présentation des concepts

### 1.1.1. Notion de domaine, de zone et de délégation

Un « domaine » est un sous-arbre de l'espace de nommage. Par exemple `.com` est un domaine, il contient toute la partie hiérarchique inférieure de l'arbre sous jacente au noeud `.com`.

Un domaine peut être organisé en sous domaines. `.pirlouit.com` est un sous domaine du domaine `.com`. Un domaine peut être assimilé à une partie ou sous-partie de l'organisation de l'espace de nommage. Voir la diapositive sur les Domaines, zones et délégations.

Figure 1.1. Les domaines



Un domaine est un sous arbre de l'espace de nommage.

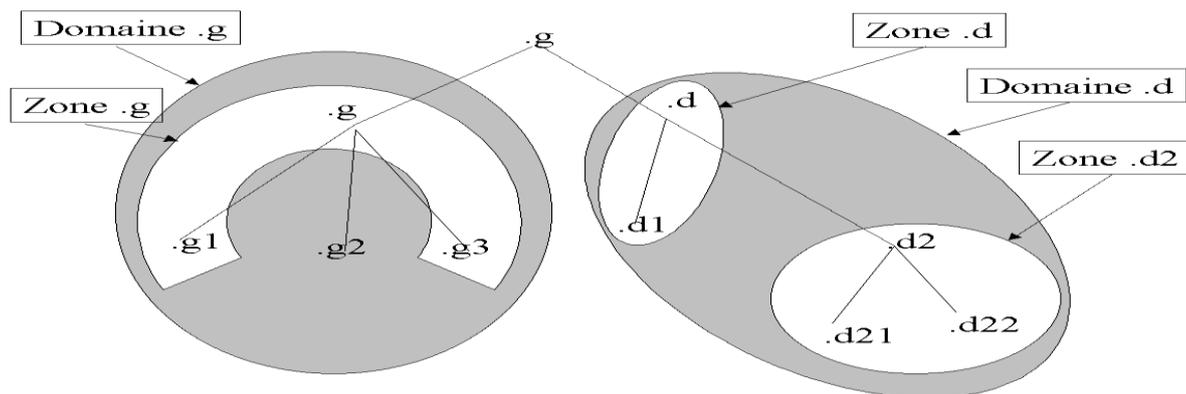
Une "zone" est une organisation logique (ou pour être plus précis, une organisation administrative) des domaines. Le rôle d'une zone est principalement de simplifier l'administration des domaines.

Le domaine `".com"` peut être découpé en plusieurs zones, `z1.com`, `z2.com...zn.com`.

L'administration des zones sera déléguée afin de simplifier la gestion globale du domaine.

Voir la diapositive sur les zones.

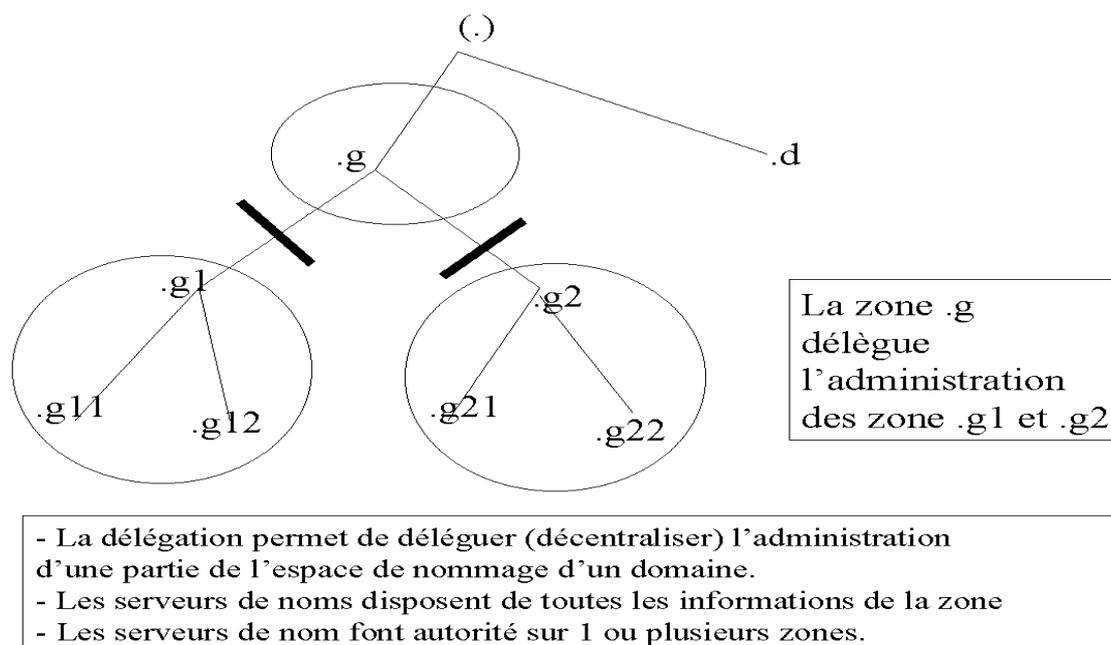
Figure 1.2. Les zones



Une zone est une organisation gérée par délégation. C'est un découpage en unités du domaine.

La délégation consiste à déléguer l'administration d'une zone (ou une sous-zone) aux administrateurs de cette zone. Voir la diapositive sur la délégation.

Figure 1.3. La délégation



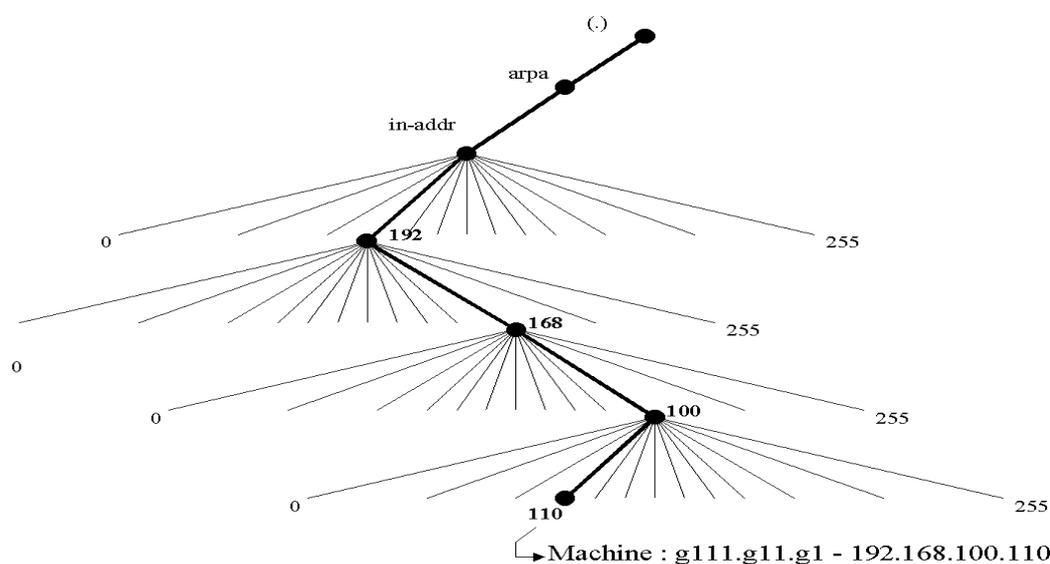
Attention à ces quelques remarques :

- Un domaine est une organisation de l'espace de nommage. Il peut être attaché à un domaine parent, et/ou peut avoir un ou plusieurs sous-domaines enfants.
- Les zones correspondent à des organisations administratives des domaines. Un domaine peut être administré par plusieurs zones administratives, mais il est possible aussi qu'une zone serve à l'administration de plusieurs domaines. Prenons l'exemple d'un domaine "MonEntreprise.fr", membre de ".fr". Il peut être composé de trois sous-domaines France.MonEntreprise.fr, Italie.MonEntreprise.fr, Espagne.MonEntreprise.fr et de deux zones d'administration. Une en France pour les sous-domaines France.MonEntreprise.fr, Italie.MonEntreprise.fr (il n'y a pas de délégation), et une pour Espagne.MonEntreprise.fr, il y a délégation.
- L'adressage IP correspond à une organisation physique des noeuds sur un réseau IP.
- L'organisation de l'espace de nommage est complètement indépendante de l'implantation géographique d'un réseau ou de son organisation physique. L'organisation physique est gérée par des routes (tables de routage). L'espace de nommage indique pour un nom de domaine N, quelles sont les serveurs de noms qui ont autorité sur cette zone. Elles ne donnent pas la façon d'arriver à ces machines.
- Les seules machines connues au niveau de l'espace de nommage, sont les serveurs de nom "déclarés". Ces informations sont accessibles par des bases de données "whois".
- La cohérence (le service de résolution de noms) entre l'organisation de l'espace de nommage global et les organisations internes des réseaux sur internet est réalisée par les serveurs de noms.

### 1.1.2. le domaine in-addr.arpa

Le principe de la resolution de noms, consiste a affecter un nom d'hte une adresse IP. On parle de resolution de noms directe. Le processus inverse doit pouvoir egalement etre mis en oeuvre. On parle de resolution de noms inverse ou reverse. Le processus doit fournir, pour une adresse IP, le nom correspondant. Pour cela il y a une zone particuliere, in-addr.arpa, qui permet la resolution inverse d'adresse IP. Voir la diapositive sur la resolution inverse.

Figure 1.4. La resolution inverse



L'adresse ip 192.168.100.110 correspond au sous-domaine 110.100.168.192.in-addr.arpa qui renvoie le nom qualifie g111.g11.g1

Par exemple, pour le reseau 192.168.1.0, on creera une zone inverse dans le domaine in-addr.arpa. La zone de recherche inverse dans le domaine deviendra : 1.168.192.in-addr.arpa. Cette zone devra repondre pour toutes les adresses declarees dans la tranche 192.168.1.0 a 192.168.1.254.

On inscrira dans cette zone tous les noeuds du reseau pour lesquels on desire que la resolution inverse fonctionne. Un serveur de noms peut, pratiquement, fonctionner sans la definition de cette zone tant que le reseau n'est pas relie a l'internet. Si cela etait le cas, il faudrait declarer cette zone, sans quoi, des services comme la messagerie electronique, ne pourrait fonctionner correctement, notamment a causes des regles anti-spam. (Voir [www.nic.fr](http://www.nic.fr))

### 1.1.3. Fichiers, structure et contenus

Sur linux nous allons utiliser deux types de fichiers :

- le fichier `/etc/bind/named.conf`, qui décrit la configuration générale du serveur DNS,
- les fichiers qui contiennent les enregistrements de ressources pour la zone dans `/etc/bind`. On crée, en général, un fichier pour la résolution directe d'une zone, et un fichier pour la résolution inverse.

Les enregistrements ont une structure et un rôle que nous verrons. Le daemon se nomme `named`, prononcer « naime dé ».

### 1.1.4. Principaux types d'enregistrements

Les types d'enregistrements qui enrichissent une base de données DNS sont de plusieurs types, dont voici les principaux :

- Enregistrement de type SOA (*Start Of Authority*) : indique l'autorité sur la zone. Ces enregistrements contiennent toutes les informations sur le domaine. Par exemple le délai de mise à jour des bases de données entre serveurs de noms primaires et secondaires, le nom du responsable du site
- Enregistrements de type NS (*Name Server*) : ces enregistrements donnent les adresses des serveurs de noms pour le domaine.
- Enregistrement de type A (*Adresse*) : ces enregistrements permettent de définir les noeuds fixes du réseau (ceux qui ont des adresses IP statiques). Serveurs, routeurs, switches ...
- Enregistrements de type MX (*Mail eXchanger*) : ils servent pour déclarer les serveurs de messagerie. Il faudra déclarer un enregistrement de type MX pour la réalisation du TP sur la messagerie.
- Enregistrements de type CNAME (*Canonical Name*) : ils permettent de définir des alias sur des noeuds existants. Par exemple `www.foo.org` peut être la même machine que `web.foo.org`. Dans ce cas, « `www` » est un alias (CNAME) de « `web` ». Cela permet de différencier le nommage des machines des standards de nommages des services (`www`, `ftp`, `news`, `smtp`, `mail`, `pop`...).
- Enregistrement de type PTR (*Pointeur*) : ils permettent la résolution de noms inverse dans le domaine `in-addr.arpa`.

Ces enregistrements caractérisent des informations de type IN - INternet. Voir l'annexe pour avoir un fichier exemple.

<b>www.ofppt.info</b>	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	8 - 27

### 1.1.5. Structure des enregistrements

*Structure d'un enregistrement SOA* : chaque fichier de ressource de zone commence par un enregistrement de type SOA. Voici un exemple d'enregistrement SOA :

```
$TTL 38400
foo.org. IN SOA ns1.foo.org. hostmaster.foo.org. (
    20001210011      ; numéro de série
    10800           ; rafraîchissement
    3600            ; nouvel essai
    604800          ; Obsolescence après une semaine
    86400 )         ; TTL minimal de 1 jour
```

*Caractéristiques des différentes informations :*

SOA Start Of Authority, enregistrement qui contient les informations de synchronisation des différents serveurs de nom.

foo.org, donne le nom de la zone. Le nom de la zone, ici "foo.org", peut être remplacé par "l'", arobase.

hostmaster.foo.org : la personne qui est responsable de la zone. Le premier point sera remplacé par l'arobase (@) pour envoyer un courrier électronique. Cela deviendra hostmaster@foo.org. En général postmaster, est un alias de messagerie électronique vers l'administrateur du DNS.

1. Numéro de série sous la forme AAAAMMJJNN, sert à identifier la dernière modification sur le serveur de noms maître. Ce numéro sera utilisé par les serveurs de nom secondaires pour synchroniser leurs bases. Si le numéro de série du serveur de noms primaire est supérieur à celui des serveurs de noms secondaire, alors le processus de synchronisation suppose que l'administrateur a apporté une modification sur le serveur maître et les bases seront synchronisées.
2. **Rafraîchissement** : Intervalle de temps donné en seconde pour indiquer au serveur la périodicité de la synchronisation.
3. **Retry** : intervalle de temps avant réitération si l'essai précédent n'a pas fonctionné.
4. **Expire** : temps au bout duquel le serveur ne remplit plus sa mission s'il n'a pu contacter le serveur maître pour mettre à jour ses données.
5. **TTL** : Time To Live, durée de vie des enregistrements. Plus la durée de vie est courte, plus l'administrateur est susceptible de considérer que ses bases sont à jour, par contre cela augmente le trafic sur le réseau.

<b>www.ofppt.info</b>	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	9 - 27

*Enregistrement de type NS pour le domaine foo.org :*

```
foo.org.      IN NS   ns1.foo.org.    ; noter le point final "."
foo.org.      IN NS   ns2.foo.org.    ; foo.org peut être remplacé par "@"
                ; IN signifie enregistrement de type INternet
```

Le « . » final signifie que le nom est pleinement qualifié. On aurait pu mettre :

```
@             IN NS   ns1
             IN NS   ns2
```

"@" signifie "foo.org" et pour le serveur de nom, comme "ns1" n'est pas pleinement qualifié, cela équivaut à "ns1.foo.org".

*Enregistrements de type A :* nous devons décrire la correspondance Nom / Adresse

```
ns1.foo.org.      IN      A      192.168.0.1
ns2.foo.org.      IN      A      192.168.0.2
localhost.foo.org. IN      A      127.0.0.1
```

S'il y avait d'autres hôtes sur la zone, il faudrait les définir ici.

*Enregistrements de type CNAME :* Ce sont les alias (Canonical Name). Une requête du type <http://www.foo.org> sera adressée à ns1.foo.org, puisque www est un alias de ns1.

```
www             IN      CNAME  ns1.foo.org.
ftp             IN      CNAME  ns1.foo.org.
```

*Enregistrement de type PTR :* il serviront à la résolution de noms inverse.

```
1.0.168.192.in-addr.arpa. IN      PTR     ns1.foo.org.
2.0.168.192.in-addr.arpa. IN      PTR     ns2.foo.org.
```

## 1.1.6. La délégation

La délégation consiste à donner l'administration d'une partie du domaine à une autre organisation. Il y a transfert de responsabilité pour l'administration d'une zone. Les serveurs de la zone auront autorité sur la zone et auront en charge la responsabilité de la résolution de noms sur la zone. Les serveurs ayant autorité sur le domaine auront des pointeurs vers les serveurs de noms ayant autorité sur chaque zone du domaine.

## 1.1.7. Serveur primaire et serveur secondaire

Le serveur maître (primaire) dispose d'un fichier d'information sur la zone. Le ou les serveurs esclaves (secondaires) obtiennent les informations à partir d'un serveur primaire ou d'un autre serveur esclave. Il y a "transfert de zone". Les serveurs maîtres et esclaves ont autorité sur la zone.

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	10 - 27

### 1.1.8. Le cache

L'organisation d'internet est assez hiérarchique. Chaque domaine dispose de ses propres serveurs de noms. Les serveurs peuvent être sur le réseau physique dont ils assurent la résolution de nom ou sur un autre réseau. Chaque zone de niveau supérieur (edu, org, fr...) dispose également de serveurs de nom de niveau supérieur. L'installation du service DNS, installe une liste de serveurs de noms de niveaux supérieurs. Cette liste permet au serveur de résoudre les noms qui sont extérieurs à sa zone. Le serveur enrichit son cache avec tous les noms résolus. Si votre réseau n'est pas relié à internet, vous n'avez pas besoin d'activer cette liste.

Ce fichier est un peu particulier. Il est fourni avec les distributions. Il est utilisé par le serveur de noms à l'initialisation de sa mémoire cache. Si vos serveurs sont raccordés à internet, vous pourrez utiliser une liste officielle des serveurs de la racine (<ftp.rs.internic.net>).

## 2. Installation et configuration d'un serveur DNS

Processus de configuration

L'application est déjà installée. Pour mettre en place le service de résolution de noms sur un serveur GNU/Linux, on va procéder successivement aux opérations suivantes :

1. vérifier les fichiers déjà installés,
2. configurer les fichiers des zones administrées,
3. configurer les fichiers de transaction sécurisée pour rndc,
4. démarrer et tester le service serveur.

### 2.1 Fichiers déjà installés

Vous devez normalement avoir déjà les fichiers suivants :

1. `/etc/bind/named.conf`, fichier de déclaration des fichiers de ressources
2. `/etc/bind/db.127`, zone locale reverse
3. `/etc/bind/db.0`, zone locale de broadcast
4. `/etc/bind/db.255`, zone locale de broadcast
5. `db.local`, zone directe locale
6. `db.root`, fichiers des serveurs racine

Le contenu de tous ces fichiers et commentaires se trouve en annexe.

Vous avez également des fichiers particuliers : `rndc.key`, `rndc.conf`. `rndc`, est un outil qui permet de passer des commandes à distance à un serveur de nom. Nous porterons une attention toute particulière à ces fichiers, à leur rôles et à l'utilité de `rndc`.

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	11 - 27

Il va suffire de rajouter les fichiers manquants à la zone administrée.

## 2.2 rndc, le fichier de configuration, le fichier de clé

rndc est un outil qui permet de réaliser des transactions sécurisées avec un serveur de nom. Le mode de fonctionnement est dit à "clé partagée", c'est à dire que le client rndc et le serveur bind doivent avoir la même clé. Vous devrez donc configurer le fichier de configuration de rndc et le fichier named.conf avec les mêmes paramètres.

Ces fichiers et exemples sont également fournis en annexe. La clé doit être strictement identique dans les 2 fichiers. Si vous avez un message d'erreur à l'utilisation de rndc, vérifiez bien ces paramètres.

rndc supporte plusieurs paramètres pour passer des commandes au serveur de nom (halt, querylog, refresh, reload, stat...). Utilisez la commande "man rndc" pour en savoir plus.

Dans le fichier rndc, vous allez avoir besoin d'au moins 3 paramètres. rndc utilisera ces paramètres si rien n'est spécifié sur la ligne de commande. Dans les autres cas, vous pouvez passer les paramètres sur la ligne de commande.

Note : vous pouvez vous passer du système de clé mais ce n'est pas conseillé. Commentez tout ce qu'il y a dans le fichier named.conf et qui concerne la clé s'il y a déjà des choses. Renommez le fichier rndc.conf en rndc.conf.orig, ça devrait fonctionner. Vous pouvez tester cela en faisant un `/etc/init.d/bind restart`. Vous ne devriez pas avoir de message d'erreur.

```
#Description du serveur et de la clé utilisés par défaut.
#Ici on utilise par défaut le serveur local, avec la clé key-name
options {
    default-server localhost;
    default-key     "<key-name>";
};
```

Il est possible de dire quelle clé utiliser en fonction d'un serveur donné.

```
server localhost {
    key "<key-name>";
};
```

Enfin il reste à définir la ou les clés avec leur noms et leurs valeurs.

```
key "<key-name>" {
    algorithm hmac-md5;
    secret "<key-value>";
};
```

Pour créer une nouvelle clé, utilisez la commande :

```
dnssec-keygen -a hmac-md5 -b <bit-length> -n HOST <key-file-name>
```

```
#Ici on génère une clé de 512 bits dans un fichier maCLE
dnssec-keygen -a hmac-md5 -b 512 -n HOST maCLE
```

www.ofppt.info	Document	Millésime	Page
	Systeme de fichiers reseau sous Gnu/Linux	août 14	12 - 27

Le fichier `named.conf` doit connaître la clé utilisée par le client,

```
// secret must be the same as in /etc/rndc.conf
key "key" {
    algorithm      hmac-md5;
    secret
    "c3Ryb25nIGVub3VnaCBmb3IgYSBtYW4gYnV0IG1hZGUgZm9yIGEgd29tYW4K";
};
```

mais doit également comprendre les paramètres qui définissent les machines clientes autorisées à passer des commandes avec une directive `controls`.

```
controls {
    inet 127.0.0.1 allow { localhost; } keys { <key-name>; };
};

# Ici on peut passer des commandes à partir de n'importe quelle machine
controls {
    inet 127.0.0.1 allow { any; } keys { "key"; };
};

# Ici on peut passer des commandes localement
controls {
    inet 127.0.0.1 allow { localhost; } keys { "key"; };
};
```

### 2.2.A Procédure de configuration du serveur

L'installation a copié les fichiers. Sur une configuration simple vous allez avoir 3 fichiers à créer ou à modifier sur le serveur primaire :

- `/etc/bind/named.conf` (fichier de configuration globale du service DNS du serveur de noms primaire),
- `/etc/bind/db.foo.org` qui contiendra la description de la correspondance nom-adresse de toutes les machines du réseau
- `/etc/bind/db.foo.org.rev` qui contiendra la correspondance inverse adresse-nom (pour la résolution inverse de nom in-addr.arpa).

### 2.2.B Configurer les fichiers

Vous pouvez configurer le serveur manuellement, c'est à dire créer les fichiers à l'aide d'un éditeur de texte ou à l'aide d'un outil de configuration graphique. En général on n'installe jamais d'interface graphique sur un serveur pour des questions de sécurité. Nous allons donc créer les fichiers complètement. La configuration est réalisable également à distance avec des requêtes HTTP grâce à des outils comme webmin.

### 2.2.C Configuration du DNS manuellement

Le fichier racine pour la configuration du serveur de noms est le fichier `/etc/bind/named.conf`. Ce fichier est lu au démarrage du service et donne la liste des fichiers qui définissent la base de données

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	13 - 27

pour la zone.

### 2.2.D Le fichier `named.conf`

Voir annexe.

### 2.2.E Le fichier `db.foo.org`

Voir annexe.

Le paramètre @, signifie qu'il s'agit du domaine "foo.org" (le nom tapé après le mot " zone " dans le fichier de configuration `named.conf`). Le paramètre "IN", signifie qu'il s'agit d'un enregistrement de type internet. Notez la présence d'un point (.) après le nom des machines pleinement qualifiés. Sans celui-ci, le nom serait " étendu ". Par exemple, `ns1.foo.org` (sans point) serait compris comme `ns1.foo.org.foo.org` (on rajoute le nom de domaine en l'absence du point terminal). Le point (.) terminal permet de signifier que le nom est pleinement qualifié.

### 2.2.F Le fichier `db.foo.org.rev`

Voir annexe.

## 2.3 Procédure de tests

Attention au fichier `hosts` et au fichier `host.conf`. Prenez le temps de regarder ce qu'il y a dedans. Faites une copie de sauvegarde de ces fichiers et renommez les. Vérifiez au besoin leur utilité avec les commandes `man host.conf` et `man hosts`.

Vous pouvez tester votre configuration avant même d'avoir configuré un client. Sur la même machine vous allez utiliser un service client du serveur (commande **ping**) qui utilisera un service serveur (DNS).

*Test sur le serveur de noms* : Tapez la commande `ping ftp.foo.org`. Si la commande répond, le serveur fonctionne. En effet `ftp` est un alias de `ns1` dans la zone `foo.org`.

*Test sur le client* : Avant de lancer une commande, vous devez vérifier que vous n'avez pas de fichier `hosts` local, sinon vous devez le supprimer.

*Pourquoi ?* L'utilisation de fichiers `hosts` et d'un serveur de noms n'est pas exclusif. Dans bien des environnements, le fichier `hosts` est consulté avant le serveur de noms (notamment windows, GNU/Linux à moins que ce ne soit précisé). Si vous avez un fichier `hosts` sur la machine, vous pouvez avoir des résultats qui ne sont pas ceux attendus.

### 2.3.A Vérifier la résolution de noms :

Pensez à bien vérifier le nom d'hôte de votre machine avec la commande **hostname**, au besoin, sous root, modifiez ce nom, toujours avec cette commande. Fermez les sessions et rouvrez les, vous aurez le bon nom d'hôte qui s'affichera sur votre console.

<b>www.ofppt.info</b>	Document	Millésime	Page
	Systeme de fichiers reseau sous Gnu/Linux	août 14	14 - 27

## Système de fichiers réseau sous Gnu/Linux

Mettons que le réseau soit configuré de la façon suivante :

Nom d'hôte	Alias (CNAME)	Adresse IP	Serveur
ns1	www		
	ftp		
	mail		
	ns1	192.68.1.1	
Client 1	Cli1	192.68.1.2	

Pour vérifier le fonctionnement de la résolution de noms à partir du client cli1, vous pouvez utiliser les commandes suivantes :

```
- ping ns1
- ping cli1
```

Vous pouvez également tester la résolution des alias (CNAME) avec les commandes :

```
ping mail.foo.org
ping www.foo.org
ping ftp.foo.org
ping ns1.foo.org
```

C'est bien la même adresse IP (voir le cache arp) qui répond, la machine a donc bien plusieurs noms.

Si vous voulez vérifier que c'est bien le serveur de noms qui réalise la résolution, il existe plusieurs solutions. La plus simple est d'arrêter le service serveur avec la commande `/etc/init.d/bind stop`, puis de refaire les manipulations. Aucune machine n'est atteignable en utilisant son nom, mais cela est toujours possible en utilisant l'adresse IP.

### 3. Dépannage et outils

Les sources de dysfonctionnement des services de noms peuvent être nombreuses et parfois complexes à résoudre. Voici quelques outils et méthodes qui peuvent être utilisés.

#### 3.1 Les erreurs de chargement de bind

Si vous avez une erreur similaire à celle-ci :

```
Problème de clés entre named et rndc
root@knoppix:/etc/bind# /etc/init.d/bind9 stop
Stopping domain name service: namedrndc: connection to remote host closed
This may indicate that the remote server is using an older version of
the command protocol, this host is not authorized to connect,
or the key is invalid.
```

Le problème est lié à rndc, et souvent à des clés qui sont différentes ou mal définies entre `named.conf` et `rndc.conf`. Vérifiez donc bien tous les paramètres.

Vérifiez dans les journaux (en général `/var/log/daemon`) qu'il n'y a pas d'erreur de chargement de named. Voici un exemple de log.

```
# Log après chargement des zones
```

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	15 - 27

## Système de fichiers réseau sous Gnu/Linux

```
Apr  8 23:12:46 knoppix named[1066]: starting BIND 9.2.1
Apr  8 23:12:46 knoppix named[1066]: using 1 CPU
Apr  8 23:12:46 knoppix  []

named[1068]: loading configuration from '/etc/bind/named.conf'
named[1068]: no IPv6 interfaces found
named[1068]: listening on IPv4 interface lo, 127.0.0.1#53
named[1068]: listening on IPv4 interface eth0, 192.168.90.100#53
named[1068]: command channel listening on 127.0.0.1#953
named[1068]: zone 0.in-addr.arpa/IN: loaded serial 1
named[1068]: zone 127.in-addr.arpa/IN: loaded serial 1
named[1068]: zone 255.in-addr.arpa/IN: loaded serial 1
named[1068]: zone localhost/IN: loaded serial 1
named[1068]: zone foo.org/IN: loaded serial 2003040101
Apr  8 23:12:46 knoppix named[1068]: running
```

Ou encore avec la commande ps :

```
root:# ps aux | grep named
root  1066  0.0  1.6 10312 2136 ?  S   23:12   0:00 /usr/sbin/named
root  1067  0.0  1.6 10312 2136 ?  S   23:12   0:00 /usr/sbin/named
root  1068  0.0  1.6 10312 2136 ?  S   23:12   0:00 /usr/sbin/named
root  1069  0.0  1.6 10312 2136 ?  S   23:12   0:00 /usr/sbin/named
root  1070  0.0  1.6 10312 2136 ?  S   23:12   0:00 /usr/sbin/named
```

Vous pouvez également faire des tests successifs pour tester la résolution de nom.

#Vérification avec des ping

```
root@ns1:~# ping -c1 ns1.foo.org
PING ns1.foo.org (192.168.90.100): 56 data bytes
64 bytes from 192.168.90.100: icmp_seq=0 ttl=64 time=0.1 ms
--- ns1.foo.org ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 0.1/0.1/0.1 ms
```

```
root@ns1:~# ping -c1 www.foo.org
PING ns1.foo.org (192.168.90.100): 56 data bytes
64 bytes from 192.168.90.100: icmp_seq=0 ttl=64 time=0.1 ms
--- ns1.foo.org ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 0.1/0.1/0.1 ms
```

### 3.2 nslookup, dig

La commande **nslookup** est de moins en moins utilisée, nous la verrons donc pas. Nous allons voir l'utilisation de dig.

Ces commandes sont très largement utilisées par les administrateurs de réseau pour résoudre les problèmes liés aux services de résolution de noms.

Tests avec dig :

# Test sur une zone

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	16 - 27

## Système de fichiers réseau sous Gnu/Linux

```
root@knoppix:/etc/bind# dig any foo.org
; <<>> DiG 9.2.1 <<>> any foo.org
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32752
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; QUESTION SECTION:
;foo.org.                IN      ANY

;; ANSWER SECTION:

foo.org.                604800 IN      SOA     ns1.foo.org.  \
      root.ns1.foo.org. 2003040102 604800 86400 2419200 604800
foo.org.                604800 IN      NS      ns1.foo.org.

;; ADDITIONAL SECTION:
ns1.foo.org.           604800 IN      A       192.168.90.100

;; Query time: 7 msec
;; SERVER: 192.168.90.100#53(192.168.90.100)
;; WHEN: Tue Apr  8 23:30:05 2003
;; MSG SIZE rcvd: 100
```

# Récupération de l'enregistrement SOA d'une zone

```
root@knoppix:/etc/bind# dig soa foo.org

; <<>> DiG 9.2.1 <<>> soa foo.org
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15982
;; flags: qr aa rd ra; QUERY:1, ANSWER:1, AUTHORITY:1, ADDITIONAL:1

;; QUESTION SECTION:
;foo.org.                IN      SOA

;; ANSWER SECTION:

foo.org.                604800 IN      SOA     ns1.foo.org.  \
      root.ns1.foo.org. 2003040102 604800 86400 2419200 604800

;; AUTHORITY SECTION:
foo.org.                604800 IN      NS      ns1.foo.org.

;; ADDITIONAL SECTION:
ns1.foo.org.           604800 IN      A       192.168.90.100

;; Query time: 2 msec
;; SERVER: 192.168.90.100#53(192.168.90.100)
;; WHEN: Tue Apr  8 23:30:43 2003
;; MSG SIZE rcvd: 100
```

#Vérification de la résolution de nom sur www.foo.org

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	17 - 27

## Système de fichiers réseau sous Gnu/Linux

```
root@knoppix:/etc/bind# dig www.foo.org

; <<>> DiG 9.2.1 <<>> www.foo.org
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52961
;; flags: qr aa rd ra; QUERY:1, ANSWER:2, AUTHORITY:1, ADDITIONAL:0

;; QUESTION SECTION:
;www.foo.org.                IN      A

;; ANSWER SECTION:
www.foo.org.                 604800 IN      CNAME   ns1.foo.org.
ns1.foo.org.                 604800 IN      A       192.168.90.100

;; AUTHORITY SECTION:
foo.org.                     604800 IN      NS      ns1.foo.org.

;; Query time: 3 msec
;; SERVER: 192.168.90.100#53(192.168.90.100)
;; WHEN: Tue Apr  8 23:31:49 2003
;; MSG SIZE rcvd: 77
```

# Vérification de la résolution de nom inverse.

```
root@ns1:/etc/bind# dig ptr 100.90.168.192.in-addr.arpa

; <<>> DiG 9.2.1 <<>> ptr 100.90.168.192.in-addr.arpa
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 30642
;; flags: qr aa rd ra; QUERY:1, ANSWER:1, AUTHORITY:1, ADDITIONAL:0
;; QUESTION SECTION:
;100.90.168.192.in-addr.arpa.  IN      PTR

;; ANSWER SECTION:
100.90.168.192.in-addr.arpa. \
                          604800 IN  PTR   ns1.90.168.192.in-addr.arpa.

;; AUTHORITY SECTION:
90.168.192.in-addr.arpa. \
                          604800 IN      NS      ns1.90.168.192.in-addr.arpa.

;; Query time: 7 msec
;; SERVER: 192.168.90.100#53(192.168.90.100)
;; WHEN: Tue Apr  8 23:45:39 2003
;; MSG SIZE rcvd: 77
```

### 3.2.A Le cache du DNS

Le cache permet également de détecter certaines causes d'erreurs. Le problème est qu'il est en mémoire. Pour le récupérer sous la forme d'un fichier utilisez la commande `kill -INT PID de named`. Vous récupérez un fichier `/var/named/named_dump.db` que vous pouvez exploiter.

### 3.2.B Les journaux

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	18 - 27

Si vous êtes en phase de configuration, pensez (ce doit être un réflexe) à consulter les fichiers de journalisation, notamment `/var/log/messages`. Cette opération permet dans bien des cas de corriger des erreurs qui se trouvent dans les fichiers de configuration. Voici comment procéder :

- Arrêt du serveur
- Nettoyage du fichier `> /var/log/messages`
- Démarrage du serveur
- Consultation des logs : `cat /var/log/daemon.log | more`

Pour les fichiers logs, utilisez, si le fichier est trop gros la commande **tail** :

```
# tail -N NomFichier
# Affiche les N dernières lignes d'un fichier
# Par exemple, affiche les 250 dernières lignes d'un fichiers
# tail -n 250 /var/log/daemon.log | more
```

## 4.Remarques

Si vous désirez mettre en place la résolution de noms sur un réseau local, il n'y a pas grand chose de plus à réaliser. Il faut rajouter les enregistrements de type MX pour la messagerie, cette opération sera réalisée pendant la configuration du service de messagerie. Il faut également mettre en place un service de synchronisation des bases de données avec un serveur secondaire pour assurer le service d'un serveur de noms de backup.

Si vous désirez vous relier sur internet, le processus est plus complexe. Il faudra approfondir la description des enregistrements et la structure des fichiers.

Par convention, on considère que chaque domaine dispose d'au moins 1 serveur de noms primaire et un serveur de noms secondaire afin d'assurer une redondance en cas de panne d'un serveur. Les clients réseau seront configurés pour utiliser indifféremment le serveur de noms primaire ou les serveurs de noms secondaires. Il en résulte une duplication de la base de données du DNS primaire sur les serveurs secondaires. La base de données est rafraîchie en fonction des paramètres de l'enregistrement SOA. Ce procédé met en oeuvre un principe de base de données répartie. Vous trouverez quelques éléments dans les annexes qui suivent.

## 5.Annexes

### 5.1 Annexe 1 - extraits de fichiers de configuration

Les extraits ci-dessous d'une zone fictive `foo.org` peuvent servir d'exemple pour bâtir une zone.

Si on respecte les conventions utilisées sur internet, voici ce que l'on devrait avoir :

- le serveur ftp est accessible par l'adresse `ftp.foo.org`
- le serveur http par l'adresse `www.foo.org`
- le serveur de noms primaire par `ns1.foo.org`
- le serveur de messagerie `mail.foo.org`

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	19 - 27

## Système de fichiers réseau sous Gnu/Linux

- le serveur de news `news.foo.org`, etc.

`ftp`, `www`, `mail` sont des alias (*canonical name* ou CNAME) de la machine `ns1` dans le domaine `foo.org`

Nous aurons donc sur le serveur de noms 5 enregistrements dans la zone `foo.org` qui concernent la machine `ns1.foo.org` : un enregistrement de type A pour déclarer `ns1` quatre enregistrements de type CNAME pour la machine `ns1`.

Nous aurons également, dans la zone reverse `in-addr.arpa`, 1 enregistrement de type pointeur (PTR) pour chaque enregistrement de type A dans la zone directe. Enfin, pour le serveur de messagerie, il faut également un enregistrement de type MX.

Tous les fichiers concernant la zone locale, et un fichier `named.conf` sont déjà installés sur votre machine.

```
; db.local
; Résolution directe pour la zone locale
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      localhost. root.localhost. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@         IN      NS       localhost.
@         IN      A        127.0.0.1
```

```
; db.127
; Résolution inverse pour l'adresse de loopback
; BIND reverse data file for local loopback interface
;
$TTL      604800
@         IN      SOA      localhost. root.localhost. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@         IN      NS       localhost.
1.0.0     IN      PTR      localhost.
```

```
; db.0
; Résolution inverse pour la zone de broadcast
; BIND reverse data file for broadcast zone
;
$TTL      604800
@         IN      SOA      localhost. root.localhost. (
```

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	20 - 27

## Système de fichiers réseau sous Gnu/Linux

```

        1          ; Serial
        604800     ; Refresh
        86400     ; Retry
        2419200   ; Expire
        604800 )   ; Negative Cache TTL
;
@      IN      NS      localhost.

; db.255
; Résolution inverse pour la zone de broadcast;
; BIND reverse data file for broadcast zone
;
$TTL   604800
@      IN      SOA     localhost. root.localhost. (
        1          ; Serial
        604800     ; Refresh
        86400     ; Retry
        2419200   ; Expire
        604800 )   ; Negative Cache TTL
;
@      IN      NS      localhost.

; db.root
; fichier des serveurs de noms de l'internet
; vous pouvez le consulter sur votre disque.

; db.foo.org
; fichier directe pour la zone foo.org
; BIND data file for local loopback interface
;
$TTL   604800
@      IN      SOA     ns1 root.ns1 (
        2003040102 ; Serial
        604800     ; Refresh
        86400     ; Retry
        2419200   ; Expire
        604800 )   ; Negative Cache TTL
;
@      IN      NS      ns1
ns1    IN      A       192.168.90.100 ; @ ip du serveur de nom
www    IN      CNAME   ns1
ftp    IN      CNAME   ns1

; db.foo.org.rev
; fichier de résolution inverse pour la zone foo
; BIND data file for local loopback interface
;
$TTL   604800
@      IN      SOA     ns1 root.ns1 (
        2003040102 ; Serial
```

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	21 - 27

## Système de fichiers réseau sous Gnu/Linux

```

        604800      ; Refresh
        86400      ; Retry
        2419200    ; Expire
        604800 )    ; Negative Cache TTL
;
@      IN      NS      ns1
100    IN      PTR     ns1

; fichier named.conf du serveur primaire
// C'est le fichier principal de configuration des DNS
// C'est ici que sont réalisées, pour chaque zone, les déclarations
// des fichiers de ressources.

options {
    directory "/var/cache/bind";
    // Serveurs à prévenir pour les transferts de zone
    forwarders {0.0.0.0;};
};

// Ici les paramètres pour les clés rndc
// Les paramètres doivent être strictement identiques à celui
// du fichier rndc.key ou rndc.conf
// Si vous avez des messages d'erreur à l'utilisation
// de cette commande, vérifier le contenu des fichiers.

key "rndc-key" {
    algorithm      hmac-md5;
    secret "c3Ryb25nIGVub3VnaCBmb3IgySBtYW4gYnV0IG1hZGUgZm9yIGEGd29tYW4K";
};

# Autorisations rndc sur la machine.
controls {
    inet 127.0.0.1 allow {localhost;} keys {"rndc-key";}
};

// Indication pour les serveurs racines
zone "." {
    type hint;
    file "/etc/bind/db.root";
};

// be authoritative for the localhost forward
// and reverse zones, and for
// broadcast zones as per RFC 1912
zone "localhost" {
    type master;
    file "/etc/bind/db.local";
};

zone "127.in-addr.arpa" {
    type master;
    file "/etc/bind/db.127";
};

zone "0.in-addr.arpa" {
```

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	22 - 27

## Système de fichiers réseau sous Gnu/Linux

```
        type master;
        file "/etc/bind/db.0";
};

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

//zone directe de foo
zone "foo.org" {
    type master;
    file "/etc/bind/db.foo.org";
};

//Zone reverse pour 192.168.90.
zone "90.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.foo.org.rev";
};

; fichier name.conf du serveur secondaire
; L'entête de bouge pas
; Tout ce qui concerne localhost non plus car chaque DNS
; est primaire pour les zones locales
; on ajoute la déclaration des autres zones, le fichier
; de stockage et l'adresse IP du serveur primaire pour
; pouvoir réaliser les transferts de zone.

; Déclaration de la zone foo.org

zone "foo.org" {
    type slave;
    file "/etc/bind/db.foo.org";
    masters {192.168.90.1;};
};

zone "90.168.192.in-addr.arpa" {
    type slave;
    file "/etc/bind/db.foo.org.rev";
    masters {192.168.90.1;};
};

; fichier rndc.conf
/* $Id: cours-dns.xml,v 1.10 2004/12/29 18:55:06 jaazzouz Exp $ */
/*
 * Exemple de fichier de rndc.conf, pris pour les TP
 */

options {
    default-server localhost;
    default-key "rndc-key";
};
```

<b>www.ofppt.info</b>	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	23 - 27

## Système de fichiers réseau sous Gnu/Linux

```
server localhost {
    key      "rndc-key";
};

key "rndc-key" {
    algorithm    hmac-md5;
    secret "c3Ryb25nIGVub3VnaCBmb3IgaSBtYW4gYnV0IG1hZGUgZm9yIGEgd29tYW4K";
};

; fichier rndc.key

key "rndc-key" {
    algorithm    hmac-md5;
    secret "49khYQyHfO4AqYO9K7by6Q==";
};
```

### 5.2 Annexe 2 - Serveur primaire et serveur secondaire

Pour configurer le serveur secondaire, vous n'avez pas grand chose à faire. Copiez le fichier `named.conf` du primaire sur le secondaire. Voyez l'exemple ci-dessus. Le dns secondaire téléchargera (processus de transfert de zone) les fichiers de ressources du dns primaire. Attention, le dns secondaire pour une zone est toujours dns primaire pour la zone locale `localhost`.

On remplace la définition `masters` par `slave` sauf pour la zone locale et les fichiers `db.local` et `db.127` qui sont lus localement. Ensuite vous avez rajouté l'adresse du serveur à partir duquel le transfert de zone doit s'effectuer.

Activer les serveurs de noms et analyser les traces (log) sur les 2 serveurs. Corrigez toutes les erreurs jusqu'à ce que cela fonctionne. Vous devriez obtenir la trace selon laquelle il y a eu un transfert de zone entre le serveur maître et le serveur esclave. Exemple :

```
Apr  6 plibre named[8821]: send AXFR query 0 to 195.115.88.38
```

*Expérience 1* : Vous pouvez expérimenter un échange entre un serveur de noms primaire et un serveur esclave. Modifiez sur le serveur primaire le N° de série comme si vous aviez modifié les fichiers de ressources de `ns1` et relancez le service. Vérifiez le transfert de zone a mis à jour la base de données répartie.

*Expérience 2* : Vous pouvez expérimenter une autre procédure d'échange, mais cette fois sans relancer le serveur de noms secondaire. Modifiez d'abord sur les deux serveurs le délai de rafraîchissement et mettez le à 2 ou 3 minute. Relancez les services. Modifiez sur le serveur primaire le N° de série dans l'enregistrement SOA, comme si vous aviez modifié les fichiers de ressources de `ns1` et relancez le service. Si vous attendez, vous verrez la synchronisation s'opérer (trace dans les fichiers de logs). Vous découvrez ainsi le mode de fonctionnement de synchronisation des serveurs de noms sur internet.

*Remarque* : si vous voulez, sur ces serveurs assurer la gestion de plusieurs domaines, il suffit de rajouter les définitions de ressources pour ces domaines, puis de déclarer ces zones dans `/etc/named.conf`.

Notez également que la notion d'autorité est différente de la notion de serveur maître ou serveur esclave.

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	24 - 27

En effet si vous avez en charge la gestion de deux zones (Z1 et Z2), vous pouvez mettre deux serveurs ayant autorité sur ces zones (ns1 et ns2), par contre ns1 peut être serveur maître pour Z1 et secondaire pour Z2, et ns2 peut être serveur maître pour Z2 et esclave pour Z1.

### 5.3 Annexe 3 - Mise en place d'une délégation de zone

Prenons l'exemple d'une zone `sd` d'adresse 192.168.254.0, rattachée à `foo.org`. Nous allons mettre en place une délégation de zone pour `sd`. La résolution des noms de la zone `sd.foo.org` est prise en charge par les serveurs de noms de la zone `sd`, nous n'avons donc pas à nous en occuper. Par contre nous devons déclarer ces serveurs afin de maintenir la cohérence de la hiérarchie.

*Configuration de la délégation* : sur le serveur de noms de la zone `foo.org` il faut rajouter les enregistrements qui décrivent les serveurs de noms de la zone `sd.foo.org` dans le fichier de zone `db.foo.org`.

```
sd                86400  NS      ns1.sd.foo.org.
                  86400  NS      ns2.sd.foo.org.
```

Et les enregistrements qui déterminent les adresses de ces serveurs de noms.

```
ns1.sd.foo.org. IN      A       192.168.254.1
ns2.sd.foo.org. IN      A       192.168.254.2
```

*La délégation de la zone in-addr.arpa* : Dans la pratique, cette délégation est différente car la zone inverse ne dépend pas de la zone supérieure, mais d'une autre entité (in-addr). Le processus est donc un peu différent.

*Pourquoi ?* parce que cette zone reverse est gérée par l'entité qui gère l'espace 192.168.0 à 192.168.255 et il est fort probable que ce n'est pas la zone `foo` qui assure la résolution inverse pour tous les réseaux compris entre 192.168.0 et 192.168.255.

Ceci dit, cela n'empêche pas de réaliser cela sur une maquette. Il est possible de mettre en place cette résolution inverse. Nous allons donc considérer que la zone `foo.org` assure la résolution de noms inverse du réseau 192.168.254. Ce reviendrait à considérer que dans la réalité, la zone `sd` serait un sous domaine de `foo`. La configuration ici est simple, les masques de sous-réseaux utilisés ici sont ceux par défaut (255.255.255.0) pour la classe C. Le principe pour la zone inverse est identique à celui de la zone directe. Il suffit de rajouter dans le fichier `db.0.168.192` les enregistrements suivants :

```
sd.foo.org.                IN NS      ns1.sd.foo.org.
                           IN NS      ns2.sd.foo.org.
1.0.168.192.in-addr.arpa  86400  IN  PTR    ns1.sd.foo.org.
2.0.168.192.in-addr.arpa  86400  IN  PTR    ns2.sd.foo.org.
```

### 5.4 Annexe 3 - Outils de diagnostic et de contrôle

<http://www.dnsreport.com/tools/dnsreport.ch> Le plus complet pour les tests. Il explique assez bien les problèmes et les modifications à faire pour les résoudre. Avec le serveur `eclis`, on peut arriver à n'avoir que 2 warnings: "Multiple MX records" et "Mail server host name in greeting" qui correspondent respectivement au fait que l'on a pas de serveur de mail de secours et que l'on fait du virtual hosting. Note: ce test utilise un des root name serveur au hasard. Il faut faire le test plusieurs fois pour avoir un

www.ofppt.info	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	25 - 27

## Système de fichiers réseau sous Gnu/Linux

aperçu complet de la situation.

<http://www.zonecut.net/dns/> Affiche la situation vue depuis un root name serveur pris au hasard et ce avec une jolie représentation graphique. Là aussi il faut tester plusieurs fois pour avoir un aperçu complet. Les quelques lignes de résumé à la fin sont intéressantes.

<http://www.squish.net/dnscheck/dnscheck.cgi> Le seul qui fait un test exhaustif de tout les root name serveur et qui fait un résumé pertinent à la fin. Il est un peut lent mais ça va finalement plus vite que de tester 10 fois les autres outils pour passer en revue tout les root name serveurs... ça démontre bien, avec des probabilités en pouces, pourquoi on obtient pas toujours les mêmes résultats quand il y a un problème avec les DNS.

<b>www.ofppt.info</b>	Document	Millésime	Page
	Système de fichiers réseau sous Gnu/Linux	août 14	26 - 27