Gestion de zones DNS

1. Gestion de zones locales

a. Création d'un fichier de zone directe

Les informations nécessaires à la résolution devront se trouver dans un fichier de déclaration de zone. L'emplacement de ce fichier est libre, puisqu'il est défini dans une section **zone** de **named.conf**. Toutefois, un usage établi veut que ce fichier soit placé dans le répertoire **/var/named**. Notez que selon les distributions, il peut aussi se trouver dans le répertoire **/etc** ou dans **/etc/bind**.

Ce fichier aura le format très strict indiqué ci-dessous. Dans la plupart des cas, un refus de démarrer est dû à un fichier de zone mal formé. Il est composé des déclarations de durée de vie en cache des informations, du serveur ayant autorité sur la zone, des serveurs de noms desservant cette zone, et de l'ensemble des enregistrements de ressources (RR) de cette zone.

Format type du fichier de zone directe

```
$TTL      ttl
nomzone  IN  SOA  serveur mailadmin (
         serial
         refresh
         retry
         expire
         negative )

nomzone  IN  NS  serveur
```

Fichier de zone directe : format type de l'en-tête	
ttl	Time To Live (durée de vie) : indique la durée de conservation en secondes des données en mémoire cache. Cette valeur est précédée par la directive $TTL.
nomzone	FQDN de la zone gérée par ce fichier. Souvent remplacé par un arobase (@) pour alléger le fichier. Attention, puisqu'il s'agit d'un FQDN, le nom de la zone doit se terminer par un point.
IN	Obsolète mais courant : classe Internet (aucune autre classe n'est plus utilisée).
SOA	Start Of Authority. Enregistrement obligatoire pour indiquer que ce serveur est légitime sur cette zone.
serveur	FQDN du serveur ayant autorité sur la zone.
mailadmin	Adresse e-mail de l'administrateur du serveur. L'arobase étant un caractère réservé dans les fichiers de zone, il est conventionnellement remplacé par un point. admin@saintmarcelin.fr devient donc admin.saintmarcelin.fr.
serial	Valeur numérique. Numéro de série du fichier. Utile quand la zone est répliquée sur d'autres serveurs pour savoir si les données ont changé et si la réplication doit être faite.
refresh	Valeur numérique. Utilisé quand la zone est répliquée. Indique au serveur esclave à quel intervalle tester la validité de sa zone.
retry	Valeur numérique. Utilisé quand la zone est répliquée. S'il est impossible pour l'esclave de contacter le serveur maître, indique au bout de combien de temps réessayer.
expire	Valeur numérique. Utilisé quand la zone est répliquée. S'il est impossible pour l'esclave de contacter le serveur maître, indique au bout de combien de temps les

	enregistrements non rafraîchis perdent leur validité et ne doivent plus être utilisés.
negative	Valeur numérique. Indique combien de temps le serveur doit conserver en cache une réponse négative.
NS	Enregistrement indiquant quel est le serveur de nom pour cette zone.

b. Création d'un fichier de zone inverse

Le fichier de zone inverse aura la même structure qu'un fichier de zone directe. Comme indiqué plus haut, le nom normalisé de la zone est formé par les octets de la partie réseau de l'adresse IP ordonnés en sens inverse, suivi de la chaîne de caractères « .in-addr.arpa ». Par exemple, la zone inverse pour le réseau **192.168.99.0** sera : **99.168.192.in-addr.arpa**, et c'est ce nom qui devra être employé dans le fichier de zone et dans le fichier **named.conf**.

Format type du fichier de zone inverse

```
$TTL    ttl
nomzoninv  IN  SOA  serveur mailadmin (
         serial
         refresh
         retry
         expire
         negative )

nomzoneinv  IN  NS  serveur
```

Fichier de zone inverse : format type de l'en-tête	
nomzoneinv	Nom normalisé de la zone inverse : *subnet-inversé*.in-addr.arpa. Où *subnet-inversé* représente les octets du subnet en ordre inversé. Attention, le nom de la zone inverse est un FQDN, il doit donc se terminer par un point.
SOA	Start Of Authority. Enregistrement obligatoire pour indiquer que ce serveur est légitime sur cette zone.
serveur	FQDN du serveur ayant autorité sur la zone.
NS	Enregistrement indiquant quel est le serveur de nom pour cette zone.

Constatez que c'est rigoureusement la même chose que pour la zone directe. C'est le format des enregistrements qui fait l'essentiel de la différence.

c. Création d'enregistrements dans les fichiers de zone

Une fois les fichiers de zone créés, il suffit d'ajouter autant d'enregistrement de ressource que l'on souhaite, à raison d'un par ligne.

Format d'un enregistrement de ressource dans un fichier de zone directe

```
nom  IN  typeRR  valeur-résolue
```

Format d'un enregistrement de ressource dans un fichier de zone inverse

```
adresse-hôte  IN  PTR  nom
```

Fichier de zone directe : format des enregistrements	
nom	Nom simple ou FQDN auquel il faut faire correspondre une adresse IP.

IN	Obsolète mais nécessaire : classe Internet.
typeRR	Type d'enregistrement. Souvent de type A : fait correspondre une adresse IP à un nom. Valeurs courantes : A, CNAME, MX.
valeur-résolue	Ce à quoi on fait correspondre le nom. Dans le cas d'un enregistrement de type A, une adresse IP.
adresse-hôte	L'octet ou les octets qui associés à l'adresse du réseau de la zone inverse formeront l'adresse IP à résoudre.
PTR	Type pointeur : fait correspondre un nom à une adresse IP. Hors enregistrements SOA et NS, c'est le seul type qu'on rencontre dans les zones inverses.

L'ajout d'un grand nombre d'enregistrements est évidemment fastidieux, et gagnera à être réalisé sous forme de script.

Exemple de script simple d'alimentation d'un fichier de zone :

Les hébergeurs et autres DNS gérant de gros volumes d'enregistrement utilisent naturellement des scripts beaucoup plus élaborés.

```
#!/bin/bash
echo "Nom à ajouter à la zone ?"
read nom
echo "Adresse IP correspondant ?"
read ip
echo "$nom  IN  A  $ip" >> /var/named/saintmarcelin.fr
```

d. Déclaration de zone principale dans le fichier named.conf

Une fois que l'on dispose d'un fichier de zone, il faut faire savoir au serveur qu'il doit le charger au démarrage. Ceci se fera avec une déclaration de zone normalisée dans le fichier **named.conf**.

Format type de la déclaration de zone dans named.conf

```
zone "nomzone" {
    type master;
    file "fichier";
    };
```

Fichier named.conf : directives et syntaxe de la déclaration de zone	
nomzone	Le FQDN de la zone gérée par le serveur.
type master	Précise qu'il s'agit d'une zone maîtresse à synchroniser éventuellement vers des serveurs esclaves.
fichier	Chemin absolu du fichier à lire pour prendre connaissance des éléments propres à la zone (configuration, RR, etc.).

e. Prise en compte de la nouvelle configuration

Il faut ensuite faire en sorte que le serveur DNS recharge ses fichiers de configuration afin de prendre en compte les nouveautés. Deux solutions pour cela : le redémarrage du service ou le chargement de la nouvelle zone par commande de pilotage **rndc**.

Rechargement du service

```
/etc/init.d/bind9 restart
```

Chargement de la nouvelle zone par rndc

```
rndc reload saintmarcelin.fr
```

2. Gestion de zones secondaires

Une zone DNS ne devrait pas dépendre d'un serveur unique et il est courant de créer sur un deuxième serveur des zones secondaires, strictement identiques aux zones primaires, et synchronisées à intervalles réguliers.

a. Déclaration de la zone secondaire dans named.conf

Il n'est évidemment pas nécessaire de créer les fichiers de zones, puisqu'ils seront synchronisés depuis le serveur autoritaire. On parle couramment de serveur maître et de serveurs esclaves.

Le chargement de la zone esclave se fait avec une déclaration de zone normalisée dans le fichier **named.conf**.

Format type de la déclaration de zone secondaire dans named.conf

```
zone "nomzone" {
    type slave;
    masters { adresse_maître ; } ;
    file "fichier";
    };
```

Fichier named.conf : directives et syntaxe de la déclaration de zone	
nomzone	Le FQDN de la zone gérée par le serveur.
type slave	Précise qu'il s'agit d'une zone esclave à synchroniser depuis un serveur maître.
adresse_maître	Adresse IP du serveur autoritaire.
fichier	Chemin absolu du fichier dans lequel stocker les éléments synchronisés. Le compte de service doit avoir les droits d'écriture sur le répertoire de travail.

b. Prise en compte de la nouvelle configuration

Il faut ensuite faire en sorte que le serveur DNS recharge ses fichiers de configuration afin de prendre en compte les nouveautés. Deux solutions pour cela : le redémarrage du service ou le chargement de la nouvelle zone par commande de pilotage **rndc**.

Rechargement du service

```
/etc/init.d/bind9 restart
```

Chargement de la nouvelle zone par rndc

```
rndc reload saintmarcelin.fr
```

3. Délégation de zone

Une délégation de zone consiste à faire gérer par un serveur tiers une zone enfant d'une zone hébergée par un serveur parent. C'est le principe de la délégation qui permet de distribuer l'ensemble de l'espace de nom DNS sur des milliers de serveurs. La délégation se configurera sur le serveur parent.

On ajoutera dans le fichier de zone du parent deux **Ressources Record** : l'un de type **NS** pour indiquer qu'il existe un serveur de nom pour la zone enfant, et l'autre de type **A** pour connaître l'adresse IP de ce serveur de nom. L'enregistrement **NS** assurant la délégation est appelé **glue record** (enregistrement colle).

Configuration de la délégation dans le fichier de la zone parente

```
zone_enfant  IN  NS  dns_enfant
dns_enfant   IN  A   A.B.C.D
```

Éléments	
zone_enfant	Nom simple de la zone enfant.
IN	Obsolète mais obligatoire : classe Internet.
NS	Cet enregistrement est de type Name Server (serveur de nom).
dns_enfant	Nom du serveur DNS qui gère la zone enfant.
A	C'est un enregistrement de type A.
A.B.C.D	Adresse IP du serveur de nom pour la zone enfant.

4. Outils de test

a. ping

Même si ça n'est pas sa fonction première, **ping** peut tout à fait servir de test rudimentaire pour la résolution de noms. On sera alors limité à tester la réponse des serveurs par défaut, renseignés dans **/etc/resolv.conf**.

Utilisation de ping pour tester une résolution de nom

Quand on utilise ping pour tester une résolution de noms, c'est la traduction de l'adresse qui importe et non la réponse ICMP de la machine distante.

```
donald:/etc/bind# ping donald.formation.fr
PING donald.formation.fr (192.168.1.1) 56(84) bytes
64 bytes from donald.formation.fr (192.168.1.1): icmp
64 bytes from donald.formation.fr (192.168.1.1): icmp
64 bytes from donald.formation.fr (192.168.1.1): icmp
```

b. nslookup

nslookup est l'outil le plus populaire pour l'interrogation des serveurs DNS. Il est présent sur la grande majorité des plates-formes Unix et Windows.

nslookup est utilisé la plupart du temps en mode interactif. C'est-à-dire qu'après avoir tapé **nslookup**, on se trouve dans son interface où on tapera des commandes spécifiques. Les serveurs de noms interrogés par défaut sont ceux référencés dans **/etc/resolv.conf**. Ceci pourra éventuellement être modifié par la suite.

Utilisation de nslookup pour une résolution de nom

Par défaut, nslookup adresse aux serveurs DNS des requêtes de type A.

```
donald:/etc/bind# nslookup
> server
Default server: 192.168.1.1
Address: 192.168.1.1#53
> coincoin.formation.fr
Server:         192.168.1.1
Address:        192.168.1.1#53

coincoin.formation.fr   canonical name = donald.formation.fr.
```

```
Name:   donald.formation.fr
Address: 192.168.1.1
>
```

nslookup	
nom	Taper un nom DNS directement dans l'interface nslookup revient à en demander la résolution. nslookup indiquera alors quel serveur DNS il a interrogé, et la réponse qui lui a été faite. Il peut s'agir d'un nom complet (FQDN) ou d'un nom simple si on s'appuie sur un suffixe de recherche défini dans /etc/resolv.conf.
server *A.B.C.D*	La commande server suivie de l'adresse IP d'un serveur à interroger indique à nslookup que toutes les interrogations futures devront être adressées à ce serveur.
set type=*TYPE*	Par défaut, nslookup fait des requêtes de type A (résolution ordinaire de nom en adresse IPv4). La commande set type permet d'adresser des requêtes d'un autre type. On s'en sert couramment pour connaître par exemple les serveurs de noms ou de messagerie associés à une zone.

Utilisation de nslookup pour trouver l'adresse d'un serveur de messagerie

On peut utiliser nslookup pour tous les types d'enregistrements courants (ici MX).

```
donald:/etc/bind# nslookup
> set type=MX
> elysee.org
Server:         192.168.1.1
Address:        192.168.1.1#53

Réponse ne faisant pas autorité :
elysee.org      MX preference = 10, mail exchanger = mail.elysee.org

mail.elysee.org internet address = 64.182.1.213
>
```

c. dig

dig est le nouvel outil proposé par l'ISC pour l'interrogation et le diagnostic des serveurs DNS. Passant pour être le plus précis et abouti des outils de test, il devrait éventuellement finir par s'imposer comme solution de référence. Toutefois, les habitudes prises par les administrateurs DNS laissent présager encore de beaux jours pour nslookup.

dig est utilisé en mode non interactif, c'est-à-dire que chaque utilisation de **dig** devra donner l'ensemble des paramètres nécessaires à la résolution.

Syntaxe simplifiée de dig

```
dig nom

dig A.B.C.D nom TYPE
```

Éléments	
nom	Le nom complet (FQDN) dont on veut assurer la résolution.
A.B.C.D	L'adresse IP du serveur DNS à interroger. En cas d'omission, les serveurs de noms interrogés sont ceux référencés dans /etc/resolv.conf.
TYPE	Par défaut, dig fait des requêtes de type A (résolution ordinaire de nom en adresse IPv4). Le paramètre type s'il est précisé permet d'adresser des requêtes d'un autre type. On s'en sert couramment pour connaître par exemple les serveurs de noms ou de messagerie associés à une zone.

Exemple d'utilisation de dig

De loin la plus précise des commandes de diagnostic DNS.

```
donald:/etc/bind# dig @127.0.0.1 coincoin.formation.fr

; <<>> DiG 9.2.4 <<>> @127.0.0.1 coincoin.formation.fr
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 18067
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;coincoin.formation.fr.         IN      A

;; ANSWER SECTION:
coincoin.formation.fr.  86400   IN      CNAME   donald.formation.fr.
donald.formation.fr.    86400   IN      A       192.168.1.1

;; AUTHORITY SECTION:
formation.fr.           86400   IN      NS      donald.formation.fr.

;; Query time: 9 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Thu Jun 15 19:49:45 2006
;; MSG SIZE  rcvd: 90
```

d. host

host est un outil simple pour faire une requête DNS en mode non interactif.

Syntaxe simplifiée pour la commande host

host *nom*

host *nom* -t *type A.B.C.D*

Éléments	
nom	Le nom DNS dont il faut assurer la résolution. Il peut s'agir d'un FQDN ou du nom simple qui sera complété par le suffixe de recherche s'il est défini dans /etc/resolv.conf.
-t *type*	Facultatif : le type de requête qui est adressée. Par défaut le type est sélectionné automatiquement parmi les types A, AAAA et MX.
A.B.C.D	Facultatif : l'adresse IP du serveur DNS à interroger. Si cet élément n'est pas renseigné, ce sont les serveurs présents dans /etc/resolv.conf qui sont utilisés.

Utilisation de host pour tester une résolution de nom

host présente un résultat concis.

```
donald:/etc/bind# host coincoin.formation.fr
coincoin.formation.fr is an alias for donald.formation.fr.
donald.formation.fr has address 192.168.1.1

donald:/etc/bind#
```

Utilisation de host pour récupérer les enregistrements NS

```
donald:/etc/bind# host -t NS formation.fr
formation.fr name server donald.formation.fr.
```

```
donald:/etc/bind#
```

e. Mesure des performances

La commande **time** qui mesure le temps consommé par une application permet de mesurer la performance d'une résolution DNS. Elle indique le temps total consommé par la commande, et le temps consommé par les processus dans les espaces d'exécution système et utilisateur.

Observation du temps pris par une résolution DNS

Les temps mesurés dépendent de la bande passante disponible, de la disponibilité du serveur, et de la rapidité de la machine cliente.

```
toto@serveur:~$ time nslookup www.eni-editions.fr
Server:    127.0.0.1
Address:  127.0.0.1#53

Non-authoritative answer:
Name:  www.eni-editions.fr
Address: 81.80.245.20


real  0m0.256s
user  0m0.000s
sys   0m0.010s
toto@serveur:~$
```