Serveur DNS sous Linux

I Présentation (pour l'installation passer directement à la page 5)

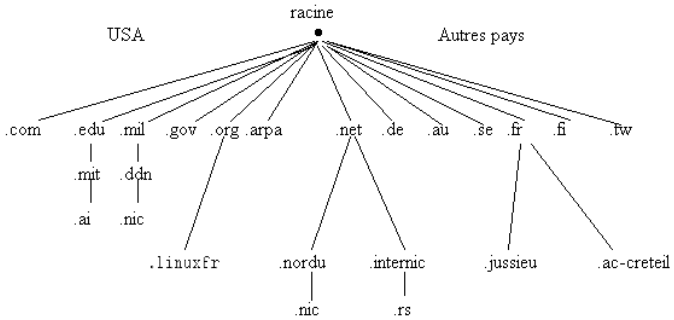
I.1 Définition

Le Domain Name System (ou DNS, système de noms de domaine) est un service permettant de traduire un nom de domaine en informations de plusieurs types qui y sont associées, notamment en adresses IP de la machine portant ce nom. Les fonctionnalités du DNS sont :

- ✗ Traduction du nom Internet en adresse IP ;
- ✗ Traduction "reverse" : de l'adresse IP vers le nom ;
- ✗ Définition d'alias de noms de machines ;
- ✗ Support au routage de messagerie (non étudié dans ce document).

I.2 Nommage

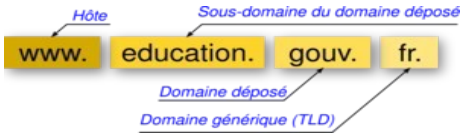
L'espace adresse DNS a une structure en arbre :



Un **domaine** est un des sous-arbres de l'arborescence, par exemple : "google.fr" ou "ac-creteil.fr".

Un domaine et ses sous-domaines forment une zone.

Le **Fully Qualified Domain Name** (FQDN) est la combinaison du nom de machine et de son nom de domaine (exemple : mail.ac-creteil.fr).



Aucune distinction n'est faite entre majuscules et minuscules.

I.3 Configuration du nom du serveur DNS sur un PC

Vérification du nom d'hôte et du nom de domaine

Tapez les commandes suivantes pour vérifier successivement :

- ✗ Le nom de la machine
- ✗ Le nom du domaine
- ✗ Le FQDN (nom de la machine + son nom de domaine)

```
root@debian:~# hostname --short
serveur
root@debian:~# hostname --domain
mydomain.lan
root@debian:~# hostname --fqdn
serveur.mydomain.lan
```

Détermination du serveur DNS utilisé par votre machine

Vous avez plusieurs méthodes pour déterminer le serveur DNS utilisé :

1. Grâce à la commande nslookup :

```
root@serveurDebian:~# nslookup www.google.fr
Server:         212.27.40.240
Address:        212.27.40.240#53

Non-authoritative answer:
Name:   www.google.fr
Address: 216.58.213.3
```

2. Grâce la commande host :

```
root@serveurDebian:~# host -v -t A www.google.fr
Trying "www.google.fr"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 51096
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;www.google.fr.                 IN      A

;; ANSWER SECTION:
www.google.fr.          871     IN      A       216.58.213.3

Received 47 bytes from 212.27.40.240#53 in 72 ms
```

Dans les deux cas, vous trouve que le serveur DNS a pour adresse 212.27.40.240. Le #53 indique sur le port TCP 53 est utilisé pour la communication avec le serveur.

Configuration du fichier « /etc/resolv.conf »

Ce fichier permet d'indiquer d'indiquer le ou les domaines de recherche et les différents serveurs DNS à utiliser.

Par exemple, dans un réseau local, nous pourrions avoir un serveur DNS à l'adresse 192.168.0.1 chargée de gérer le domaine « mon-domaine.local ». En cas de défaillance du DNS local, nous pourrions faire appel aux serveurs DNS de notre fournisseur d'accès. Dans ce cas, le contenu du fichier « **/etc/resolv.conf** », pourrait ressembler à cela :

```
nameserver 192.168.0.1
nameserver 212.27.53.252
nameserver 212.27.52.252
search mon-domaine.local
```

La première ligne indique l'adresse du serveur DNS du réseau local. En cas de défaillance de ce serveur, les serveurs suivants seront utilisés (Serveurs du fournisseur d'accès à Internet dans cet exemple).

La dernière ligne permet d'indiquer le nom du domaine géré par le serveur DNS local. Par exemple, si nous cherchons à contacter le serveur « MonServeur », le système cherchera en fait à contacter l'adresse complète « MonServeur.mon-domaine.local », car le nom du serveur indiqué ne comportait pas le domaine de recherche.

Utilisation de resolvconf

Le paquet « resolvconf » permet de gérer très finement le contenu du fichier « /etc/resolv.conf » servant à la résolution des noms en fonction du type de connexion utilisé et en récupérant les informations à différents endroits statiques ou dynamiques (clients ppp, dhcp ou autres).

L'installation de ce paquet se fait grâce à la commande suivante :

```
root@debian:~# apt-get install resolvconf
```

Une fois le paquet « resolvconf » installé, il ne faut plus modifier le fichier « /etc/resolv.conf », car le contenu de celui-ci sera automatiquement géré et remplacé par « resolvconf ».

Dans le cas où le système a une interface de type « ens33 », la configuration est faite dans le fichier « /etc/network/interfaces ». Il faut ajouter les lignes « dns-nameservers » et « dns-search » au contenu du fichier « **/etc/network/interfaces** ».

Exemple :

```
iface ens33 inet dhcp
dns-nameservers 192.168.0.1 212.27.53.252 212.27.52.252
dns-search mon-domaine.local
```

Pour vérifier que tout fonctionne correctement, il faut commencer par désactiver l'interface « eth0 » :

```
root@debian:~# ifdown ens33
```

Après cette commande, le fichier « /etc/resolv.conf » doit être vidé et devrait donc ressembler à cela :

```
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by
resolvconf(8)
# DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
```

Et après l'activation de l'interface « eth0 » :

```
root@debian:~# ifup ens33
```

Le fichier « **/etc/resolv.conf** » devrait ressembler à cela :

```
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by
resolvconf(8)
# DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 192.168.0.1
nameserver 212.27.53.252
nameserver 212.27.52.252
search mon-domaine.local
```

Remarque : Il est possible de personnaliser le message d'avertissement des deux premières lignes (pour le mettre par exemple en français) en modifiant le contenu du fichier « /etc/resolvconf/resolv.conf.d/head ».

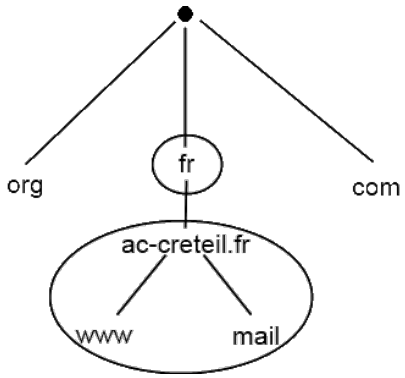
II Principe du Domain Name Service

II.1 Serveurs de noms

Les machines qui stockent les informations sur l'espace de noms sont appelées des **serveurs de noms**. Les serveurs de noms disposent de toutes les informations concernant une partie de l'espace de noms, appelée **zone**. On dit que le serveur de noms a l'autorité (authority) pour la zone.

La différence entre une zone et un domaine est qu'un domaine est souvent découpé en unités plus petites, les zones. Le domaine "fr" est divisé en de nombreuses zones dont "ac-creteil.fr". La zone "fr" contient essentiellement les informations de délégation aux sous-domaines de "fr".

Le sous-domaine "ac-creteil.fr" est lui-même sous-divisé en zones.

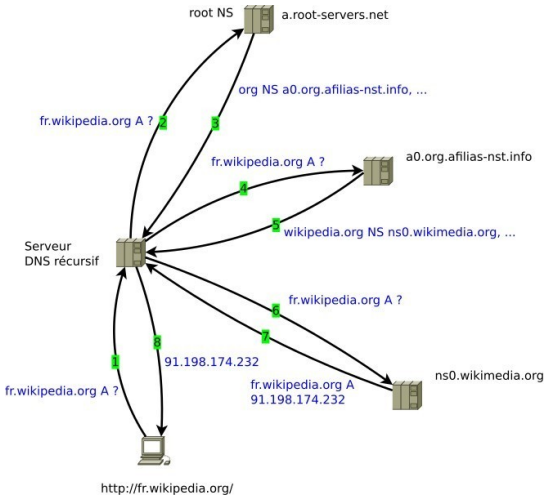


II.2 Traduction du nom

Le DNS se base sur une **architecture répartie** : les informations de nommage sont réparties sur une multitude de serveurs à travers les réseaux Internet. Le DNS est donc l'ensemble des services de nommage : les serveurs de noms. Ils sont répertoriés et peuvent assurer la "couverture" de plusieurs domaines, être serveurs de secours... Chaque domaine possède au moins un serveur de noms.

Soit une application désirant se connecter à une machine distante dont le nom est "fr.wikipedia.org".

1. L'application demande une résolution de nom pour ouvrir une connexion vers une adresse IP correspondante à "fr.wikipedia.org". Le "resolver" local transmet la requête au serveur de noms du domaine .
2. Le DNS demande au serveur de la racine (".") la résolution du nom "fr.wikipedia.org".
3. Le serveur de nom de la racine retourne l'adresse IP du serveur de noms ayant autorité sur ".org".
4. Le DNS demande au serveur de ".org" la résolution du nom "fr.wikipedia.org".
5. Le serveur de noms de ".org" retourne l'adresse IP du serveur de noms ayant autorité sur "wikipedia.org".
6. Le DNS demande au serveur de "wikipedia.org" la résolution du nom "fr.wikipedia.org".
7. Le serveur de noms de "wikipedia.org" retourne l'adresse IP de la machine "fr.wikipedia.org".
8. Le DNS retourne au "resolver" de la machine l'adresse obtenue.
9. L'application ouvre la connexion désirée.



II.3 Modes récursif et itératif

Les serveurs n'ont pas tous la même tâche à effectuer. Le premier serveur contacté doit renvoyer l'adresse complète, alors que les autres ne donnent qu'une réponse partielle, ils ne renvoient que la meilleure réponse dont ils disposent.

Pourquoi cette différence ? Le resolver n'a pas la capacité de suivre une référence. Il a donc envoyé une requête récursive.

Il existe deux types de requêtes DNS : les requêtes récursives ou itératives.

- ✗ **Récursive** : on accède si nécessaire aux différents serveurs DNS (racine, .fr, ac-creteil.fr) pour connaître l'adresse IP de la machine ; le serveur enverra des requêtes itératives aux autres serveurs et il renverra une réponse complète.
- ✗ **Itérative** : on demande une adresse IP connue par le serveur DNS.

Un serveur qui reçoit une requête récursive pourrait lui-même envoyer une requête récursive à un autre serveur, mais les mises en œuvre actuelles sont courtoises et utilisent la seconde méthode.

II.4 La recherche inverse

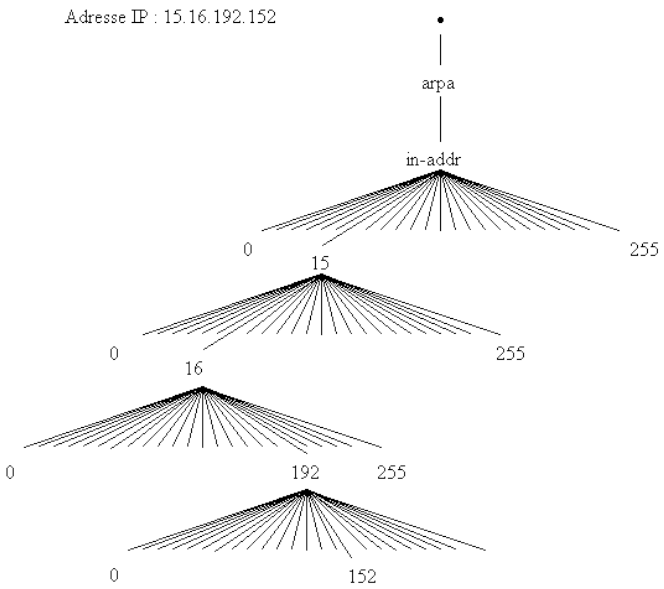
La résolution d'une adresse vers un nom est utilisée pour produire des affichages compréhensibles (dans les fichiers traces par exemple).

Cette correspondance est aisée sous Unix grâce aux fichiers .rhosts et hosts.equiv. Mais avec le DNS, cette correspondance n'est pas aussi simple.

La solution adoptée consiste à créer une section de l'espace de nommage qui utiliserait des adresses comme noms. Cette section est le domaine in-addr.arpa.

Les nœuds du domaine in-addr.arpa sont composés des numéros de la représentation en décimal pointé des adresses IP.

Le domaine in-addr.arpa pourrait avoir 256 sous-domaines :



L'espace de noms de in-addr.arpa ne sert que pour les correspondances de l'adresse vers le nom.

La recherche exhaustive dans toute l'arborescence n'est possible dans une certaine mesure et s'appelle la recherche inverse. Elle ne peut être exécutée que par le serveur de noms qui reçoit la requête. Ce serveur effectue une recherche dans ses données locales. S'il ne peut trouver la réponse, il se contente de le signaler, aucune tentative de passage de requête vers un autre serveur n'est réalisée.

III Installation du serveur

III.1 Installation du serveur

Il faudra avant toute chose rechargez la liste des paquets, pour être sûr que tout est propre :

```
root@debian:~# apt-get update
```

Puis installez le serveur :

```
root@debian:~# apt-get install bind9
```

Acceptez la résolution des dépendances en appuyant sur « Entrée ».

III.2 Test du serveur DNS par défaut

Exécutez une recherche en demandant à votre serveur :

```
root@debian:~# host -v -t A www.google.fr 127.0.0.1
```

On remarque que le dernier paramètre de la commande host est « 127.0.0.1 », c'est pour forcer la demande vers un serveur DNS particulier, car par défaut, la machine utilise celui configuré dans le fichier « /etc/resolv.conf » (voir doc *Configuration des interfaces reseau partie V*).

Vous pouvez aussi utiliser la commande :

```
root@debian:~# nslookup www.google.fr 127.0.0.1
```

On se rend compte que notre serveur DNS bind9, est capable tout seul de résoudre des FQDN d'internet.

Comment cela se fait-il que ça marche tout seul ? Lors de l'installation du serveur ont été installé les fichiers qui permettent à votre serveur de faire une recherche récursive vue ci-dessus.

Vous pouvez voir tous les fichiers installés :

```
root@debian:~# cd /etc/bind
root@debian:~# ls -l
```

C'est très bien, mais pas suffisant pour faire son travail de résolution de noms à l'intérieur de votre réseau.

IV Configuration du serveur de nom du réseau local

IV.1 Configuration par défaut pour la suite de la configuration suivez les étapes 1,2,3, 4 et 5.

Le fichier principal de la configuration du serveur DNS est /etc/bind/named.conf » qui fait lui-même appel à trois autres fichiers de configuration :

- ✗ **named.conf.options**
- ✗ **named.conf.local**
- ✗ **named.conf.default-zones**

Le fichier named.conf ne devrait (sur Debian) jamais être modifié, sauf par les mises à jour futures de la distribution.

named.conf.options

Il n'y a rien à dire sur ce fichier, il contient la configuration et options du serveur. Il peut permettre de définir un serveur DNS de forward.

named.conf.local

Idem par défaut, tout est commenté.

named.conf.default-zones

Dans ce fichier, on trouve toutes les déclarations des zones.

- ✗ Destinées à la résolution normale :

```
« . »
« localhost »
```

- ✗ Destinées à la résolution inverse (qui permet d'obtenir un FQDN à partir d'un adresse IP) :

```
« 127.in-addr.arpa »
« 0.in-addr.arpa »
« 255.in-addr.arpa »
```

Chacune de ces zones fait référence à un fichier de configuration.

La zone "localhost" : /etc/bind/db.local

Pas bien utile en général, elle permet de résoudre localhost. Nous y apprenons que localhost dispose des adresses 127.0.0.1 en IPv4 et ::1 en Ipv6.

La zone "." : /etc/bind/db.root

Plus intéressant est le fichier « db.root », il contient en effet toutes les informations sur les root-servers, sans quoi, notre bind n'aurait rien pu faire. Un serveur racine du DNS (root-server) est un

serveur DNS qui répond aux requêtes qui concernent les noms de domaine de premier niveau (top-level domain, TLD) et qui les redirige vers le serveur DNS de premier niveau concerné.

Le contenu de ce fichier évolue peu dans le temps et les mises à jour de la distribution suffisent normalement à le maintenir dans un état satisfaisant.

IV.2 Configuration pour les adresses locales

Supposons que le serveur DNS écoute sur les interfaces correspondantes aux adresses 127.0.0.1 et ens33 (par exemple 192.168.20.2).

Nous allons maintenant définir une zone pour notre intranet avec un nom de domaine comme par exemple « mydomain.lan ». Nous allons donc créer une zone de recherche directe pour le domaine « mydomain.lan » et une zone de recherche inversée pour le sous-réseau « 192.168.20.0 ».

Remarque personnelle : pour les fichiers de zones il est possible de faire une copie de db.empty pour débuter.

Déclaration des zones locales « /etc/bind/named.conf.local »

Etape 1 : Pour configurer les deux zones pour les recherches directe et inversée, il faut modifier le fichier « /etc/bind/named.conf.local » afin de les définir.

Exemple : le domaine s'appelle mydomain.lan et l'adresse du réseau est 192.168.20.0/24.

```
zone "mydomain.lan" {
   type master;
   file "/etc/bind/db.mydomain.lan";
};

zone "20.168.192.in-addr.arpa" {
   type master;
   file "/etc/bind/db.192.168.20";
};
```

Explication : Le nom de la zone correspond au nom du domaine. Le type est master car il s'agit du serveur principal et non du secondaire (qui est utilisé en cas de panne du serveur principal).

Les fichiers de configuration de zone

La zone sera définie dans les fichiers « db.mydomain.lan » et « db.192.168.20 » :

- ✗ Le fichier « db.mydomain.lan » établit la correspondance nom → adresse ;
- ✗ Le fichier « db.192.168.20 » établit la correspondance adresse → nom.

Ces fichiers contiennent essentiellement des enregistrements ressources dont les plus communs sont :

- ✗ Enregistrement de type A (Address) qui fait correspondre une adresse IP à un nom de machine.

  Exemple :

```
www     IN     A       1.2.3.4
```

- ✗ Enregistrement de type CNAME (Alias) utilisé pour créer un alias depuis un enregistrement de type A

  Exemple :

```
mail    IN     CNAME   www
```

- ✗ Enregistrement MX (Mail Exchange) utilisé pour définir vers quel serveur de la zone un email à destination du domaine doit être envoyé, et avec quelle priorité.

  Exemple :

```
IN      MX  10     mail.ubuntu-fr.lan.
```

```
Mail    IN     A        1.2.3.4
```

- ✗ Enregistrement NS (Name Server) utilisé pour définir quels serveurs répondent pour cette zone.

Définition de la zone directe du domaine « /etc/bind/db.mydomain.lan »

Etape 2 : Dans le dossier /etc/bind effectuer la commande suivante : cp db.local db.mydomain.lan puis ouvrez le fichier db.mydomain.lan avec l'utilitaire nano et le remplir de la manière suivante :

```
ns1.mydomain.lan. : fqdn du serveur dns.
root.mydomain.lan. : adresse mail admin réseau exemple : toto.commercial1.lan.
```

```
$TTL 86400
@ IN SOA ns1.mydomain.lan. root.mydomain.lan. (
     2016021301   ; se = serial number
     172800 ; ref = refresh = 2d
     900    ; ret = update retry = 15m
     1209600; ex = expiry = 2w
     3600    ; min = minimum = 1h
);
@ IN NS  ns1.mydomain.lan.
ns1  A 192.168.20.2
www  A 192.168.20.3
```

Le champ SOA comprend :

- ✗ @ qui désigne la zone décrite par le fichier de configuration ;
- ✗ IN qui signifie que l'on a affaire à une zone Internet(presque toujours le cas) ;
- ✗ le serveur DNS qui dispose du fichier zone de référence (important lorsque que l'on a des DNS secondaires) ; `ns1.mydomain.lan.`
- ✗ l'adresse email de la personne responsable de la zone.`root`

Il faut faire attention à deux choses :

- ✗ Le nom de domaine se finit par un point.
- ✗ L'adresse mail de l'administrateur ne contient pas de « @ » car c'est un symbole qui a une signification particulière pour bind9.

Les valeurs entre les parenthèses sont respectivement :

- ✗ le numéro de série (souvent on met la date courante suivie d'un numéro d'ordre) AAAAMMJJxx ;
- ✗ le temps de rafraîchissement (refresh; ici, 48 heures); la valeur recommandée est de 24 heures;
- ✗ le temps entre deux essais (retry; ici, 15 m); la valeur recommandée est de 2 heures;
- ✗ le temps d'expiration (expire; ici,2 semaines); la valeur recommandée est de 1000 heures ;
- ✗ la valeur TTL minimum (minimum; ici,1 heure); la valeur recommandée est de 2 jours.

Reprenons les trois dernières lignes :

```
@ IN NS  ns1.mydomain.lan.
ns1  A 192.168.20.2
www  A 192.168.20.3
```

- ✗ La première ligne indique que la machine « ns1.mydomain.local. » est un serveur de nom dans la zone, attention au point final.
- ✗ La deuxième ligne fait la correspondance entre le nom du serveur et son adresse IP.
- ✗ La troisième ligne fait la correspondance entre le nom d'une autre machine (www) et son adresse IP.

Remarque : La seule adresse définie est celle du serveur DNS. Il faudrait ajouter les lignes pour les autres ordinateurs du réseau.

Définition de la zone inversée du sous-réseau « /etc/bind/db.192.168.20 »

Etape 3 : Dans le dossier /etc/bind effectuer la commande suivante : cp db.local db.192.168.20 puis remplir le fichier db.192.168.2.0 de la manière suivante :

```
$TTL 86400
@ IN SOA ns1.mydomain.lan. root.mydomain.lan.
  ( 2016021301   ; se = serial number
    172800 ; ref = refresh = 2d
    900    ; ret = update retry = 15m
    1209600; ex = expiry = 2w
    3600    ; min = minimum = 1h
);
@ IN NS  ns1.mydomain.lan.
2 PTR  ns1.mydomain.lan.
3 PTR  www.mydomain.lan.
```

Attention on ne met uniquement que le numéro d'hôte : ex 2 ou 3

L'avant-dernière ligne indique que la machine ns1 correspond à l'adresse 192.168.20.2 car le premier caractère correspond aux poids faibles de l'adresse IP et « 192.168.20. » est donné par la zone indiquée dans le fichier « named.conf.local ». La dernière ligne indique donc que la machine www correspond à l'adresse 192.168.20.3.

Il faut également ajouter les déclarations pour les autres adresses.

Définition des interfaces d'écoute du serveur « /etc/bind/named.conf.options »

Étape 4

Ce fichier est modifié pour indiquer qu'il faut écouter sur les interfaces locales uniquement (IPV4).

```
options {
  directory "/var/cache/bind";

  dnssec-validation auto;



  listen-on-v6 { none; };
  listen-on { 127.0.0.1; 192.168.20.2; };
};
```

IV.3 Test du fonctionnement

Pour que soit pris en compte la nouvelle configuration, après chaque modification des fichiers de configuration, il faut redémarrer le serveur :

```
root@debian:~# systemctl restart bind9
```

Pour vérifier l'état du serveur (s'il y a du rouge, c'est pas bon !)

```
root@debian:~# systemctl -l status bind9
```

En cas d'erreur, vous pouvez grâce aux indications de la commande « systemctl -l status », la ou les déterminer. Si ce n'est pas le cas, vous pouvez avoir plus d'informations sur les erreurs en entrant la commande suivante après de redémarrage du service :

```
root@debian:~# tail /var/log/syslog
```

Si vous ne voyez pas le message d'erreur, vous pouvez augmenter le nombre de lignes affichées grâce à l'option -n, par exemple, pour avoir 20 lignes :

```
root@debian:~# tail -n 20 /var/log/syslog
```

Etape 5 Vous devez modifier le fichier « /etc/network/interfaces » afin d'indiquer que l'adresse du serveur DNS et le nom du domaine :

```
  dns-nameservers 127.0.0.1
  dns-search  mydomain.lan
```

Puis redémarrer le service networking

```
root@debian:~# systemctl restart networking
```

Maintenant vous pouvez utiliser nslookup pour vérifier le fonctionnement du serveur DNS :

```
root@debian:~# nslookup
> ns1
Server:         127.0.0.1
Address:        127.0.0.1#53

Name:   ns1.mydomain.lan
Address: 192.168.20.2
> 192.168.20.2
Server:         127.0.0.1
Address:        127.0.0.1#53

2.20.168.192.in-addr.arpa       name = ns1.mydomain.lan.
> exit
```

La résolution est bien exécutée en direct et en inverse.

Il faudra maintenant définir ns1 comme serveur DNS sur les machines de votre domaine.