

Dr. ATTA Ferdinand	<u>Procédure d'installation d'un serveur DNS avec BIND9</u>	L3 ASSRI 24/02/2023
-----------------------	--	------------------------

But :

Dans cette procédure, nous allons voir comment installer un serveur DNS avec l'outil bind9 sur une distribution Ubuntu. Les commandes sont les mêmes pour Debian

Nous prendrons comme l'utilisation du domaine : **mondomaine.lan**

Notre serveur primaire aura une IP configurée statiquement en **192.168.1.10 /24**

Téléchargement des paquets nécessaires

1. Mettez les bons dépôts dans le fichier /etc/apt/sources.list. Normalement vous devez avoir quelque chose qui ressemble au screenshot ci-dessous :

```
jordan@debian: ~
GNU nano 2.2.6      Fichier : /etc/apt/sources.list

# deb cdrom:[Debian GNU/Linux 8.2.0 _Jessie_ - Official amd64 CD Binary-1 20150906-11:13]/ jessie main
#deb cdrom:[Debian GNU/Linux 8.2.0 _Jessie_ - Official amd64 CD Binary-1 20150906-11:13]/ jessie main

deb http://ftp.fr.debian.org/debian/ jessie main
deb-src http://ftp.fr.debian.org/debian/ jessie main

deb http://security.debian.org/ jessie/updates main
deb-src http://security.debian.org/ jessie/updates main

# jessie-updates, previously known as 'volatile'
deb http://ftp.fr.debian.org/debian/ jessie-updates main
deb-src http://ftp.fr.debian.org/debian/ jessie-updates main
```

2. Télécharger **bind9** (serveur DNS) grâce à la commande :

apt-get install bind9

```
jordan@debian: ~
D  paquetage de libisc95 (1:9.9.5.dfsg-9+deb8u6) sur (1:9.9.5.dfsg-9+deb8u3) ...
Pr  paration du d  paquetage de .../libdns100_1  3a9.9.5.dfsg-9+deb8u6_amd64.deb ...
D  paquetage de libdns100 (1:9.9.5.dfsg-9+deb8u6) sur (1:9.9.5.dfsg-9+deb8u3) ...
Pr  paration du d  paquetage de .../libisc90_1  3a9.9.5.dfsg-9+deb8u6_amd64.deb ...
D  paquetage de libisc90 (1:9.9.5.dfsg-9+deb8u6) sur (1:9.9.5.dfsg-9+deb8u3) ...
Pr  paration du d  paquetage de .../libisc90_1  3a9.9.5.dfsg-9+deb8u6_amd64.deb ...
D  paquetage de libisc90 (1:9.9.5.dfsg-9+deb8u6) sur (1:9.9.5.dfsg-9+deb8u3) ...
Pr  paration du d  paquetage de .../liblwres90_1  3a9.9.5.dfsg-9+deb8u6_amd64.deb ...
D  paquetage de liblwres90 (1:9.9.5.dfsg-9+deb8u6) sur (1:9.9.5.dfsg-9+deb8u3) ...
Pr  paration du d  paquetage de .../libbind9-90_1  3a9.9.5.dfsg-9+deb8u6_amd64.deb ...
D  paquetage de libbind9-90 (1:9.9.5.dfsg-9+deb8u6) sur (1:9.9.5.dfsg-9+deb8u3) ...
S  lection du paquet bind9utils pr  c  demment d  s  lectionn  .
Pr  paration du d  paquetage de .../bind9utils_1  3a9.9.5.dfsg-9+deb8u6_amd64.deb ...
D  paquetage de bind9utils (1:9.9.5.dfsg-9+deb8u6) ...
S  lection du paquet bind9 pr  c  demment d  s  lectionn  .
Pr  paration du d  paquetage de .../bind9_1  3a9.9.5.dfsg-9+deb8u6_amd64.deb ...
D  paquetage de bind9 (1:9.9.5.dfsg-9+deb8u6) ...
Traitement des actions diff  r  es (    triggers     ) pour man-db (2.7.0.2-5)   ...
Traitement des actions diff  r  es (    triggers     ) pour systemd (215-17+deb8u2)   ...
Param  trage de libisc95 (1:9.9.5.dfsg-9+deb8u6) ...
Param  trage de libdns100 (1:9.9.5.dfsg-9+deb8u6) ...
Param  trage de libisc90 (1:9.9.5.dfsg-9+deb8u6) ...
Param  trage de libisc90 (1:9.9.5.dfsg-9+deb8u6) ...
Param  trage de libbind9-90 (1:9.9.5.dfsg-9+deb8u6) ...
Param  trage de liblwres90 (1:9.9.5.dfsg-9+deb8u6) ...
Param  trage de bind9-host (1:9.9.5.dfsg-9+deb8u6) ...
Param  trage de dnsutils (1:9.9.5.dfsg-9+deb8u6) ...
Param  trage de bind9utils (1:9.9.5.dfsg-9+deb8u6) ...
Param  trage de bind9 (1:9.9.5.dfsg-9+deb8u6) ...
Ajout du groupe     bind      (GID 122) ...
Fait.
Ajout de l'utilisateur syst  me     bind      (UID 113) ...
Ajout du nouvel utilisateur     bind      (UID 113) avec pour groupe d'appartenance     bind      ...
Le r  pertoire personnel     /var/cache/bind      n'a pas   t   cr    .
wrote key file "/etc/bind/rndc.key"
#
Traitement des actions diff  r  es (    triggers     ) pour libc-bin (2.19-18+deb8u1)   ...
Traitement des actions diff  r  es (    triggers     ) pour systemd (215-17+deb8u2)   ...
root@debian:/home/jordan#
```

Dr. ATTA Ferdinand	<u>Procédure d'installation d'un serveur DNS avec BIND9</u>	L3 ASSRI 24/02/2023
-----------------------	---	------------------------

3. Modifier votre fichier /etc/resolv.conf

```

GNU nano 2.2.6      Fichier : /etc/resolv.conf      Modifié
domain mondomaine.lan
search mondomaine.lan
nameserver 127.0.0.1
nameserver 192.168.133.138_

```

[^]G Aide [^]O Écrire [^]R Line fich. [^]Y Page préc. [^]K Couper [^]C Pos. cur.
[^]X Quitter [^]J Justifier [^]W Chercher [^]V Page suiv. [^]U Coller [^]T Orthograp.

On spécifie le domaine dans lequel on évolue : **mondomaine.lan**, et on indique ici quels sont les serveurs de résolution de noms (DNS) que l'on souhaite utiliser, à savoir lui-même (127.0.0.1 et 192.168.1.10 qui sont les IPs de notre serveur)

Dr. ATTA Ferdinand	<u>Procédure d'installation d'un serveur DNS avec BIND9</u>	L3 ASSRI 24/02/2023
-----------------------	--	------------------------

Configuration du serveur primaire DNS Bind9

1. Aller dans le dossier **/etc/bind**. Vous devez avoir les fichiers suivants :

```
root@debian:/home/jordan# cd /etc/bind
root@debian:/etc/bind# ls -l
total 52
-rw-r--r-- 1 root root 2389 mars  8 2016 bind.keys
-rw-r--r-- 1 root root  237 mars  8 2016 db.0
-rw-r--r-- 1 root root  271 mars  8 2016 db.127
-rw-r--r-- 1 root root  237 mars  8 2016 db.255
-rw-r--r-- 1 root root  353 mars  8 2016 db.empty
-rw-r--r-- 1 root root  270 mars  8 2016 db.local
-rw-r--r-- 1 root root 3048 mars  8 2016 db.root
-rw-r--r-- 1 root bind  463 mars  8 2016 named.conf
-rw-r--r-- 1 root bind  490 mars  8 2016 named.conf.default-zones
-rw-r--r-- 1 root bind  165 mars  8 2016 named.conf.local
-rw-r--r-- 1 root bind  890 sept. 22 14:57 named.conf.options
-rw-r----- 1 bind bind   77 sept. 22 14:57 rndc.key
-rw-r--r-- 1 root root 1317 mars  8 2016 zones.rfc1918
root@debian:/etc/bind#
```

2. Créer les databases pour notre domaine **mondomaine.lan** :

cp db.local db.mondomaine.lan

cp db.127 db.mondomaine.lan.inv

Dr. ATTA Ferdinand	<u>Procédure d'installation d'un serveur DNS avec BIND9</u>	L3 ASSRI 24/02/2023
-----------------------	--	--------------------------------------

3. On configure notre fichier **named.conf.local** comme ci-dessous :

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

zone "192.168.133.in-addr.arpa" {
    type master;
    file "/etc/bind/db.mondomaine.lan.inv";
};
```

Ainsi on crée notre zone DNS **mondomaine.lan**, en spécifiant que le serveur en est le « maître », et que la liste des éléments à résoudre se trouve dans le fichier `/etc/bind/db.mondomaine.lan`.

De la même manière on crée notre zone reverse DNS (c'est-à-dire capable de faire la translation d'une adresse IP vers un nom) qui couvre les IPs appartenant au réseau **192.168.133.0 /24** et dont la liste des éléments à résoudre se trouve dans le fichier `/etc/bind/db.mondomaine.lan.inv`.

4. On configure notre fichier **db.mondomaine.lan** comme suivant :

```
GNU nano 2.2.6      Fichier : db.mondomaine.lan
;
; Fichier de données BIND pour la zone mondomaine.lan
;
$TTL      604800
@         IN      SOA      ns.mondomaine.lan. root.mondomaine.lan. (
                                4                ; Serial
                                604800           ; Refresh
                                86400            ; Retry
                                2419200         ; Expire
                                604800 )        ; Negative Cache TTL
;
@         IN      NS       ns.mondomaine.lan.
ns        IN      A        192.168.133.138
```

SOA : permet de définir les informations relatives à la zone.

- Nom du serveur DNS primaire "ns.mondomaine.lan."
- Adresse mail du contact technique (root.mondomaine.lan; le @ est remplacé par un point).

Serial : est un entier de 32 bits correspondant au numéro de série à incrémenter à chaque modification du fichier. Il permet au serveur secondaire de recharger les informations qu'ils ont. L'usage général vient à le formater de cette manière YYYYMMDDXX, soit pour la première modification du 01/04/2007 -> 2007040101, pour la seconde 2007040102.

Refresh : définit la période de rafraîchissement des données. Si une erreur survient au cours du dernier rafraîchissement, celle-ci sera répétée au bout du délai **Retry**.

Expire : le serveur sera considéré comme non disponible au bout du délai Expire.

Negative cache TTL : définit la durée de vie d'une réponse NXDOMAIN de notre part.

NS : renseigne le nom des serveurs de noms pour le domaine.

A : associe un nom d'hôte à une adresse ipv4 (32 bits) et **AAAA** : associe un nom d'hôte à une adresse ipv6 (128 bits)

CNAME : identifie le nom canonique d'un alias (un nom pointant sur un autre nom)

PTR : c'est simplement la résolution inverse (le contraire du type A).

IN : détermine l'association à la classe Internet. D'autres classes sont disponibles (CH et HS). Pour de plus amples informations vous pouvez consulter le RFC 1035.

5. On configure notre fichier **db.mondomaine.lan.inv** comme suivant :

```
GNU nano 2.2.6      Fichier : db.mondomaine.lan.inv
:
; Fichier de donnees inversees pour Bind9
;
$TTL      604800
@         IN      SOA      ns.mondomaine.lan. root.mondomaine.lan. (
                        5      ; Serial
                        604800 ; Refresh
                        86400  ; Retry
                        2419200; Expire
                        604800 ) ; Negative Cache TTL
;
@         IN      NS       ns.mondomaine.lan.
138       IN      PTR      ns.mondomaine.lan.

[ Lecture de 13 lignes ]
^G Aide      ^O Écrire    ^R Line fich.^Y Page préc.^K Couper      ^C Pos. cur.
^X Quitter   ^J Justifier ^W Chercher  ^V Page suiv.^U Coller     ^T Orthograp.
```

6. On fait ensuite un restart du service bind9

sudo service bind9 restart | systemctl restart bind9

7. Pour vérifier s'il n'y a pas eu d'erreurs lors du chargement de nos bases, on peut utiliser la commande :

tail -n 10 /var/log/syslog

```
root@debian:/etc/bind# service bind9 restart
root@debian:/etc/bind# tail -n 10 /var/log/
alternatives.log daemon.log      faillog          kern.log
apt/             debug          fontconfig.log  lastlog
auth.log         dmesg          fsck/           messages
btmpe            dpkg.log       hp/             syslog
cups/            exim4/         installer/       wtmp
root@debian:/etc/bind# tail -n 10 /var/log/syslog
Sep 23 05:21:00 debian named[3750]: managed keys zone: loaded serial 2
Sep 23 05:21:00 debian named[3750]: zone 0.in-addr.arpa/IN: loaded serial 1
Sep 23 05:21:00 debian named[3750]: zone 127.in-addr.arpa/IN: loaded serial 1
Sep 23 05:21:00 debian named[3750]: zone mondomaine.lan/IN: loaded serial 4
Sep 23 05:21:00 debian named[3750]: zone 133.168.192.in-addr.arpa/IN: loaded serial 5
Sep 23 05:21:00 debian named[3750]: zone 255.in-addr.arpa/IN: loaded serial 1
Sep 23 05:21:00 debian named[3750]: zone localhost/IN: loaded serial 2
Sep 23 05:21:00 debian named[3750]: all zones loaded
Sep 23 05:21:00 debian named[3750]: running
```

8. On peut aussi utiliser la commande suivante dig qui permet d'interroger directement le serveur DNS de son choix et d'obtenir de nombreuses informations, en plus de la résolution de noms et la résolution inverse :

dig ns.mondomaine.lan :

```
root@debian:/etc/bind# dig ns.mondomaine.lan
; <<>> DiG 9.9.5-9+deb8u6-Debian <<>> ns.mondomaine.lan
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 9147
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
ns.mondomaine.lan.                IN      A

;; ANSWER SECTION:
ns.mondomaine.lan.        604800  IN      A      192.168.133.138

;; AUTHORITY SECTION:
mondomaine.lan.          604800  IN      NS      ns.mondomaine.lan.

;; Query time: 2 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Fri Sep 23 05:23:48 CEST 2016
;; MSG SIZE rcvd: 76
```

dig -x 192.168.133.138 :

```
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 34997
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
138.133.168.192.in-addr.arpa.  IN      PTR

;; ANSWER SECTION:
138.133.168.192.in-addr.arpa. 604800  IN      PTR      ns.mondomaine.lan.

;; AUTHORITY SECTION:
133.168.192.in-addr.arpa. 604800  IN      NS      ns.mondomaine.lan.

;; ADDITIONAL SECTION:
ns.mondomaine.lan.        604800  IN      A      192.168.133.138

;; Query time: 3 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Fri Sep 23 05:27:15 CEST 2016
;; MSG SIZE rcvd: 118
```


9. On vérifie que le ping fonctionne correctement :

```
root@debian:/etc/bind# ping ns
PING ns.mondomaine.lan (192.168.133.138) 56(84) bytes of data.
64 bytes from ns.mondomaine.lan (192.168.133.138): icmp_seq=1 ttl=64 time=0.018
ms
64 bytes from ns.mondomaine.lan (192.168.133.138): icmp_seq=2 ttl=64 time=0.055
ms
64 bytes from ns.mondomaine.lan (192.168.133.138): icmp_seq=3 ttl=64 time=0.058
ms
64 bytes from ns.mondomaine.lan (192.168.133.138): icmp_seq=4 ttl=64 time=0.058
ms
64 bytes from ns.mondomaine.lan (192.168.133.138): icmp_seq=5 ttl=64 time=0.052
ms
64 bytes from ns.mondomaine.lan (192.168.133.138): icmp_seq=6 ttl=64 time=0.057
ms
^C
--- ns.mondomaine.lan ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5016ms
rtt min/avg/max/mdev = 0.018/0.049/0.058/0.016 ms
```

```
root@debian:/etc/bind# ping ns.mondomaine.lan
PING ns.mondomaine.lan (192.168.133.138) 56(84) bytes of data.
64 bytes from ns.mondomaine.lan (192.168.133.138): icmp_seq=1 ttl=64 time=0.021
ms
64 bytes from ns.mondomaine.lan (192.168.133.138): icmp_seq=2 ttl=64 time=0.057
ms
64 bytes from ns.mondomaine.lan (192.168.133.138): icmp_seq=3 ttl=64 time=0.054
ms
64 bytes from ns.mondomaine.lan (192.168.133.138): icmp_seq=4 ttl=64 time=0.052
ms
^C
--- ns.mondomaine.lan ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3021ms
rtt min/avg/max/mdev = 0.021/0.046/0.057/0.014 ms
root@debian:/etc/bind# _
```