

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

DNS DYNAMIQUE DEBIAN

SOMMAIRE :

I)	Objectif.....	2
II)	Prérequis.....	2
III)	Définitions.....	2
IV)	Serveur DNS Maître.....	2-6
	a) Installation du service DNS « bind9 ».....	2
	b) Configuration des fichiers du dossier « /etc ».....	3
	c) Déclaration des zones DNS.....	3-4
	d) Configuration de la zone directe.....	4
	e) Configuration de la zone inversée.....	5
	f) Tests des résolutions DNS.....	5-6
V)	Serveur DNS Esclave.....	7-9
	a) Installation du service DNS « bind9 ».....	7
	b) Configuration des fichiers du dossier « /etc ».....	7
	c) Déclaration des zones DNS.....	8-9
VI)	Tests de résolutions de noms pour les 2 serveurs.....	10-14
	a) Réponse du DNS Maître.....	10-11
	b) Réponse du DNS Esclave.....	11-12
	c) Test avec un client.....	12-14
VII)	DNS dynamique.....	14-17
	a) Configuration du service DHCP.....	14-15
	b) Configuration pour les clients.....	16-17
VIII)	Tests sur les clients.....	17-19
IX)	Conclusion.....	19

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

I) Objectif

Dans cette procédure, nous allons montrer comment installer et configurer un serveur **DNS Maître (principal)** et **Esclave (secondaire)** sous Debian.

II) Prérequis

Pour réaliser cette procédure, nous avons besoin des éléments suivants :

OS	Distribution	Version	C/S	IP du serveur DNS Maître	IP du serveur DNS Esclave
Debian Jessie	Linux	8.5	S	192.168.1.132	192.168.1.133

Nom complet du serveur DNS Maître	Nom complet du serveur DNS Esclave	Nom de domaine
DNSMaitre.ettori.local	DNSEsclave.ettori.local	ettori.local

Nom du serveur DHCP pour le DNS dynamique	Adresse IP du serveur DHCP pour le DNS dynamique
DHCP	192.168.1.135

III) Définitions

- Le service **DNS (Domain Name System)** permet de résoudre un nom de domaine, les adresses IP en noms d'hôtes et les noms d'hôtes en adresses IP. Celui-ci permet aux utilisateurs de naviguer sur Internet.
- Le service **DHCP (Dynamic Host Configuration Protocol)** permet d'attribuer dynamiquement au minimums 3 éléments : une adresse IP, un masque de sous-réseau et un bail **DHCP** (durée de vie l'adresse IP attribuée) selon une plage d'adresses définie.
- Le **DNS dynamique** permet, au moyen d'un serveur **DHCP**, de mettre à jour automatiquement les zones **DNS** créées.

IV) Serveur DNS Maître

a) Installation du service DNS « bind9 »

- Tout d'abord, nous mettons à jour les paquets :

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

- Nous installons le service « **bind9** » :

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

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

b) Configuration des fichiers du dossier « /etc »

- Tout d'abord, nous renommons la machine :

```
GNU nano 2.2.6      Fichier : /etc/hostname
DNSMaitre.ettori.local
```

- Dans le fichier « **/etc/hosts** », nous ajoutons l'adresse IP du serveur, son nom complet et simplifié :

```
GNU nano 2.2.6      Fichier : /etc/hosts
127.0.0.1      localhost
127.0.1.1      DNSMaitre
192.168.1.132  DNSMaitre.ettori.local      DNSMaitre
```

- Dans le fichier « **/etc/resolv.conf** », nous modifions le nom et la recherche du domaine, et l'adresse IP du serveur **DNS** :

```
GNU nano 2.2.6      Fichier : /etc/resolv.conf
domain ettori.local
search ettori.local
nameserver 192.168.1.132
```

- Maintenant, nous ajoutons l'adresse IP du serveur en nom **DNS** dans le fichier « **/etc/network/interfaces** » :

```
allow-hotplug eth0
iface eth0 inet static
address 192.168.1.132
netmask 255.255.255.0
gateway 192.168.1.254
dns-nameservers 192.168.1.132
```

c) Déclaration des zones DNS

- Nous éditons le fichier « **/etc/bind/named.conf.local** » pour renseigner les zones :

⇒ Voici la configuration zone directe :

```
GNU nano 2.2.6      Fichier : /etc/bind/named.conf.local
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not use
// organization
//include "/etc/bind/zones.rfc1918";
zone "ettori.local" IN {
type master;
file "/var/cache/bind/db.ettori.local";
};
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

Remarque : le fichier de zone **directe** se nommera « **db.ettori.local** ».

⇒ Voici la zone inversée :

```
zone "1.168.192.in-addr.arpa" IN {
type master;
file "/var/cache/bind/rev.ettori.local";
};
```

Remarque : le fichier de zone **inversée** se nommera « **rev.ettori.local** ».

- Nous vérifions la configuration si celle-ci est correcte :

```
root@DNSMaitre:~# named-checkconf /etc/bind/named.conf.local
root@DNSMaitre:~# _
```

Ici, nous constatons que c'est le cas.

d) Configuration de la zone directe

- Nous créons le fichier de la zone directe dans le dossier « **/var/cache/bind** » :

```
root@DNSMaitre:/var/cache/bind# touch db.ettori.local
root@DNSMaitre:/var/cache/bind# _
```

- Nous l'éditons et renseignons les enregistrements suivants :

```
GNU nano 2.2.6      Fichier : /var/cache/bind/db.ettori.local
$TTL 86400
@ IN SOA DNSMaitre.ettori.local. root.ettori.local (
2016092201
3600
180
3600
60 )
@ IN NS DNSMaitre.ettori.local.
DNSMaitre.ettori.local. IN A 192.168.1.132
```

- ⇒ « **2016092201** » correspond au numéro de série.
- ⇒ Le premier « **3600** » est la valeur numérique de rafraîchissement.
- ⇒ « **180** » correspond à la tentative de connexion au serveur **DNS Esclave**.
- ⇒ Le second « **3600** » représente l'absence de communication au serveur.
- ⇒ « **60** » représente au temps de réponse négatif.

- Enfin, nous testons la configuration de la zone directe et constatons que celle-ci est correcte grâce au message « **OK** » :

```
root@DNSMaitre:~# named-checkzone ettori.local /var/cache/bind/db.ettori.local
zone ettori.local/IN: loaded serial 2016092201
OK
root@DNSMaitre:~# _
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

e) Configuration de la zone inversée

- Nous créons le fichier de la zone inversée dans le dossier « **/var/cache/bind** » :

```
root@DNSMaitre:/var/cache/bind# touch rev.ettori.local
root@DNSMaitre:/var/cache/bind# _
```

- Nous l'éditons et mettons les enregistrements suivants :

```
GNU nano 2.2.6 Fichier : /var/cache/bind/rev.ettori.local
$TTL 86400
@ IN SOA DNSMaitre.ettori.local. root.ettori.local (
2016092201
3600
180
3600
60 )
@ IN NS DNSMaitre.ettori.local.
108 IN PTR DNSMaitre.ettori.local.
```

- Nous testons la configuration de la zone inversée et constatons que celle-ci est correcte :

```
root@DNSMaitre:~# named-checkzone rev.ettori.local /var/cache/bind/rev.ettori.local
zone rev.ettori.local/IN: loaded serial 2016092201
OK
root@DNSMaitre:~# _
```

- Nous redémarrons le service « **bind9** » pour prendre en compte les modifications :

```
root@DNSMaitre:~# systemctl restart bind9.service
root@DNSMaitre:~# _
```

f) Tests des résolutions DNS

- Nous testons la résolution de noms grâce à la commande « **nslookup** » :

```
root@DNSMaitre:~# nslookup DNSMaitre.ettori.local
Server:          192.168.1.132
Address:         192.168.1.132#53

Name:   DNSMaitre.ettori.local
Address: 192.168.1.132
```

```
root@DNSMaitre:~# nslookup 192.168.1.132
Server:          192.168.1.132
Address:         192.168.1.132#53

132.1.168.192.in-addr.arpa    name = DNSMaitre.ettori.local.
```

Nous constatons que la résolution **DNS** des 2 zones fonctionne.

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

- Nous pouvons également la tester grâce à la commande « **dig** » :

```

root@DNSMaitre:~# dig DNSMaitre.ettori.local

; <<>> DiG 9.9.5-9+deb8u6-Debian <<>> DNSMaitre.ettori.local
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5452
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;DNSMaitre.ettori.local.          IN      A

;; ANSWER SECTION:
DNSMaitre.ettori.local. 86400   IN      A      192.168.1.132

;; AUTHORITY SECTION:
ettori.local.           86400   IN      NS     DNSMaitre.ettori.local.

;; Query time: 26 msec
;; SERVER: 192.168.1.132#53(192.168.1.132)
;; WHEN: Thu Sep 22 11:15:04 CEST 2016
;; MSG SIZE rcvd: 81

```

```

root@DNSMaitre:~# dig 192.168.1.132

; <<>> DiG 9.9.5-9+deb8u6-Debian <<>> 192.168.1.132
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 12778
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

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

;; AUTHORITY SECTION:
.                               10800   IN      SOA     a.root-servers.net. nstld.verisign-grs.com. 2016092200 1800 900 604800 86400

;; Query time: 417 msec
;; SERVER: 192.168.1.132#53(192.168.1.132)
;; WHEN: Thu Sep 22 11:16:08 CEST 2016
;; MSG SIZE rcvd: 117

```

- Enfin, nous pouvons redémarrer les fichiers de zone sans redémarrer le service DNS « **bind9** » pour assurer une continuité de services :

```

root@DNSMaitre:~# rndc reload
server reload successful
root@DNSMaitre:~# _

```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

V) Serveur DNS Esclave

a) Installation du service DNS « bind9 »

- Nous mettons à jour les paquets :

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

- Nous installons le service « **bind9** » :

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

b) Configuration des fichiers du dossier « /etc »

- Nous renommons la machine :

```
GNU nano 2.2.6 Fichier : /etc/hostname
DNSEslave.ettori.local
```

- Dans le fichier « **/etc/hosts** », nous ajoutons l'adresse IP du serveur, son nom complet et son nom comme sur le serveur **Maître** :

```
GNU nano 2.2.6 Fichier : /etc/hosts
127.0.0.1 localhost
127.0.1.1 DNSEslave
192.168.1.133 DNSEslave.ettori.local DNSEslave
```

- Dans le fichier « **/etc/resolv.conf** », nous modifions le nom de domaine, la recherche de celui-ci et l'adresse IP des serveurs **DNS Maître** et **Esclave** :

```
GNU nano 2.2.6 Fichier : /etc/resolv.conf
domain ettori.local
search ettori.local
nameserver 192.168.1.132
nameserver 192.168.1.133
```

- Maintenant, nous ajoutons l'adresse IP du serveur en nom **DNS** dans le fichier « **/etc/network/interfaces** » :

```
allow-hotplug eth0
iface eth0 inet static
address 192.168.1.133
netmask 255.255.255.0
gateway 192.168.1.254
dns-nameservers 192.168.1.132
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

c) Déclaration des zones DNS

- Nous éditons le fichier « **/etc/bind/named.conf.local** » pour saisir les zones :

⇒ Voici la configuration de la zone directe :

```
GNU nano 2.2.6      Fichier : /etc/bind/named.conf.local
//
// Do any local configuration here
//
// Consider adding the 1918 zones here, if they are not use
// organization
//include "/etc/bind/zones.rfc1918";

zone "ettori.local" IN {
type slave;
masters {192.168.1.132;};
file "/var/cache/bind/db.ettori.local";
};
```

⇒ Voici la configuration de la zone inversée :

```
zone "1.168.192.in-addr.arpa" IN {
type slave;
masters {192.168.1.132;};
file "/var/cache/bind/rev.ettori.local";
};
```

Remarque : Les fichiers de zone portent le même nom que ceux du serveur **Maître**.

- Nous vérifions la configuration et constatons que celle-ci est correcte :

```
root@DNSEslave:~# named-checkconf /etc/bind/named.conf.local
root@DNSEslave:~# _
```

- Nous retournons sur le serveur **Maître** et ajoutons les 4 lignes en jaune pour les 2 zones qui permettent de transférer les zones du serveur **Esclave** vers le serveur **Maître** :

```
zone "ettori.local" IN {
type master;
file "/var/cache/bind/db.ettori.local";
notify yes;
allow-transfer {192.168.1.133;};
};
```

```
zone "1.168.192.in-addr.arpa" IN {
type master;
file "/var/cache/bind/rev.ettori.local";
notify yes;
allow-transfer {192.168.1.133;};
};
```


ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

- Nous devons modifier les fichiers de zone :

⇒ Contenu du fichier de zone directe :

```
GNU nano 2.2.6 Fichier : /var/cache/bind/db.ettori.local
$TTL 86400
@ IN SOA DNSMaitre.ettori.local. root.ettori.local (
2016092204
3600
180
3600
60 )
@ IN NS DNSMaitre.ettori.local.
@ IN NS DNSEslave.ettori.local.
DNSMaitre.ettori.local. IN A 192.168.1.132
DNSEslave.ettori.local. IN A 192.168.1.133
```

⇒ Contenu du fichier de zone inversée :

```
GNU nano 2.2.6 Fichier : /var/cache/bind/rev.ettori.local
$TTL 86400
@ IN SOA DNSMaitre.ettori.local. root.ettori.local (
2016092204
3600
180
3600
60 )
@ IN NS DNSMaitre.ettori.local.
@ IN NS DNSEslave.ettori.local.
132 IN PTR DNSMaitre.ettori.local.
133 IN PTR DNSEslave.ettori.local.
```

- Nous redémarrons le service « **bind9** » sur les 2 serveurs pour prendre en compte les modifications :

« **systemctl restart bind9.service** ».

- Nous visualisons les logs à la fin du fichier « **/var/log/syslog** » pour voir si tout s'est bien passé pour la résolution de noms et constatons que cela a fonctionné car comme nous voyons que les 2 zones ont bien été transférées :

```
Sep 29 10:05:29 DNSMaitre named[1345]: zone ettori.local/IN: loaded serial 2016092204
Sep 29 10:05:29 DNSMaitre named[1345]: zone 255.in-addr.arpa/IN: loaded serial 1
Sep 29 10:05:29 DNSMaitre named[1345]: zone localhost/IN: loaded serial 2
Sep 29 10:05:29 DNSMaitre named[1345]: zone 1.168.192.in-addr.arpa/IN: loaded serial 2016092204
Sep 29 10:05:29 DNSMaitre named[1345]: all zones loaded
Sep 29 10:05:29 DNSMaitre named[1345]: running
Sep 29 10:05:29 DNSMaitre named[1345]: zone 1.168.192.in-addr.arpa/IN: sending notifies (serial 2016092204)
Sep 29 10:05:29 DNSMaitre named[1345]: zone ettori.local/IN: sending notifies (serial 2016092204)
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

VI) Tests de résolutions de noms pour les 2 serveurs

Nous testons les résolutions des adresses IP et les noms des 2 serveurs **DNS** sur les 2 serveurs.

a) Réponse du DNS Maître

- Sur le serveur DNS Maître :

```
root@DNSMaitre:~# nslookup 192.168.1.132
Server:      192.168.1.132
Address:     192.168.1.132#53

132.1.168.192.in-addr.arpa      name = DNSMaitre.ettori.local.
```

```
root@DNSMaitre:~# nslookup DNSMaitre.ettori.local
Server:      192.168.1.132
Address:     192.168.1.132#53

Name:       DNSMaitre.ettori.local
Address:    192.168.1.132
```

```
root@DNSMaitre:~# nslookup 192.168.1.133
Server:      192.168.1.132
Address:     192.168.1.132#53

133.1.168.192.in-addr.arpa      name = DNSEsclave.ettori.local.
```

```
root@DNSMaitre:~# nslookup DNSEsclave.ettori.local
Server:      192.168.1.132
Address:     192.168.1.132#53

Name:       DNSEsclave.ettori.local
Address:    192.168.1.133
```

- Sur le serveur DNS Esclave :

```
root@DNSEsclave:~# nslookup 192.168.1.132
Server:      192.168.1.132
Address:     192.168.1.132#53

132.1.168.192.in-addr.arpa      name = DNSMaitre.ettori.local.
```

```
root@DNSEsclave:~# nslookup DNSMaitre.ettori.local
Server:      192.168.1.132
Address:     192.168.1.132#53

Name:       DNSMaitre.ettori.local
Address:    192.168.1.132
```

```
root@DNSEsclave:~# nslookup 192.168.1.133
Server:      192.168.1.132
Address:     192.168.1.132#53

133.1.168.192.in-addr.arpa      name = DNSEsclave.ettori.local.
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

```
root@DNSEslave:~# nslookup DNSEslave.ettori.local
Server:          192.168.1.132
Address:         192.168.1.132#53

Name:   DNSEslave.ettori.local
Address: 192.168.1.133
```

Donc, nous constatons que c'est le serveur **DNS Maître** est fonctionnelle qui répond par défaut, soit l'adresse IP : **192.168.1.132**.

b) Réponse du DNS Esclave

- Pour effectuer les tests de réponse du serveur **DNS Esclave**, nous éteignons d'abord le service « **bind9** » sur le serveur **DNS Maître** ou mettre en commentaire dans le fichier « **/etc/resolv.conf** » le nom du serveur **DNS Maître** :

« **systemctl stop bind9.service** ».

OU :

« **#nameserver 192.168.1.132** ».

- Sur le serveur **DNS Maître** :

```
root@DNSMaitre:~# nslookup 192.168.1.132
Server:          192.168.1.133
Address:         192.168.1.133#53

132.1.168.192.in-addr.arpa      name = DNSMaitre.ettori.local.
```

```
root@DNSMaitre:~# nslookup DNSMaitre.ettori.local
Server:          192.168.1.133
Address:         192.168.1.133#53

Name:   DNSMaitre.ettori.local
Address: 192.168.1.132
```

```
root@DNSMaitre:~# nslookup 192.168.1.133
Server:          192.168.1.133
Address:         192.168.1.133#53

133.1.168.192.in-addr.arpa      name = DNSEslave.ettori.local.
```

```
root@DNSMaitre:~# nslookup DNSEslave.ettori.local
Server:          192.168.1.133
Address:         192.168.1.133#53

Name:   DNSEslave.ettori.local
Address: 192.168.1.133
```

- Sur le serveur **DNS Esclave** :

```
root@DNSEslave:~# nslookup 192.168.1.132
Server:          192.168.1.133
Address:         192.168.1.133#53

132.1.168.192.in-addr.arpa      name = DNSMaitre.ettori.local.
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

```
root@DNSEslave:~# nslookup DNSMaitre.ettori.local
Server:          192.168.1.133
Address:         192.168.1.133#53

Name:   DNSMaitre.ettori.local
Address: 192.168.1.132
```

```
root@DNSEslave:~# nslookup 192.168.1.133
Server:          192.168.1.133
Address:         192.168.1.133#53

133.1.168.192.in-addr.arpa      name = DNSEslave.ettori.local.
```

```
root@DNSEslave:~# nslookup DNSEslave.ettori.local
Server:          192.168.1.133
Address:         192.168.1.133#53

Name:   DNSEslave.ettori.local
Address: 192.168.1.133
```

Donc, nous constatons que le serveur **DNS Esclave** répond en cas de défaillance du serveur **Maître**, soit l'adresse IP : **192.168.1.133**.

c) Test avec un client

- Nous ajoutons un client dans les 2 fichiers de zones sur le serveur **DNS Maître** en modifiant également le numéro de série pour prendre en compte la résolution de noms :

```
GNU nano 2.2.6      Fichier : /var/cache/bind/db.ettori.local

$TTL 86400
@ IN SOA DNSMaitre.ettori.local. root.ettori.local (
2016092205
3600
180
3600
60 )
@ IN NS DNSMaitre.ettori.local.
@ IN NS DNSEslave.ettori.local.
DNSMaitre.ettori.local. IN A 192.168.1.132
DNSEslave.ettori.local. IN A 192.168.1.133
POSTE24.ettori.local. IN A 192.168.1.74
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

```
GNU nano 2.2.6 Fichier : /var/cache/bind/rev.ettori.local
$TTL 86400
@ IN SOA DNSMaitre.ettori.local. root.ettori.local (
2016092205
3600
180
3600
60 )
@ IN NS DNSMaitre.ettori.local.
@ IN NS DNSEsclave.ettori.local.
132 IN PTR DNSMaitre.ettori.local.
133 IN PTR DNSEsclave.ettori.local.
74 IN PTR POSTE24.ettori.local.
```

- Nous redémarrons le service « **bind9** » sur les 2 serveurs DNS :

« **systemctl restart bind9.service** ».
- Maintenant, nous allons tester la résolution de noms du client intégré :
 ⇒ Test de la réponse du serveur DNS Maître sur le DNS Maître :

```
root@DNSMaitre:~# nslookup 192.168.1.74
Server:      192.168.1.132
Address:     192.168.1.132#53

74.1.168.192.in-addr.arpa      name = POSTE24.ettori.local.

root@DNSMaitre:~# nslookup POSTE24.ettori.local
Server:      192.168.1.132
Address:     192.168.1.132#53

Name:       POSTE24.ettori.local
Address:    192.168.1.74
```

- ⇒ Test de la réponse du serveur DNS Maître sur le DNS Esclave :

```
root@DNSEsclave:~# nslookup 192.168.1.74
Server:      192.168.1.132
Address:     192.168.1.132#53

74.1.168.192.in-addr.arpa      name = POSTE24.ettori.local.

root@DNSEsclave:~# nslookup POSTE24.ettori.local
Server:      192.168.1.132
Address:     192.168.1.132#53

Name:       POSTE24.ettori.local
Address:    192.168.1.74
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

- Maintenant, nous éteignons le service « **bind9** » sur les 2 serveurs **DNS** et testons la réponse du **DNS Esclave** :

Test de la réponse du serveur **DNS Esclave** sur le **DNS Maître** :

```
root@DNSMaître:~# nslookup 192.168.1.74
Server:      192.168.1.133
Address:     192.168.1.133#53

74.1.168.192.in-addr.arpa      name = POSTE24.ettori.local.
```

```
root@DNSMaître:~# nslookup POSTE24.ettori.local
Server:      192.168.1.133
Address:     192.168.1.133#53

Name:       POSTE24.ettori.local
Address:    192.168.1.74
```

Test de la réponse du serveur **DNS Esclave** sur le **DNS Esclave** :

```
root@DNSEsclave:~# nslookup 192.168.1.74
Server:      192.168.1.133
Address:     192.168.1.133#53

74.1.168.192.in-addr.arpa      name = POSTE24.ettori.local.
```

```
root@DNSEsclave:~# nslookup POSTE24.ettori.local
Server:      192.168.1.133
Address:     192.168.1.133#53

Name:       POSTE24.ettori.local
Address:    192.168.1.74
```

Donc, nous constatons que les 2 serveurs **DNS** sont fonctionnels car les 2 zones résolvent bien leurs noms et leurs adresses IP.

NB : Le serveur **DNS Esclave** étant mis en place et fonctionnel, il permettra, en cas de panne du serveur **DNS Maître**, de prendre le relais et d'assurer une continuité de services.

VII) DNS dynamique

a) Configuration du service DHCP

- Tout d'abord, nous mettons à jour les paquets sur le serveur **DHCP** :

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

- Nous installons sur ce nouveau serveur le service « **isc-dhcp-server** » :

```
root@DHCP:~# apt-get install isc-dhcp-server
```

- Ici, nous ne possédons que 4 adresses IP (3 en IP fixe et 1 en **DHCP**), Donc, nous n'avons qu'une adresse IP à distribuer. Avant de configurer le serveur **DHCP**, si nous voulons attribuer 2 adresses IP pour une étendue **DHCP**, nous éteignons le serveur **DNS Esclave** et mettons son adresse IP dans la plage d'adresses IP en plus de celle qui n'est pas utilisée pour tester la récupération des configurations TCP/IP différentes des 2 clients :

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

```
root@DNSEslave:~# poweroff
```

Remarque : Dans un réseau normal, nous ne procédons pas de cette manière.

- Pour ce faire, nous le configurons en nous rendant dans le fichier « `/etc/dhcp/dhcpd.conf` » afin qu'il puisse démarrer et distribuer une configuration TCP/IP :

⇒ Nous mettons le nom de domaine et les adresse IP des 2 serveurs **DNS** :

```
# option definitions common to all supported networks...
option domain-name "ettori.local";
option domain-name-servers 192.168.1.132, 192.168.1.133;

default-lease-time 600;
max-lease-time 7200;
```

⇒ Nous définissons le réseau sur lequel nous voulons distribuer les adresses IP aux clients grâce à une étendue **DHCP**, prenant en compte l'adresse IP du serveur **DNS Esclave** pour tester, son masque et la passerelle par défaut :

```
# This is a very basic subnet declaration.

subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.133 192.168.1.134;
    option routers 192.168.1.254;
}
```

- Nous essayons de démarrer le service « **isc-dhcp-server** » :

```
root@DHCP:~# systemctl start isc-dhcp-server.service
root@DHCP:~# _
```

- Nous vérifions que celui-ci est bien démarré :

```
root@DHCP:~# systemctl status isc-dhcp-server.service
• isc-dhcp-server.service - LSB: DHCP server
  Loaded: loaded (/etc/init.d/isc-dhcp-server)
  Active: active (running) since jeu. 2016-10-13 09:40:30 CEST;
  Process: 14751 ExecStart=/etc/init.d/isc-dhcp-server start (co
s=0/SUCCESS)
  CGroup: /system.slice/isc-dhcp-server.service
          └─14759 /usr/sbin/dhcpd -q -cf /etc/dhcp/dhcpd.conf -
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

b) Configuration pour les clients

Pour un client DHCP Windows :

- Nous retournons sur le serveur **DNS Maître**, nous nous rendons dans le fichier « **/etc/bind/named.conf.local** » et ajoutons cette nouvelle ligne (en jaune) avec l'adresse IP du serveur **DHCP** pour permettre la mise à jour des zones **DNS** :

```
GNU nano 2.2.6      Fichier : /etc/bind/named.conf.local
notify yes;
allow-transfer {192.168.1.133;};
allow-update {192.168.1.135;};
};

zone "1.168.192.in-addr.arpa" IN {
type master;
file "/var/cache/bind/rev.ettori.local";
notify yes;
allow-transfer {192.168.1.133;};
allow-update {192.168.1.135;};
};
```

- Nous nous rendons sur le serveur **DHCP** et dans le fichier « **/etc/dhcp/dhcpd.conf** » et ajoutons au début de ce fichier, ces lignes qui permettent d'activer la mise à jour des clients **DHCP** avec le **DNS** :

```
# have support for DDNS.)
ddns-update-style interim;
ddns-updates on;
ignore client-updates;
update-static-leases on;
allow-unknown-clients;

# option definitions common to all supported networks...
option domain-name "ettori.local";
option domain-name-servers 192.168.1.132, 192.168.1.133;
```

- Nous ajoutons ces nouvelles lignes à la fin de ce fichier où dans l'attribut « **primary** » pour les 2 zones (directe et inversée), nous mettons l'adresse IP du serveur **DNS Maître** :

```
zone ettori.local. {primary 192.168.1.132;}
zone 1.168.192.in-addr.arpa. {primary 192.168.1.132;}
```


ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

Pour un client DHCP Linux :

- Nous nous rendons dans le fichier « `/etc/dhcp/dhclient.conf` » et remplaçons cette ligne par le nom d'un **client Linux** :

```
GNU nano 2.2.6 Fichier : /etc/dhcp/dhclient.conf
#
option rfc3442-classless-static-routes code 121 = array of unsigned integer
#send host-name "andare.fugue.com";
send host-name = "clientDHCP.ettori.local";
```

- Enfin, nous redémarrons les services « **bind9** » et « **isc-dhcp-server** » pour prendre en compte les modifications :

```
root@DNSMaitre:~# systemctl restart bind9.service
root@DNSMaitre:~# _

root@DHCP:~# systemctl restart isc-dhcp-server.service
root@DHCP:~# _
```

VIII) Tests sur les clients

- Sur le client **Windows**, nous pouvons voir la nouvelle configuration TCP/IP :

```
Carte Ethernet Connexion au réseau local :
    Suffixe DNS propre à la connexion. . . : ettori.local
    Adresse IPv4. . . . . : 192.168.1.133
    Masque de sous-réseau. . . . . : 255.255.255.0
    Passerelle par défaut. . . . . : 192.168.1.254

Carte Tunnel isatap.ettori.local :
    Statut du média. . . . . : Média déconnecté
    Suffixe DNS propre à la connexion. . . : ettori.local
```

- Sur le client **Linux**, nous vérifions la même chose :

```
root@clientDHCP:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:e4:40:6f
          inet adr:192.168.1.134  Bcast:192.168.1.255
          adr inet6: fe80::a00:27ff:fee4:406f/64 Scope:
          adr inet6: fd23:6507:b29b:1:a00:27ff:fee4:406
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Met
          RX packets:5380 errors:0 dropped:0 overruns:0
          TX packets:298 errors:0 dropped:0 overruns:0
          collisions:0 lg file transmission:1000
          RX bytes:441579 (431.2 KiB)  TX bytes:45468 (
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

- Pour confirmer, nous nous rendons dans « `/var/lib/dhcp/dhcpd.leases` » pour visualiser les baux **DHCP** attribués sur les 2 clients :

⇒ Client Windows :

```
lease 192.168.1.133 {
  starts 4 2016/10/13 09:32:35;
  ends 4 2016/10/13 09:42:35;
  cltt 4 2016/10/13 09:32:35;
  binding state active;
  next binding state free;
  rewind binding state free;
  hardware ethernet 08:00:27:7d:e9:d9;
  uid "\001\010\000'\351\331";
  set ddns-txt = "31abd7ab87e02fa9cebfcae50cffcca355d";
  set ddns-fwd-name = "DHCPClient.ettori.local";
  client-hostname "DHCPClient";
```

⇒ Client Linux :

```
lease 192.168.1.134 {
  starts 4 2016/10/13 09:46:15;
  ends 4 2016/10/13 09:56:15;
  cltt 4 2016/10/13 09:46:15;
  binding state active;
  next binding state free;
  rewind binding state free;
  hardware ethernet 08:00:27:e4:40:6f;
  set ddns-txt = "005c0c1684b2cbb0c7ef2725804517a729";
  set ddns-fwd-name = "clientDHCP.ettori.local.ettori.local";
  client-hostname "clientDHCP.ettori.local";
```

Nous constatons que les clients **DHCP** reçoivent bien leurs configurations TCP/IP automatiquement.

- Nous pouvons consulter les logs pour vérifier :

```
GNU nano 2.2.6      Fichier : /var/log/syslog
Oct 13 11:32:07 DNSMaitre dhcpd: DHCPPOFFER on 192.168.1.134 to 08:00:27:5d:15:
Oct 13 11:32:07 DNSMaitre dhcpd: DHCPREQUEST for 192.168.1.109 (192.168.1.111)
Oct 13 11:32:16 DNSMaitre dhcpd: DHCPDISCOVER from 08:00:27:3d:fb:65 via eth0:
Oct 13 11:32:16 DNSMaitre dhcpd: DHCPREQUEST for 192.168.1.3 (192.168.1.254) f
Oct 13 11:32:20 DNSMaitre dhcpd: DHCPDISCOVER from 08:00:27:5d:15:aa (clientli
Oct 13 11:32:20 DNSMaitre dhcpd: DHCPPOFFER on 192.168.1.134 to 08:00:27:5d:15:
Oct 13 11:32:20 DNSMaitre dhcpd: DHCPREQUEST for 192.168.1.109 (192.168.1.111)
Oct 13 11:32:30 DNSMaitre dhcpd: DHCPRELEASE of 192.168.1.133 from 08:00:27:7d:
Oct 13 11:32:30 DNSMaitre dhcpd: Removed forward map from DHCPClient.ettori.lc
Oct 13 11:32:30 DNSMaitre dhcpd: Removed reverse map on 133.1.168.192.in-addr.
Oct 13 11:32:33 DNSMaitre dhcpd: DHCPDISCOVER from 08:00:27:5d:15:aa (clientli
Oct 13 11:32:33 DNSMaitre dhcpd: DHCPPOFFER on 192.168.1.134 to 08:00:27:5d:15:
Oct 13 11:32:33 DNSMaitre dhcpd: DHCPREQUEST for 192.168.1.109 (192.168.1.111)
Oct 13 11:32:34 DNSMaitre dhcpd: DHCPDISCOVER from 08:00:27:7d:e9:d9 via eth0:
Oct 13 11:32:35 DNSMaitre dhcpd: DHCPPOFFER on 192.168.1.133 to 08:00:27:7d:e9:
Oct 13 11:32:35 DNSMaitre dhcpd: DHCPREQUEST for 192.168.1.133 (192.168.1.135)
Oct 13 11:32:35 DNSMaitre dhcpd: DHCPACK on 192.168.1.133 to 08:00:27:7d:e9:d9
Oct 13 11:32:35 DNSMaitre dhcpd: Added new forward map from DHCPClient.ettori.
Oct 13 11:32:35 DNSMaitre dhcpd: Added reverse map from 133.1.168.192.in-addr.
Oct 13 11:32:37 DNSMaitre dhcpd: DHCPDISCOVER from 08:00:27:e9:85:b1 via eth0:
```

ETTORI Bastien	BTS SIO 2 ^{ème} année
29 Septembre 2016	Année scolaire : 2016/2017
Option : SISR	Version 1

- Nous pouvons également consulter les logs via l'utilitaire « Putty » :

```
GNU nano 2.2.6      Fichier : /var/log/syslog
Oct 13 11:45:41 DNSMaitre dhcpd: DHCPDISCOVER from 08:00:27:e4:40:6f via eth0
Oct 13 11:45:41 DNSMaitre dhcpd: DHCPREQUEST for 192.168.1.103 (192.168.1.102) from 08:00:27:e4:40:6f via eth0: un
Oct 13 11:45:42 DNSMaitre dhcpd: DHCPOFFER on 192.168.1.134 to 08:00:27:e4:40:6f (clientDHCP.ettori.local) via eth
Oct 13 11:46:03 DNSMaitre dhcpd: DHCPDISCOVER from 08:00:27:43:6d:de via eth0: network 192.168.1.0/24; no free lea
Oct 13 11:46:03 DNSMaitre dhcpd: DHCPREQUEST for 192.168.1.25 (192.168.1.254) from 08:00:27:43:6d:de via eth0: unk
Oct 13 11:46:15 DNSMaitre dhcpd: DHCPDISCOVER from 08:00:27:e4:40:6f (clientDHCP.ettori.local) via eth0
Oct 13 11:46:15 DNSMaitre dhcpd: DHCPOFFER on 192.168.1.134 to 08:00:27:e4:40:6f (clientDHCP.ettori.local) via eth
Oct 13 11:46:15 DNSMaitre dhcpd: DHCPREQUEST for 192.168.1.134 (192.168.1.135) from 08:00:27:e4:40:6f (clientDHCP.
Oct 13 11:46:15 DNSMaitre dhcpd: DHCPACK on 192.168.1.134 to 08:00:27:e4:40:6f (clientDHCP.ettori.local) via eth0
Oct 13 11:46:15 DNSMaitre dhcpd: Added new forward map from clientDHCP.ettori.local.ettori.local to 192.168.1.134
Oct 13 11:46:15 DNSMaitre dhcpd: Added reverse map from 134.1.168.192.in-addr.arpa. to clientDHCP.ettori.local.ett
```

Ici, nous voyons l'ajout des 2 zones **DNS** aux 2 clients **DHCP**.

- Enfin, nous pouvons constater que les fichiers de zones **DNS** répliqués avec l'extension « .jnl » ont été créés automatiquement sur le serveur **DNS Maître** :

```
root@DNSMaitre:~# ls -l /var/cache/bind/
total 40
-rw-r--r-- 1 bind bind 551 oct. 13 11:40 db.ettori.local
-rw-r--r-- 1 bind bind 14400 oct. 13 11:46 db.ettori.local.jnl
-rw-r--r-- 1 bind bind 720 oct. 13 09:23 managed-keys.bind
-rw-r--r-- 1 bind bind 528 oct. 13 11:41 rev.ettori.local
-rw-r--r-- 1 bind bind 10738 oct. 13 11:46 rev.ettori.local.jnl
root@DNSMaitre:~#
```

IX) Conclusion

En conclusion, nous pouvons dire que les 2 serveurs **DNS** et le serveur **DHCP** pour le **DNS dynamique** sont fonctionnels car les clients **Linux** et **Windows** reçoivent bien leurs configurations TCP/IP dynamiquement et que les fichiers de zones sont mis à jour automatiquement.