

Rapport de Projet : Configuration d'un Serveur DHCP et Web DNS avec Bind9 et Apache2

ANDOME B'AKOE MANEMBE Yannick Stéphane Rodly

21 janvier 2025

Table des matières

1	Introduction	2
2	Prérequis et environnement	2
2.1	Environnement utilisé	2
2.2	Objectifs du projet	2
3	Configuration du serveur DHCP	2
3.1	Installation et configuration de <code>isc-dhcp-server</code>	2
3.2	Test du serveur DHCP	3
4	Configuration du serveur DNS avec Bind9	3
4.1	Installation et configuration de Bind9	3
4.2	Test du serveur DNS	4
5	Configuration du serveur Web avec Apache2	5
5.1	Installation et configuration	5
5.2	Test du serveur web	6
6	Résultats et observations	6
7	Conclusion	6

1 Introduction

Dans ce projet, nous avons configuré un serveur DHCP et un serveur DNS avec Bind9, ainsi qu'un serveur web utilisant Apache2. Ce rapport décrit en détail les étapes de configuration, les tests réalisés, ainsi que les résultats obtenus.

2 Prérequis et environnement

2.1 Environnement utilisé

- Système d'exploitation : Linux (par exemple Debian ou Ubuntu).
- Logiciels : Bind9, Apache2, outils de virtualisation (VMware ou VirtualBox).
- Machines virtuelles : une pour le serveur et une pour le client.

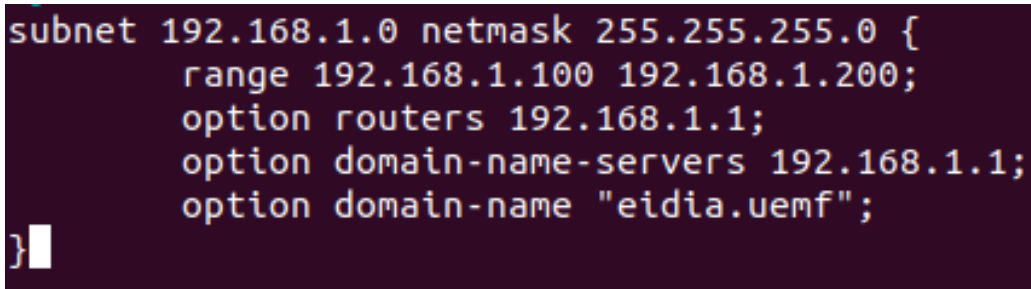
2.2 Objectifs du projet

- Configurer un serveur DHCP pour attribuer des adresses IP dynamiques.
- Configurer un serveur DNS avec Bind9 pour résoudre des noms de domaine.
- Configurer un serveur web avec Apache2 pour héberger une page test.
- Tester la communication avec une machine cliente.

3 Configuration du serveur DHCP

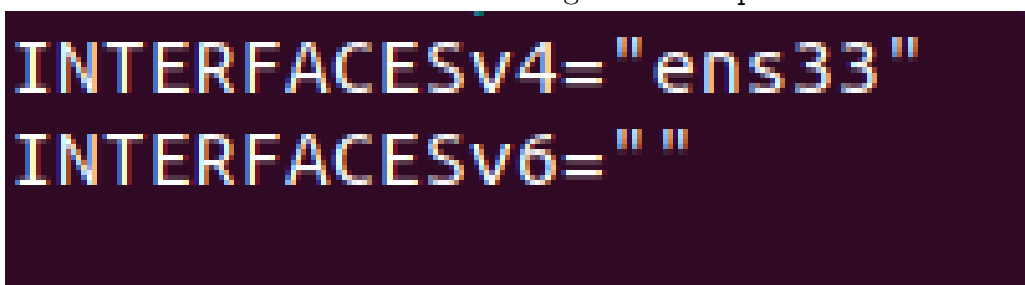
3.1 Installation et configuration de isc-dhcp-server

On commence par installer isc-dhcp-server sur la machine qui sera le serveur, puis on l'ouvre le fichier dhcpd.conf(`sudo nano /etc/dhcp/dhcpd.conf`) on y fait la configuration du serveur, subnet utilisé, pool des adresses, nom de domaine et autres options. Puis on définit l'interface d'écoute du serveur, dans notre cas ens33.



```
subnet 192.168.1.0 netmask 255.255.255.0 {  
    range 192.168.1.100 192.168.1.200;  
    option routers 192.168.1.1;  
    option domain-name-servers 192.168.1.1;  
    option domain-name "eidia.uemf";  
}
```

FIGURE 1 – Fichier de configuration dhcpd.conf.



```
INTERFACESv4="ens33"  
INTERFACESv6=""
```

FIGURE 2 – Configuration de l'interface d'écoute /etc/default/isc-dhcp-server.

3.2 Test du serveur DHCP

Pour tester que le serveur fonctionne correctement après l'avoir redémarrer, l'on a utilisé la commande "sudo systemctl status isc-dhcp-server", affichant que le serveur est "active(running)" donc qu'il fonctionne correctement. Puis depuis la machine client on ping le serveur pour voir si la communication s'effectue ce qui est bel et bien le cas. Sans oublier de passer les connexions des machines en custom.

```
root@ubuntu:/home/bamsyr# sudo systemctl restart isc-dhcp-server
root@ubuntu:/home/bamsyr# sudo systemctl enable isc-dhcp-server
Synchronizing state of isc-dhcp-server.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable isc-dhcp-server
root@ubuntu:/home/bamsyr# sudo systemctl status isc-dhcp-server
● isc-dhcp-server.service - ISC DHCP IPv4 server
   Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2025-01-17 12:25:32 PST; 32s ago
     Docs: man:dhcpd(8)
    Main PID: 51235 (dhcpd)
      Tasks: 4 (limit: 3345)
     Memory: 4.4M
    CGroup: /system.slice/isc-dhcp-server.service
            └─51235 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -cf /etc/dhcp/dhcpd.conf ens33

Jan 17 12:25:32 ubuntu sh[51235]: Wrote 0 leases to leases file.
Jan 17 12:25:32 ubuntu dhcpd[51235]: PID file: /run/dhcp-server/dhcpd.pid
Jan 17 12:25:32 ubuntu dhcpd[51235]: Wrote 0 leases to leases file.
Jan 17 12:25:32 ubuntu dhcpd[51235]: Listening on LPF/ens33/00:0c:29:34:a6:cc/192.168.1.0/24
Jan 17 12:25:32 ubuntu sh[51235]: Listening on LPF/ens33/00:0c:29:34:a6:cc/192.168.1.0/24
Jan 17 12:25:32 ubuntu sh[51235]: Sending on LPF/ens33/00:0c:29:34:a6:cc/192.168.1.0/24
Jan 17 12:25:32 ubuntu sh[51235]: Sending on Socket/fallback/fallback-net
Jan 17 12:25:32 ubuntu dhcpd[51235]: Sending on LPF/ens33/00:0c:29:34:a6:cc/192.168.1.0/24
Jan 17 12:25:32 ubuntu dhcpd[51235]: Sending on Socket/fallback/fallback-net
Jan 17 12:25:32 ubuntu dhcpd[51235]: Server starting service.
root@ubuntu:/home/bamsyr#
```

FIGURE 3 – Capture d'écran du statut du serveur DHCP.

```
bamsyr@ubuntu:~$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=2.43 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=1.67 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=1.68 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=1.70 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=1.61 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=64 time=1.34 ms
^C
--- 192.168.1.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5011ms
rtt min/avg/max/mdev = 1.336/1.736/2.426/0.331 ms
bamsyr@ubuntu:~$
```

FIGURE 4 – Communication réussie entre la machine cliente et le serveur(le client ping le serveur).

4 Configuration du serveur DNS avec Bind9

4.1 Installation et configuration de Bind9

On installe bind9 et les composants nécessaires (bind9utils, bind9-doc, dnsutils). On modifie le fichier de configuration principale named.conf.options, on spécifie DNS server options.

forwarders : serveur DNS externe (e.g., Google DNS) pour se charger des requêtes de domaine non régulé par notre serveur. listen-on : Restreint le serveur DNS à des interfaces ou IPs spécifiques (e.g., 192.168.1.1).

Par la suite on définit les zones du DNS dans le fichier named.conf.local (/etc/bind/named.conf.local), pointant les fichiers de zone. On crée les fichiers de zone, db.eidia.uemf et db.192 dans lesquels on spécifie : SOA (Start of Authority) ; NS (Name Server) ; A (Address) / Et on redémarre le service DNS pour appliquer la nouvelle configuration.

```
GNU nano 4.8
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if th
// organization
//include "/etc/bind/zones.rfc1918";

//Définition de la zone principale (directe)
zone "eidia.uemf" {
    type master;
    file "/etc/bind/db.eidia.uemf";
};

//Définition de la zone inverse
zone "1.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.192;
};
```

FIGURE 5 – Fichier `named.conf.local` configuré.

4.2 Test du serveur DNS

Description des tests réalisés et des outils utilisés (`dig`, `nslookup`). Pour s'assurer du bon fonctionnement du DNS on exécute la commande `dig` sur le serveur, qui retourne "status : NOERROR" indiquant que tout fonctionne et la commande `nslookup` sur le client retournant le nom du serveur et son adresse IP.

```

root@ubuntu:/home/bamsyr# dig @192.168.1.1 www.eidia.uemf

; <<>> DiG 9.18.30-0ubuntu0.20.04.1-Ubuntu <<>> @192.168.1.1 www.eidia.uemf
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 32782
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 80eab07cf82b4ada01000000678ac8ef60ed48dafdac16eb (good)
;; QUESTION SECTION:
;www.eidia.uemf.                                IN      A

;; ANSWER SECTION:
www.eidia.uemf.      86400    IN      A      192.168.1.10

;; Query time: 0 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Fri Jan 17 13:17:35 PST 2025
;; MSG SIZE rcvd: 87

root@ubuntu:/home/bamsyr#

```

FIGURE 6 – Résultat d’une commande dig.

```

bamsyr@ubuntu:~$ sudo nano /etc/resolv.conf
[sudo] password for bamsyr:
bamsyr@ubuntu:~$ nslookup www.eidia.uemf
Server:          192.168.1.1
Address:         192.168.1.1#53

Name:   www.eidia.uemf
Address: 192.168.1.10

bamsyr@ubuntu:~$ nslookup mail.eidia.uemf
Server:          192.168.1.1
Address:         192.168.1.1#53

Name:   mail.eidia.uemf
Address: 192.168.1.20

bamsyr@ubuntu:~$

```

FIGURE 7 – Communication réussie entre la machine cliente et le serveur nslookup de deux adresses du serveur.

5 Configuration du serveur Web avec Apache2

5.1 Installation et configuration

On installe apache2, on crée un fichier Virtual Host, `eidia.uemf.conf`, pour le domaine dans le répertoire de configuration `/etc/apache2/sites-available` et dans le répertoire par dé-

faut d'apache2 on créera le fichier html de notre page test.

5.2 Test du serveur web

Pour tester notre serveur web, l'on tente d'accéder à notre page test uniquement à partir de l'URL qui lui est attribué.

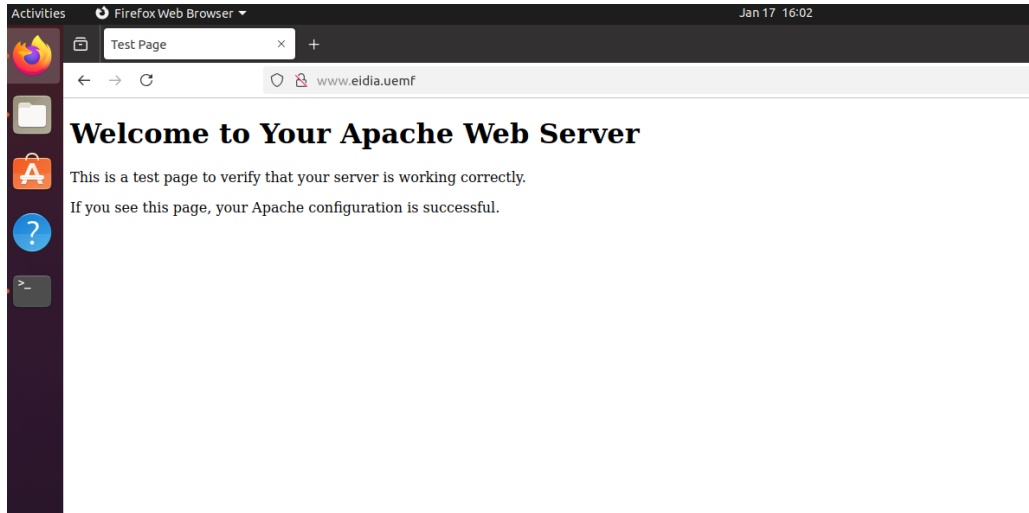


FIGURE 8 – Page web test hébergée sur Apache2.

6 Résultats et observations

On observe que le serveur fonctionne sans soucis, que le client arrive à se connecter et reçoit une adresse du serveur, et la connexion à la page test se fait sans accroc.

7 Conclusion

Le projet bien que complexe, c'est bien déroulé et m'a permis d'acquérir les connaissances nécessaires pour configurer, de façon basique, un serveur dhcp, DNS et Web de façon fonctionnelle.