

Mise en place d'un serveur DNS

Préambule :

Le DNS devra fonctionner sous Linux. C'est un travail à réaliser seul. Démonstration des solutions fonctionnelles avant le 20/11/2023 à 18h. Remise des comptes rendus (envoi sur monge.vf@gmail.com) avant le 24/11/2023 à 18h.

Contexte :

Vous travaillez pour un prestataire de services informatiques. Egnom, la PME qui vous emploie assure diverses missions liées à l'infrastructure matérielle pour le compte d'autres entreprises dépourvues de personnel dédié à ces tâches.

Vous êtes en charge de l'installation du LAN dans les nouveaux locaux d'une PME. Parmi les diverses tâches qui vous sont assignées, la mise en place d'un accès Internet et l'installation d'un serveur DNS local sont prioritaires.

Demande du client :

L'entreprise a acquis un nom de domaine et souhaite héberger localement son site vitrine. Par ailleurs, il est essentiel que les machines de l'entreprise accèdent à Internet et idéalement il s'agit de ne pas autoriser la résolution de certains domaines non nécessaires à l'activité professionnelle de l'entreprise.

Sommaire :

1. Installation des ressources (page 2)
2. Mise en place du DHCP (page 2 à 3)
3. Mise en place du serveur DNS (page 3 à 5)
4. Mise en place de la page Web (page 5)
5. Empêcher la résolution de domaine (page 6 à 7)
6. Effectuer des tests (page 7 à 8)
7. Partie Explication (page 9 à 10)
8. Conclusion (page 11)

1. Installation des ressources

Pour commencer je mets l'adresse IP adéquat (172.30.NP. ?/24).

```
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
  link/ether 00:0c:29:e7:0b:04 brd ff:ff:ff:ff:ff:ff
  altname enp2s1
  inet 172.30.5.50/24 brd 172.30.5.255 scope global ens33
    valid_lft forever preferred_lft forever
  inet6 fe80::20c:29ff:fee7:b04/64 scope link
    valid_lft forever preferred_lft forever
```

Puis j'installe bind9 (j'ai utilisé ce serveur DNS car c'est le plus connu) qui sera le serveur DNS, isc-dhcp-server (j'ai utilisé ce DHCP car je l'avais déjà utilisé) qui sera le serveur DHCP et apache2 (j'ai utilisé ce serveur web car je l'ai déjà utilisé auparavant) qui sera le serveur web :

```
root@NicoDNS:~# apt install bind9
```

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

```
root@debian:/etc/bind# apt install apache2
```

2. Mise en place du DHCP

Définition de DHCP :

Le **Dynamic Host Configuration Protocol (DHCP)** est un protocole réseau utilisé pour attribuer **automatiquement** des **adresses IP** et d'autres paramètres de configuration aux appareils connectés à un réseau. Cela permet aux appareils de se connecter facilement au réseau et de communiquer entre eux.

Le **DHCP** fonctionne selon un modèle client-serveur. Les **clients DHCP** sont les appareils qui souhaitent obtenir une **adresse IP**, tandis que les **serveurs DHCP** sont les ordinateurs qui fournissent les **adresses IP**. Lorsqu'un appareil se connecte à un réseau, il envoie une **requête DHCP** à un **serveur DHCP**. Le **serveur DHCP** attribue ensuite une **adresse IP à l'appareil** et lui fournit d'autres paramètres de configuration, tels que le masque de sous-réseau et la passerelle par défaut.

Le **DHCP** est un protocole important car il simplifie la gestion des réseaux. Sans **DHCP**, les **administrateurs réseau** devraient configurer **manuellement les adresses IP** de chaque appareil sur le réseau. Cela serait fastidieux et sujet à des erreurs.

Pour commencer la configuration je configure le fichier dhcpd.conf avec la commande nano :

```
GNU nano 7.2                                     /etc/dhcp/dhcpd.conf
option domain-name "lunar.monge.";
option domain-name-servers 172.30.5.50;

default-lease-time 21600;
max-lease-time 21600;

ddns-update-style none;

subnet 172.30.5.0 netmask 255.255.255.0 {
    range      172.30.5.50 172.30.5.100;
    option routers      172.30.5.254;
    option subnet-mask  255.255.255.0;
}
```

Ensute je modifie le fichier isc-dhcp-server :

```
GNU nano 7.2                                     /etc/default/isc-dhcp-server *
```

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

Puis je redémarre le service isc-dhcp-server :

```
root@debian:~# service isc-dhcp-server restart
```

Pour finir on vérifie si le service isc-dhcp-server fonctionne :

```
root@debian:~# service isc-dhcp-server status
● isc-dhcp-server.service - LSB: DHCP server
  Loaded: loaded (/etc/init.d/isc-dhcp-server; generated)
  Active: active (running) since Mon 2023-11-20 08:02:13 CET; 1h 23min ago
    Docs: man:systemd-sysv-generator(8)
  Process: 532 ExecStart=/etc/init.d/isc-dhcp-server start (code=exited, status=0/SUCCESS)
    Tasks: 1 (limit: 4644)
   Memory: 7.1M
      CPU: 538ms
     CGroup: /system.slice/isc-dhcp-server.service
             └─607 /usr/sbin/dhcpd -4 -q -cf /etc/dhcp/dhcpd.conf ens33

nov. 20 08:47:13 debian dhcpd[607]: DHCPREQUEST for 10.10.0.190 from a8:1e:84:35:20:b8 via ens33: i>
nov. 20 08:47:28 debian dhcpd[607]: reuse_lease: lease age 1528 (secs) under 25% threshold, reply w>
nov. 20 08:47:28 debian dhcpd[607]: DHCPDISCOVER from a8:1e:84:35:20:b8 (DESKTOP-DGB06LQ) via ens33
nov. 20 08:47:28 debian dhcpd[607]: DHCPoffer on 172.30.5.51 to a8:1e:84:35:20:b8 (DESKTOP-DGB06LQ)>
nov. 20 08:47:28 debian dhcpd[607]: reuse_lease: lease age 1528 (secs) under 25% threshold, reply w>
nov. 20 08:47:28 debian dhcpd[607]: DHCPREQUEST for 172.30.5.51 (172.30.5.50) from a8:1e:84:35:20:b8
nov. 20 08:47:28 debian dhcpd[607]: DHCPACK on 172.30.5.51 to a8:1e:84:35:20:b8 (DESKTOP-DGB06LQ) v>
nov. 20 09:10:06 debian dhcpd[607]: reuse_lease: lease age 2886 (secs) under 25% threshold, reply w>
nov. 20 09:10:06 debian dhcpd[607]: DHCPREQUEST for 172.30.5.51 from a8:1e:84:35:20:b8 (DESKTOP-DGB06LQ)
nov. 20 09:10:06 debian dhcpd[607]: DHCPACK on 172.30.5.51 to a8:1e:84:35:20:b8 (DESKTOP-DGB06LQ) v>
```

3. Mise en place du serveur DNS

Définition de DNS :

Le système de **noms de domaine (DNS)** est un système informatique distribué qui permet de traduire les noms de domaine en **adresses IP**. Les noms de domaine sont les noms que les utilisateurs saisissent dans leur navigateur Web pour accéder à un site Web, tandis que les **adresses IP** sont les numéros uniques qui identifient chaque appareil sur Internet.

Le **DNS** fonctionne comme **un annuaire téléphonique**. Lorsque vous tapez un **nom de domaine** dans **votre navigateur Web**, votre ordinateur envoie **une requête à un serveur DNS** pour demander **l'adresse IP** du **site Web** correspondant. Le **serveur DNS** renvoie alors **l'adresse IP**, ce qui permet à votre navigateur de se connecter au **site Web**.

Le **DNS** est un élément essentiel d'Internet, car il permet aux utilisateurs d'accéder aux sites Web en utilisant des noms de domaine faciles à mémoriser au lieu d'adresses IP difficiles à retenir.

Pour commencer la configuration du serveur DNS on configure le fichier named.conf.options :

```
GNU nano 7.2                               /etc/bind/named.conf.options
options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    forwarders {
        10.10.0.1;
    };

    //=====
    // If BIND logs error messages about the root key being expired,
    // you will need to update your keys.  See https://www.isc.org/bind-keys
    //=====
    dnssec-validation auto;

    listen-on-v6 { any; };
    allow-recursion { any; };
};
```

Puis on modifie le fichier named.conf dans le dossier /etc/bind/ en rajoutant la zone DNS qu'on veut :

```
GNU nano 7.2                               named.conf
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local

include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";

zone "lunar.monge." IN {
    type master;
    file "/etc/bind/db.lunar.monge";
};
```

Ensuite on copie le fichier db.local en dossier db.lunar.monge (le nom db.lunar.monge est dans le fichier named.conf quand vous ajouter votre DNS)

```
root@debian:/etc/bind# cp /etc/bind/db.local /etc/bind/db.lunar.monge|
```

Puis on modifie le fichier db.lunar.monge :

```
root@debian:~# nano db.lunar.monge
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     ns.lunar.monge. root.lunar.monge. (
                      2           ; Serial
                      604800      ; Refresh
                      86400       ; Retry
                     2419200     ; Expire
                     604800 )     ; Negative Cache TTL
;
@       IN      NS      ns.lunar.monge.
NS      IN      A       172.30.5.50
@       IN      A       172.30.5.50
```

Pour finir on redémarre le system bind9 :

```
root@debian:/etc/bind# systemctl restart bind9.service
```

4. Mise en place de la page Web

Pour commencer la mise en place le la page Web on va dans le répertoire /var/www/html/ :

```
root@debian:/etc/bind# cd /var/www/html/
```

Puis on crée ou modifie le fichier index.html :

```
root@debian:/var/www/html# nano index.html |
```

Pour finir on mets ce qu'on veut (pour ma part j'ai mis un paragraphe test et un image dont la source est un lien) :

```
<html>
<p>test<p>
      
<html>
```

5. Empêcher la résolution de domaine

Qu'est-ce que la résolution de domaine ?

La résolution de domaine est le processus de traduction d'un nom de domaine en une adresse IP. Les noms de domaine sont des noms faciles à retenir que les utilisateurs saisissent dans leur navigateur Web pour accéder à un site Web, tandis que les adresses IP sont des numéros uniques qui identifient chaque appareil sur Internet.

Lorsqu'un utilisateur saisit un nom de domaine dans son navigateur Web, son ordinateur envoie une requête à un serveur DNS (Domain Name System) pour demander l'adresse IP du site Web correspondant. Le serveur DNS renvoie alors l'adresse IP, ce qui permet au navigateur de se connecter au site Web.

La résolution de domaine est un élément essentiel d'Internet, car il permet aux utilisateurs d'accéder aux sites Web en utilisant des noms de domaine faciles à mémoriser au lieu d'adresses IP difficiles à retenir.

Pour empêcher la résolution de domaine on configure le fichier named.conf en ajoutant les zone de DNS que vous voulez empêcher la résolution :

```
GNU nano 7.2                                     named.conf
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local

include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";

zone "lunar.monge." IN {
    type master;
    file "/etc/bind/db.lunar.monge";
};

zone "amazon.fr" {
    type master;
    file "/etc/bind/amazon.fr";
};

zone "fnac.com" {
    type master;
    file "/etc/bind/fnac.com";
};
```

[Fermer la lecture de 26 lignes]

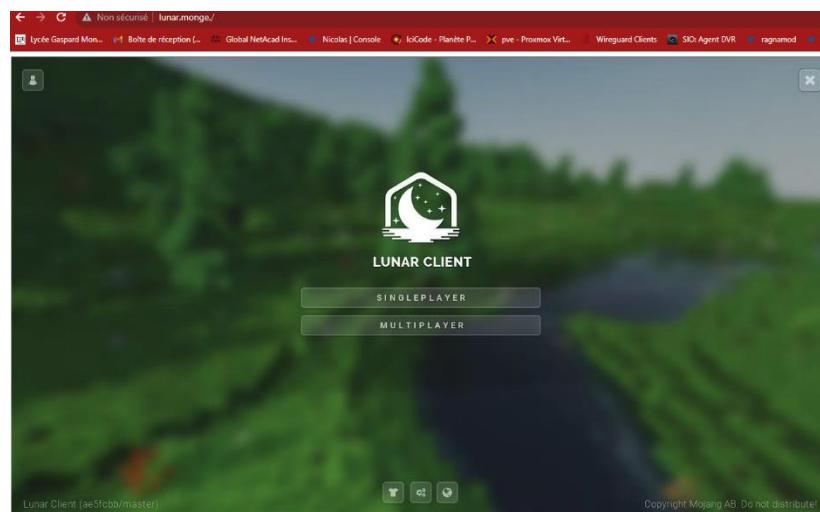
Pour finir on modifie le fichier amazon.fr et fnac.com (les zones DNS que vous voulez empêcher la résolution) :

```
GNU nano 7.2                                         amazon.fr
$TTL    604800
@      IN      SOA     localhost. root.localhost. (
                  2           ; Serial
                  604800    ; Refresh
                  86400     ; Retry
                  2419200   ; Expire
                  604800 )  ; Negative Cache TTL
;
@      IN      NS      localhost.
@      IN      A       127.0.0.1
```

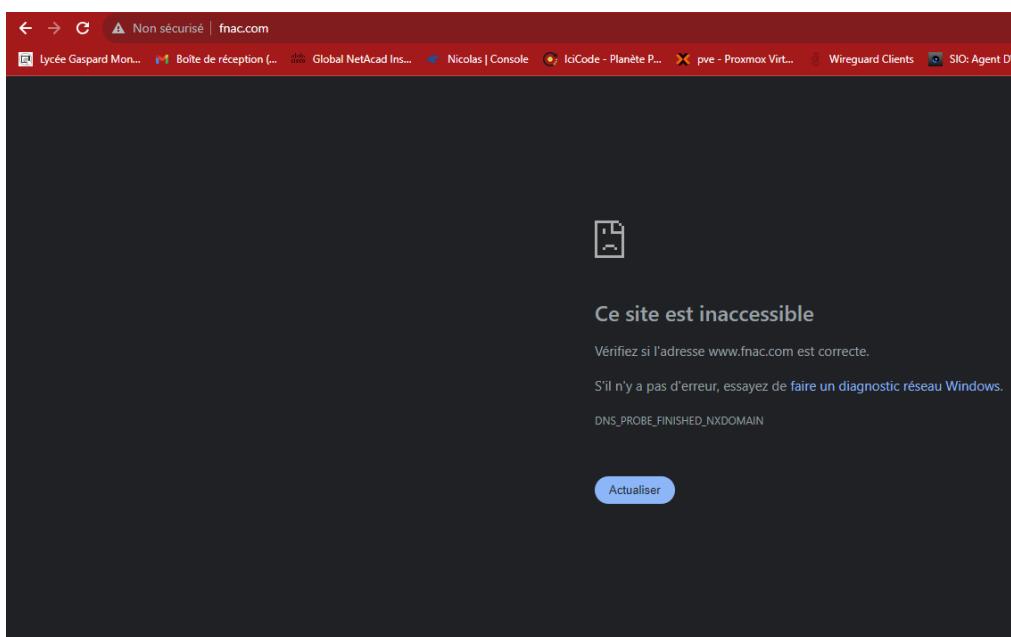
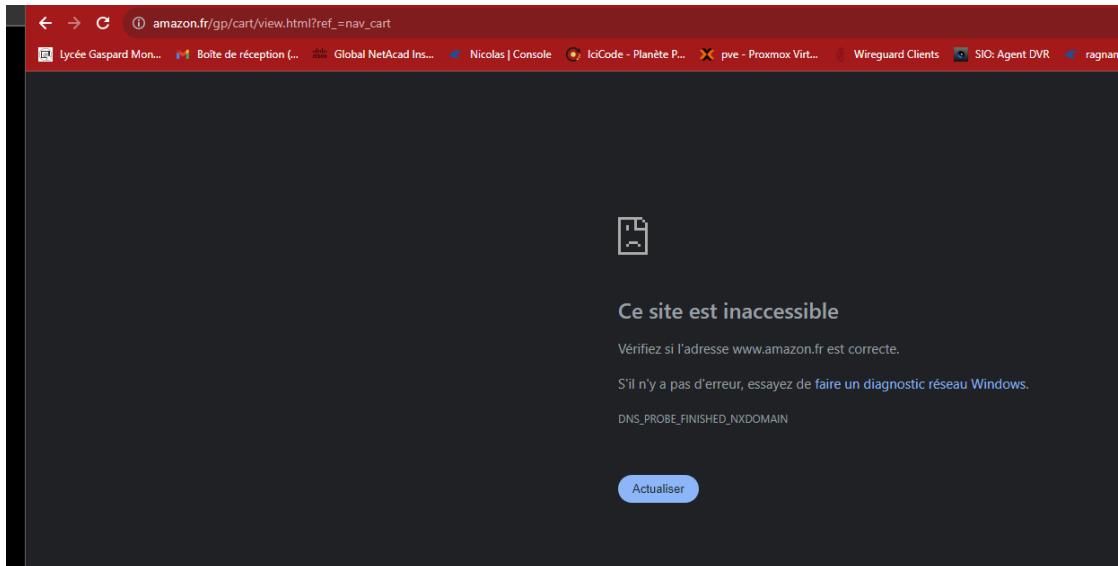
```
GNU nano 7.2                                         fnac.com
$TTL    604800
@      IN      SOA     localhost. root.localhost. (
                  2           ; Serial
                  604800    ; Refresh
                  86400     ; Retry
                  2419200   ; Expire
                  604800 )  ; Negative Cache TTL
;
@      IN      NS      localhost.
@      IN      A       127.0.0.1
```

6. Effectuer des tests

Puis on fait les tests donc là je teste si j'arrive à joindre la page web avec le DNS sur mon poste :



Ensute je teste si j'arrive à joindre la zone DNS amazon.fr et fnac.com sur mon poste :



Puis un poste relié au réseau 10.10.0.0 je teste si j'arrive à joindre mon DNS :

```
C:\Users\Admin1>nslookup lunar.monge
Serveur :  pfSense.home.arpa
Address:  10.10.0.1

Réponse ne faisant pas autorité :
Nom :  lunar.monge
Address:  172.30.5.50

C:\Users\Admin1>
```

7. Partie explication

Dans le fichier de configuration du DHCP (dhcpd.conf) les 2 premières lignes permettent de connecter l'appareil directement au domain.

Ensuite les 2 lignes suivantes sont la durée de connexion.

Puis ensuite on configure le DHCP en mettant le réseau du DHCP (172.30.5.0 avec le masque 255.255.255.0) puis le nombre d'adresse IP disponible (de 50 à 100 donc 50 adresse disponible) pour la suite on mets le routeur qui permet d'avoir de la connexion (172.30.5.254) pour et le masque du réseau (255.255.255.0) :

```
GNU nano 7.2
option domain-name "lunar.monge.";
option domain-name-servers 172.30.5.50;

default-lease-time 21600;
max-lease-time 21600;

ddns-update-style none;

subnet 172.30.5.0 netmask 255.255.255.0 {
    range          172.30.5.50 172.30.5.100;
    option routers    172.30.5.254;
    option subnet-mask 255.255.255.0;
}
```

Dans le fichier isc-dhcp-server on ajoute le port (en IPv4 ou/et IPv6) donc pour savoir le port vous pouvez faire un **ip a** et vous pouvez voir le port pour ma part c'est ens33 :

```
GNU nano 7.2
INTERFACESv4="ens33"
INTERFACESv6=""
```

Ici on ajoute le transitaire (là on va passer le DNS pour se connecter à la racine (monge.) :

```
forwarders {
    10.10.0.1;
};

//=====
// If BIND logs error messages here
// you will need to update
//=====
dnssec-validation auto;

listen-on-v6 { any; };
allow-recursion { any; };
};
```

Puis on ajoute la zone `lunar.monge.` de type maître avec le fichier de configuration du DNS (`db.lunar.monge`) :

```
zone "lunar.monge." IN {
    type master;
    file "/etc/bind/db.lunar.monge";
};
```

Un fichier de configuration BIND définit les paramètres d'un serveur DNS BIND. Les paramètres comprennent l'adresse IP du serveur, le nom du domaine du serveur et les enregistrements DNS que le serveur doit servir.

Dans ce fichier de configuration, les paramètres suivants sont définis :

Le TTL (Time To Live) est défini sur 604 800 secondes, ce qui signifie que les enregistrements DNS dans cette zone peuvent être mis en cache par d'autres serveurs DNS pendant 7 jours.

Le SOA (Start Of Authority) est défini sur `ns.lunar.monge.` `root.lunar.monge.`, ce qui signifie que le serveur DNS faisant autorité pour cette zone est `ns.lunar.monge.`

Le NS (Name Server) est défini sur `ns.lunar.monge.` et 172.30.5.50, ce qui signifie que les deux serveurs DNS sont responsables de la fourniture de réponses pour cette zone.

L'enregistrement A (Address) est défini sur 172.30.5.50, ce qui signifie que l'adresse IP de l'hôte auquel le nom d'hôte @ résout est 172.30.5.50.

```
;;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     ns.lunar.monge. root.lunar.monge. (
                      2                   ; Serial
                      604800            ; Refresh
                      86400             ; Retry
                      2419200           ; Expire
                      604800 )          ; Negative Cache TTL
;
@       IN      NS      ns.lunar.monge.
NS      IN      A       172.30.5.50
@       IN      A       172.30.5.50
```

Ici c'est la même chose juste vous remplacer votre DNS par `localhost` et l'adresse IP par celui du `localhost` (127.0.0.1) :

```
$TTL    604800
@       IN      SOA     localhost. root.localhost. (
                      2                   ; Serial
                      604800            ; Refresh
                      86400             ; Retry
                      2419200           ; Expire
                      604800 )          ; Negative Cache TTL
;
@       IN      NS      localhost.
@       IN      A       127.0.0.1
```

8. Conclusion

Dans ce TP j'ai pu apprendre à utiliser bind9 (création d'un DNS, joindre le DNS à la racine et empêcher la résolution de domaine), puis j'ai pu me rappeler de comment utiliser isc-dhcp-server et apache2.

Et le fait d'apprendre à utiliser bind9 m'a permis de comprendre comment fonctionne les DNS car et les serveurs DNS.

Source :

<https://doc.ubuntu-fr.org/bind9>

<https://bard.google.com/>