



I. Installation du DNS :

Actuellement nos hôtes sont *pingables* l'un envers l'autre par leurs IP, mais ils ne connaissent pas leurs noms sur le réseau.

Que signifie DNS ?

Le DNS (Domain Name System) est un système qui traduit les noms de domaine lisibles par les humains (comme google.com) en adresses IP (comme 8.8.8.8)

A quoi sert le DNS ?

Le DNS est essentiel car il permet de faire correspondre les noms de domaine à des adresses IP. Sans le DNS, il serait nécessaire de mémoriser des adresses IP numériques pour accéder aux sites ou services en ligne, ce qui serait peu pratique. Le DNS facilite ainsi la navigation et la gestion des services sur internet.

Qu'est-ce qu'un FQDN ? Le FQDN (Fully Qualified Domain Name) est un nom de domaine complet qui inclut tous les éléments nécessaires pour localiser un serveur ou une ressource sur internet. Un FQDN comprend :

1. Le nom de l'hôte (par exemple, www),
2. Le domaine de second niveau (par exemple, example),
3. Le domaine de premier niveau (par exemple, .com).

Donner un exemple de FQDN : Un exemple de FQDN est : www.example.com

Vous avez appelé votre PC à la maison « superpc.com » et vous êtes propriétaire de ce FQDN. A quoi cela vous sert-il ?

Posséder un FQDN comme superpc.com permet d'accéder à votre PC via internet à partir de n'importe quel endroit, sans avoir à se souvenir de son adresse IP. Cela vous permet d'héberger un site web ou un service, de rendre votre PC accessible à distance, ou encore de configurer des services spécifiques comme des serveurs de messagerie, des VPN, ou des applications accessibles en ligne.

Pour installer un DNS sur un système linux, il y a plusieurs possibilités qui dépendent du contexte et de l'usage. On peut simplifier les possibilités en deux grandes options :

- ✓ Configuration avec le fichier hosts simple sur chaque machine lorsqu'on est sur un réseau interne ou bien
- ✓ configuration avec un serveur DNS (bind par exemple) si l'on veut installer des services internet.



a) Configuration simple avec le fichier hosts sur chaque PC (ou VM) :

Notre serveur doit faire office de serveur DNS de la zone logesimmo.ev. Pour cela, il faut qu'il soit en IP fixe, or nous avons vu que enp0s8 est en dhcp. Il faut donc le mettre en IP fixe.

Serveur : Commençons par lui mettre une configuration fixe sur cette interface :

Détermination des éléments :

(se référer au schéma de l'architecture virtuelle précédent page)

Quelle va être son IP : **IP fixe du serveur : 172.17.255.250**

Quel va être son masque : **Masque de sous-réseau : 255.255.0.0**

Broadcast : **Adresse de broadcast : 172.17.255.255**

Gateway : **Gateway (passerelle par défaut) : 172.17.255.254**

Le DNS auquel il devra se référer (le DNS de zone supérieure) :

Serveur DNS supérieur : 192.168.1.1

Tester le paramétrage et reporter ci-dessous, votre configuration finale :

```
GNU nano 7.2 /etc/network/interfaces *

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

auto enp0s3
iface enp0s3 inet static
address 10.30.1.1
netmask 255.255.0.0
broadcast 10.30.255.255

auto enp0s8
iface enp0s8 inet static
address 10.0.3.15
netmask 255.255.255.0
broadcast 10.0.3.255
dns-server 192.168.1.1
```

Rappels :

Pour paramétrer les interfaces réseau :

/etc/network/interfaces

Procéder au paramétrage pour obtenir :

```
# The loopback network interface
auto lo
iface lo inet loopback

auto enp0s3
iface enp0s3 inet static
address 10.30.1.1
netmask 255.255.0.0
broadcast 10.30.255.255
dns-nameservers 10.0.3.15

auto enp0s8
iface enp0s8 inet static
address 10.0.3.15
netmask 255.255.255.0
broadcast 10.0.3.255
gateway 10.0.3.2
dns-server 192.168.1.1
```

DNS : paramétrons maintenant le DNS de chaque machine virtuelle du réseau interne de façon « simple » :

Rappel : pour mettre les ip des serveurs avec leur nom : /etc/hosts

Exemple :



Client

127.0.0.1 localhost
127.0.0.1 clientlogesimmo
10.30.5.2 clientlogesimmo
10.30.1.1 serveurlogesimmo.logesimmo.ev

Serveur :

127.0.0.1 localhost
127.0.0.1 serveurlogesimmo.logesimmo.ev
10.30.1.1 serveurlogesimmo.logesimmo.ev
10.30.5.2 clientlogesimmo

- Faire un test de ping pour vérifier la communication avec les noms d'hôtes.



Ecrivez ici les commandes à faire pour tester cela :

Sur le serveur

Ping 10.30.5.2

Sur le client

Ping 10.30.5.2

- Expliquez ci-dessous ce que vous avez fait, pourquoi vous l'avez fait et comment vous avez vérifié la bonne réalisation du travail. Rédigez correctement et sur au moins 10 lignes !

Pour configurer le DNS de manière simple entre le serveur et le client, j'ai modifié le fichier /etc/hosts sur chaque machine pour y associer les noms d'hôtes et leurs adresses IP respectives. Cela permet à chaque machine de résoudre les noms d'hôtes localement sans dépendre d'un serveur DNS externe. Sur le serveur, j'ai ajouté l'entrée pour le client, et sur le client, l'entrée pour le serveur. Ensuite, j'ai vérifié la connectivité en utilisant la commande ping sur chaque machine, en testant le nom d'hôte de l'autre. Les pings réussis ont confirmé que la résolution des noms était bien en place, permettant une communication directe entre les machines via leurs noms d'hôtes.

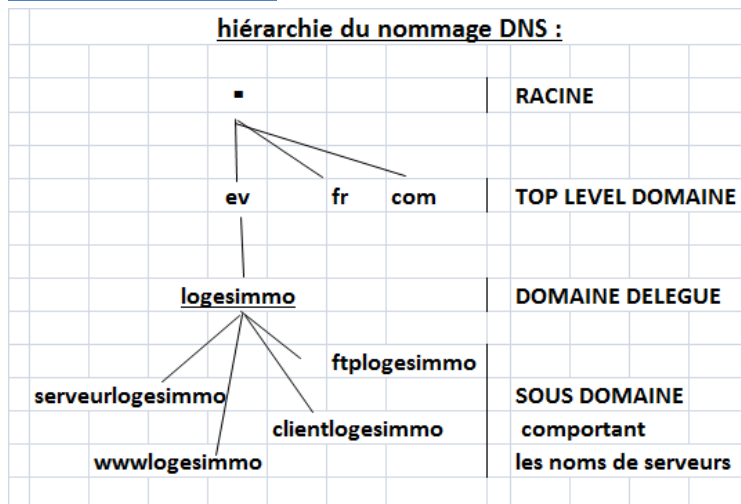
b) Configuration d'un service DNS avec Bind

A ce niveau, si vous êtes perdu, vous pouvez récupérer les VM niveau 1 pour la suite du travail.

(la version 2 comporte des paramètres dns inutiles) A la fin de cette partie, la version sera la numéro 3

Voici un schéma qui synthétise la hiérarchie DNS de notre sous-domaine à créer, avec les noms d'hôtes que l'on indiquera :

Schéma de zone DNS



Le serveur DNS sera installé sur le serveur logesimmo. C'est donc lui qui fera autorité sur notre sous-domaine logesimmo.ev (sous-domaine fictif sans relation avec un réel top level domain)

Donner le FQDN de clientlogesimmo dans cette arborescence :

Le FQDN de clientlogesimmo dans cette arborescence est : clientlogesimmo.ev.

Nous allons utiliser le logiciel Bind dont la version actuelle est la 9. Le logiciel Bind est un logiciel libre, gratuit, développé pour les systèmes UNIX. Bind = Berkeley Internet Name Daemon(ou Domain)

La mise en place d'un serveur DNS va permettre de créer des zones de noms et donc de déclarer des FQDN.

1 - installation de bind9 :

Apt-get install bind9 (dernière version)

Rappel : faire ces manipulations sur le serveur (10.30.1.1)

Il va falloir, soit insérer le CD, soit aller sur internet pour le chargement des paquets

Trois fichiers sont importants dans /etc/bind :

- ✓ Named.conf.options
- ✓ Named.conf.local
- ✓ Named.conf.default-zones

(stratégie sur Debian 9)

- Entourez ces trois fichiers dans la copie-écran :
- Vérifiez la présence de ces trois fichiers dans votre installation.

Utilité de chaque fichier :

```

root@serveurlogesimmo:/etc/bind# ls -l
total 52
-rw-r--r-- 1 root root 3923 août 28 09:36 bind.keys
-rw-r--r-- 1 root root 237 août 28 09:36 db.0
-rw-r--r-- 1 root root 271 août 28 09:36 db.127
-rw-r--r-- 1 root root 237 août 28 09:36 db.255
-rw-r--r-- 1 root root 353 août 28 09:36 db.empty
-rw-r--r-- 1 root root 270 août 28 09:36 db.local
-rw-r--r-- 1 root root 3171 août 28 09:36 db.root
-rw-r--r-- 1 root bind 463 août 28 09:36 named.conf
-rw-r--r-- 1 root bind 490 août 28 09:36 named.conf.default-zones
-rw-r--r-- 1 root bind 165 août 28 09:36 named.conf.local
-rw-r--r-- 1 root bind 890 déc. 31 16:35 named.conf.options
-rw-r--r-- 1 bind bind 77 déc. 31 16:35 rndc.key
-rw-r--r-- 1 root root 1317 août 28 09:36 zones.rfc1918
  
```

Named.conf.default-zones	Named.conf.local	Named.conf.options
Indique à bind où aller pour chercher les configurations nécessaires. Il n'y a rien à modifier lors d'une utilisation normale.	Configuration de zone locale : Il faudra y créer les deux fichiers qui vont déclarer notre zone locale et sa reverse zone : - logesimmo.ev.zone - logesimmo.reverse	C'est là que vous allez paramétrer vos options de dns. Il ne faut modifier que la partie qui indique où écouter en plus du loopback.

- Nos options de DNS (voir le schéma de zone ci-dessus et le plan d'adressage) :

La zone (domaine appelé « zone » sur Bind) : logesimmo.ev

Nom du fichier de zone : logesimmo.ev.zone

2-Configuration des fichiers de bind9 :Commençons par le fichier **named.conf.local** :

```
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "logesimmo.ev" IN {
    type master;
    file "/var/cache/bind/logesimmo.ev.zone";
    allow-update { none; };
};

zone "0.30.10.in-addr.arpa" IN {
    type master;
    file "/var/cache/bind/logesimmo.ev.reverse";
    allow-update { none; };
};

~
"/etc/bind/named.conf.local" 19L, 397C
```

Editez ce fichier et modifier si besoin, comme indiqué sur la copie-écran ci-contre.

On indique où sera situé le fichier de zone

Puis où sera celui de la zone inverse

(Les noms de fichiers sont à choisir comme vous le voulez, seul le chemin est imposé).

Voyons ensuite le fichier **named.conf.options** :

Il sert à paramétrer les zones de base de notre DNS.

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

    //Puisque on est en situation virtuelle et q'on n'a pas de nom acheté, logesimmo
    .ev est une situation de cours, on va dire que notre DNS qui joue le role de google,
    pour les itératives, sera le serveur dns de notre passerelle 192.168.1.1 (
    a adapter à votre cas) :
    forwarders {
        192.168.1.1;
    };

    // à ne pas modifier :

    dnssec-validation auto;

    auth-nxdomain no;    # conform to RFC1035
    listen-on-v6 { any; };

    // on indique à bind d'écouter sur l'IP externe en plus du loopback :

    listen-on { 127.0.0.1; 10.30.1.1; };

    // on précise de répondre aux requetes locales du réseau :

    allow-query { localhost; 10.30.0.0/16; };
    version none;
};

~
"/etc/bind/named.conf.options" 25L, 700C
```

Editez ce fichier et modifier si besoin, comme indiqué sur la copie-écran ci-contre.

Il n'y a pas grand-chose à modifier :

Ecoute sur notre passerelle extérieure (le forwarder définit les DNS récursifs (ceux sollicités par Bind quand il ne peut pas répondre lui-même).

Ecoute passerelle interne de la zone

Etendue de la zone locale

3- création des fichiers zones :

Il faut maintenant créer les deux fichiers de zone :



Fichier de zone directe

Un fichier de zone contient l'ensemble des services liés au domaine.

On y inscrit les lignes suivantes (voir ci-contre)

Les points après le nom de domaine sont obligatoires.

```
$TTL      3600
@          IN      SOA      serveurlogesimmo.logesimmo.ev.  root.logesimmo.ev. (
2018010101 ; Serial
3600       ; Refresh
600        ; Retry
86400      ; Expire
600)       ; Negative Cache TTL
; Serveurs de noms :
@          IN      NS       serveurlogesimmo.logesimmo.ev.
serveurlogesimmo      IN      A          10.30.1.1
wwwlogesimmo          IN      CNAME      serveurlogesimmo
ftplogesimmo          IN      CNAME      serveurlogesimmo

~
"/var/cache/bind/logesimmo.ev.zone" 14L, 351C
```

Que signifie TTL ?

Le TTL est une valeur utilisée dans les requêtes DNS pour indiquer la durée pendant laquelle une réponse DNS peut être mise en cache par un serveur ou un client avant de devoir être actualisée. Il est exprimé en secondes et permet de limiter la durée de vie d'une entrée DNS dans les caches.

A quoi sert la ligne SOA

La ligne SOA dans un fichier de zone DNS définit le serveur principal (maître) qui est responsable de la gestion des données DNS de ce domaine. Elle contient également des informations comme le contact de l'administrateur, le numéro de série de la zone, et les intervalles pour la mise à jour et l'actualisation des données DNS.

Décomposez le numéro Serial (aide : YYYYMMDDnum)

Le numéro de série est souvent formaté comme YYYYMMDDnum, où :

- YYYY est l'année,
- MM est le mois,
- DD est le jour,
- num est un chiffre supplémentaire utilisé pour représenter plusieurs modifications faites dans la même journée.

Quel est son rôle ?

Le numéro de série indique la version des données de la zone DNS. Chaque fois que des modifications sont apportées à la zone, ce numéro est incrémenté. Les serveurs DNS secondaires vérifient ce numéro pour savoir s'ils doivent actualiser leurs copies des données de la zone.

A quoi sert la ligne @ IN NS

Cette ligne dans un fichier de zone DNS spécifie les serveurs de noms (NS) pour le domaine. Le symbole @ représente le domaine racine de la zone, et la ligne informe quel serveur DNS est responsable de ce domaine.

A quoi servent les lignes qui suivent ?

Les lignes qui suivent indiquent généralement les adresses des serveurs de noms spécifiés dans la ligne NS. Elles peuvent inclure des enregistrements A (adresse IP pour les serveurs de noms) ou AAAA (adresse IPv6). Cela permet aux résolveurs DNS de connaître les adresses des serveurs de noms pour interroger le domaine.

➤ **Créez ce fichier logesimmo.ev.zone dans /var/cache/bind . Et imprimez-le (ou copie-écran) Insérer cette impression en page 35**

Afin de trouver le nom de domaine correspondant à une adresse IP, nous devons créer aussi un fichier de zone inverse.

Pour plus de facilité, vous pouvez d'abord copier le fichier de zone directe en lui donnant le nom de la zone inverse et ensuite, l'ouvrir et le modifier comme ci-contre.

```
$TTL      3600
@         IN      SOA      serveurlogesimmo.logesimmo.ev. root.logesimmo.ev. (
        2018010101      ; Serial
        3600            ; Refresh
        600             ; Retry
        86400           ; Expire
        600)            ; Negative Cache TTL
; Serveurs de noms :
@         IN      NS       serveurlogesimmo.logesimmo.ev.
@         IN      PTR      logesimmo.ev.

serveurlogesimmo      IN      A       10.30.1.1
1                     IN      PTR      serveurlogesimmo.logesimmo.ev.

~
"/var/cache/bind/logesimmo.ev.reverse" 16L, 338C
```

- **Créez ce fichier `logesimmo.ev.reverse` dans `/var/cache/bind` . Et imprimez-le. Insérer cette impression en page 35**

Les informations du début sont sensiblement les mêmes. Il faut cependant ensuite indiquer en dessous de chaque adressage direct, l'adressage de retour (le reverse) avec une commande PTR (pointeur).

4- tests de la configuration :

Il faut tester si notre configuration de bind est correcte maintenant :

Tapez les commandes à gauche et vérifiez que le système répond comme à droite :

#named-checkconf -z

```
root@seveurlogesimmo:~# named-checkconf -z
zone logesimmo.ev/IN: loaded serial 2018010101
zone 0.30.10.in-addr.arpa/IN: loaded serial 2018010101
zone localhost/IN: loaded serial 2
zone 127.in-addr.arpa/IN: loaded serial 1
zone 0.in-addr.arpa/IN: loaded serial 1
zone 255.in-addr.arpa/IN: loaded serial 1
```

#named-checkzone
logesimmo.ev
/var/cache/bind/logesimmo.ev.zone

```
root@seveurlogesimmo:~# named-checkzone logesimmo.ev /var/cache/bind/logesimmo.ev.zone
zone logesimmo.ev/IN: loaded serial 2018010101
OK
```

#named-checkzone
0.30.10.in-addr.arpa
/var/cache/bind/logesimmo.ev.reverse

```
root@seveurlogesimmo:~# named-checkzone 0.30.10.in-addr.arpa /var/cache/bind/logesimmo.ev.reverse
zone 0.30.10.in-addr.arpa/IN: loaded serial 2018010101
OK
```


Si ces commandes vous renvoient des erreurs, c'est qu'il y a des erreurs de saisie dans vos fichiers précédents. **Faites des copies écran de ces commandes dans votre machine et insérez les en page 35**



Si tout est OK, il faut lancer le service DNS (en root, sur le serveur DNS) :

```
# systemctl enable bind9.service
```

```
# systemctl start bind9.service
```

5- tests de fonctionnement :

Pour lancer les outils de diagnostic, il faut peut-être les installer : Apt-get install dnsutils (il faut le CD room)

a) Test sur le serveur :

Faites des copies écran de cette commande dans votre machine et insérez les en page 35

La commande dig permet de spécifier le type de serveurs ou de machines que l'on veut contacter. Si elle renvoie une réponse c'est donc que le système DNS fonctionne (du moins en local !)

On voit sur cet écran que logesimmo.ev répond et que la commande a été passée sur la machine où est installé le serveur DNS (127.0.0.1#53)

```
root@seveurlogesimmo:~# dig logesimmo.ev

; <<>> DiG 9.10.3-P4-Debian <<>> logesimmo.ev
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 16711
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

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

;; AUTHORITY SECTION:
logesimmo.ev.                600     IN      SOA     serveurlogesimmo.logesimmo.e
ot.logesimmo.ev.logesimmo.ev. 2016072301 3600 600 86400 600

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Tue Jan 02 17:04:40 CET 2018
;; MSG SIZE rcvd: 112
```

b) Test(s) sur le client :

- Attention, si vous avez suivi ce TP, pour l'instant, le client n'est pas configuré avec un DNS ou le bon DNS. Donc première chose à faire c'est lui dire qui est le serveur DNS :

```
#vim.tiny /etc/resolv.conf
```

```
Search logesimmo .ev
Nameserver 10.30.1.1
```

- Puis on redémarre le service :

```
# systemctl restart networking.service
```
- On installe les outils de diagnostic :

```
# Apt-get install dnsutils
```
- Puis on peut faire les tests :
 - Que donne le test dig logesimmo.ev ?

Lorsque j'ai lancé la commande `dig logesimmo.ev` depuis le client, j'ai obtenu une réponse qui indique si le serveur DNS a correctement résolu le nom de domaine. Par exemple, la section de réponse affichera que le nom `logesimmo.ev` est associé à l'adresse IP `10.30.1.1`. Cela signifie que la résolution DNS est bien effectuée et que le serveur DNS fonctionne correctement.

- Quelle adresse y a-t-il à la ligne `SERVER` ?

En ce qui concerne la ligne `SERVER`, elle contient l'adresse IP du serveur DNS qui a traité la requête. Dans ce cas, l'adresse visible devrait être `10.30.1.1`, ce qui correspond à l'adresse du serveur DNS configuré.

- Quel autre test peut-on faire depuis le client pour voir si les noms d'hôtes sont bien configurés ?

Pour vérifier si les noms d'hôtes sont bien configurés, vous pouvez également utiliser la commande `ping logesimmo.ev` depuis le client. Si la commande renvoie des réponses, cela prouve que la résolution DNS est fonctionnelle et que le client peut communiquer avec le serveur. Une autre méthode de test consiste à utiliser la commande `nslookup logesimmo.ev`, qui permet de vérifier si le nom de domaine est bien résolu et de voir quel serveur DNS a répondu à la requête.

Le point :

A ce niveau, nous avons :

- ☐ Notre serveur `logesimmo` sur héberge un DNS qui fonctionne sur le réseau `10.30.0.0/16`
- ☐ Les interfaces de notre serveur sont configurées correctement ;
- ☐ Notre client fonctionne correctement et contact le serveur avec son nom d'hôte.

Intégrez ces données dans une fiche de test, testez. Vous ne passez au niveau suivant que si tout est OK

c) Autres méthodes :

- Sur le client :

fichier `/etc/resolv.conf`:

```
# Generated by NetworkManager
nameserver IP du serveur de nom
nameserver 8.8.8.8
```

A retenir :

Le fichier `resolv.conf` permet de paramétrer un (ou plusieurs) serveur DNS sur un client.

(rem : le dns ne sera actif que lorsque le service aura été paramétré sur le serveur !)

✓ Serveur : Installation du serveur de noms

Ancienne méthode : `/etc/hostname`

On y met son nom en FQDN : `strech.logesimmo.ev`

Et on relance le `hostname`

Nouvelle méthode : `hostnamectl set-hostname NOM`

Puis `hostnamectl` pour vérifier :

```
root@strech:~# hostnamectl
Static hostname: strech.logesimmo.ev
Icon name: computer-vm
Chassis: vm
Machine ID: 4e3f9f32319348dca26153c9f95f
Boot ID: 6eb013f74c4d4277970abf0bf454
Virtualization: oracle
Operating System: Debian GNU/Linux 9 (stretch)
Kernel: Linux 4.9.0-4-amd64
Architecture: x86-64
```



Copie des fichiers de zone et des tests faits

```
root@debianserveurs:/etc/bind# cd named.conf.local
bash: cd: named.conf.local: N'est pas un dossier
root@debianserveurs:/etc/bind# nano named.conf.local
root@debianserveurs:/etc/bind# nano named.conf.local
root@debianserveurs:/etc/bind# nano named.conf.options
root@debianserveurs:/etc/bind# nano named.conf.options
root@debianserveurs:/etc/bind# cd ..
root@debianserveurs:/etc# cd ..
root@debianserveurs:/# cd var
root@debianserveurs:/var# cd/cache
bash: cd/cache: Aucun fichier ou dossier de ce type
root@debianserveurs:/var# ls
backups  cache  lib  local  lock  log  mail  opt  run  spool  tmp
root@debianserveurs:/var# cd cache
root@debianserveurs:/var/cache# cd binf
bash: cd: binf: Aucun fichier ou dossier de ce type
root@debianserveurs:/var/cache# cd bind
root@debianserveurs:/var/cache/bind# touch logesimmo.ev.zone
root@debianserveurs:/var/cache/bind# touch logesimmo.ev.reverse
root@debianserveurs:/var/cache/bind# cat logesimmo.ev.zone
root@debianserveurs:/var/cache/bind# ls
logesimmo.ev.reverse  managed-keys.bind
logesimmo.ev.zone     managed-keys.bind.jnl
root@debianserveurs:/var/cache/bind# █
```

Fiche de test « *DNS* »

Objectifs : Vérifier le bon fonctionnement de la résolution des noms de domaine à l'aide d'un serveur DNS configuré sur un réseau interne.

Nom du testeur : Cyril RAMBAUD		Environnement, machine, système testé : Virtual Box		
Date du test : 09/10/2024 Heure du test : 14h		Nom ou but du test : Vérification de la résolution DNS pour le domaine logesimmo.ev.		
Liste des besoins étudiés	Liste des moyens mis en place pour répondre à ces besoins	Tests effectués (éventuellement par qui)	Résultats	Commentaires
Configuration du serveur DNS pour logesimmo.ev.	Installation et configuration d'un serveur DNS sur l'adresse IP 10.30.1.1. Le fichier de zone pour logesimmo.ev a été correctement configuré, y compris les enregistrements A, SOA et NS nécessaires.	dig logesimmo.ev sur le serveur DNS	Le serveur DNS a correctement résolu le nom logesimmo.ev, et l'adresse IP retournée est 10.30.1.1.	Le serveur DNS est fonctionnel localement.
Vérification de la résolution DNS depuis le client.	Configuration du fichier /etc/resolv.conf du client pour pointer vers le serveur DNS 10.30.1.1. Redémarrage des services réseau et installation de l'outil dnsutils pour réaliser les tests depuis le client.	dig logesimmo.ev et ping logesimmo.ev sur le client	Le client a réussi à résoudre logesimmo.ev, et les réponses de ping montrent que la communication est possible avec l'IP 10.30.1.1.	La résolution DNS fonctionne correctement depuis le client.
Test de la communication entre les hôtes par nom DNS.	Test de ping des noms d'hôtes pour vérifier la résolution des noms sur le réseau interne et la connectivité réseau.	ping logesimmo.ev et ping 10.30.1.1 sur le client.	Les pings vers logesimmo.ev et l'adresse IP du serveur ont bien fonctionné.	La connectivité réseau est assurée et la résolution des noms d'hôtes fonctionne correctement.

Remarques :

Le serveur DNS a répondu correctement aux requêtes locales, confirmant une configuration adéquate. Après avoir configuré l'adresse IP du serveur DNS dans le fichier `/etc/resolv.conf`, le client a pu résoudre les noms d'hôtes comme `logesimmo.ev` sans difficulté. Les temps de réponse lors des tests de ping entre le client et le serveur étaient satisfaisants, indiquant une communication fluide au sein du réseau interne. Les fichiers de zone DNS, avec les enregistrements A, NS, et SOA, sont bien paramétrés, ce qui a permis une résolution efficace des noms de domaine. Pour améliorer encore la performance, des ajustements du TTL et la mise en place d'un serveur DNS secondaire pourraient être envisagés afin d'optimiser la disponibilité et la gestion des requêtes DNS.



Compte-rendu à faire

Le compte rendu va être inclus dans la fiche E5. En utilisant l'annexe ci-après , réalisez ici votre compte rendu jusqu'à l'étape DNS.

I. Présentation du contexte

1. Présentation de l'entreprise

L'entreprise est une agence immobilière nouvellement créée à Évry, spécialisée dans la location immobilière. Pour soutenir ses activités, elle nécessite la mise en place complète d'un système d'information (SI) qui assure la gestion des opérations internes, la communication réseau et le partage sécurisé des ressources.

2. Problématique du travail

La mission consiste à mettre en place un SI complet incluant l'architecture réseau, un plan d'adressage IP, la configuration d'un serveur Linux, et l'intégration de clients Windows. Le défi est de garantir la cohabitation entre ces deux environnements, en prenant en compte la gestion des droits, des comptes utilisateurs, des partages de fichiers, et des services essentiels comme DNS et DHCP.

II. Présentation du travail réalisé

1. Conditions de travail

Le travail a été réalisé individuellement avec l'utilisation d'outils spécifiques, entre autres : machines virtuelles, routeurs configurables, services DNS et SAMBA. Le projet s'est étalé sur plusieurs jours, avec des tests itératifs pour valider la configuration.

2. Activités

réalisées

- a) **Cahier des charges** : Un document formalisé a été élaboré pour définir les besoins de l'entreprise, tels que les services réseau, le partage de ressources et la gestion des comptes utilisateurs.
- b) **Plan d'adressage** : Un plan d'adressage IP a été défini, structurant les adresses des différentes machines du réseau pour une communication optimale.
- c) **Installation du routeur** : Un routeur a été installé pour connecter les différentes machines du réseau et assurer la redirection du trafic.
- d) **Installation du serveur** : Un serveur Linux a été installé pour héberger les services de base du réseau.
- e) **Configuration du serveur** : Le serveur a été configuré pour inclure les services nécessaires tels que DNS et DHCP.
- f) **Fiche de test** : Une fiche a été élaborée pour tester l'adressage, les configurations réseau et les services DNS.
- g) **Installation des services DNS/DHCP** : Le DNS a été configuré pour résoudre les noms de domaine locaux (logesimmo.ev), et le service DHCP pour attribuer des adresses IP dynamiques aux clients.
- h) **Partage SAMBA** : Le service SAMBA a été installé pour permettre le partage de fichiers entre

les utilisateurs Windows et Linux, avec une gestion fine des droits d'accès.

i) **Tests** : Des tests fonctionnels ont été effectués pour valider la configuration DNS et vérifier la connectivité entre les différentes machines.

j) **Compte rendu** : Un compte rendu final a été rédigé pour documenter les étapes, les résultats des tests, et les améliorations potentielles.

III. Conclusion personnelle

Ce projet a permis de réaliser une intégration réussie entre un serveur Linux et des clients Windows, en garantissant un réseau fonctionnel et sécurisé. La cohabitation entre les environnements a été facilitée par la mise en place du DNS et du partage SAMBA, assurant une communication fluide et un accès sécurisé aux ressources partagées. Cette expérience m'a permis de consolider mes compétences en administration réseau, notamment dans la configuration de serveurs et services essentiels pour une entreprise.



Annexe : méthodologie du Compte-rendu

Le compte rendu va être inclus dans la fiche E5

Structure du compte rendu :

I. Présentation du contexte

1) Présentation de l'entreprise

Création agence Evry

Location immobilière

Mise en place du SI

2) Problématique du travail

Création complète d'un SI : archi, adressage, plan, comptes, droits, partage et gestion d'une cohabitation entre un serveur linux et des clients windows.

II. Présentation du travail réalisé

1) Conditions de travail

Individuel, Méthodes outils et dates

2) Activités réalisées

- a) Cahier des charges
- b) Plan d'adressage
- c) Installation routeur
- d) Installation serveur
- e) Configuration serveur
- f) Fiche de test
- g) Installation des services : DNS DHCP
- h) Mise en place du partage SAMBA (identifiants, mdp...)
- i) Tests
- j) Compte rendu

III. Conclusion personnelle