

IUT DE COLMAR

SAÉ - CONCEVOIR UN RÉSEAU MULTI-SITES

ANNÉE 2022-23

Partie services réseaux

MARTIN BAUMGAERTNER

10 février 2023

Table des matières

1	Introduction	2
2	les DNS	2
2.1	Explication	2
2.2	Le DNS primaire	2
2.2.1	Zone de résolution directe	2
2.2.2	Zone de résolution inverse	4
2.2.3	Test du premier DNS	5
2.3	Le DNS secondaire	5
2.3.1	Zone de résolution directe	5
2.3.2	Zone de résolution inverse	6
2.3.3	Test du DNS secondaire	6
3	le DHCP	7
3.1	Explication du DHCP	7
4	Mise en place du DHCP	7
4.0.1	test du serveur DHCP	9
5	Le serveur WEB	9
6	Conclusion	10

1 Introduction

Le but principale de cette SAE fût bien sûr de développer un réseau informatique qui s'étend sur plusieurs sites. Pour cela, mes camarades ont utilisés des protocoles de routage propose à leurs besoins. Pour ma part, j'ai choisi de mettre en place les services réseaux. Donc bien entendu, les protocoles et technologies utilisés ont été différentes. Je vais donc vous expliquer à travers ce rapport, comment j'ai mis en place les différents services réseaux nécessaires au projet, à savoir ; les DNS, le serveur DHCP, le serveur WEB et le serveur mail.

2 les DNS

2.1 Explication

Pour la mise en place des DNS, j'ai utilisé le logiciel Bind9 le logiciel libre Bind9, présent exclusivement sur Linux. J'ai fait le choix de ce logiciel car c'était celui que nous utilisions pendant les cours de M. BINDEL en début d'année, mais également celui que nous utilisions en TP l'an dernier avec Mme LACROIX. Il y aura donc deux DNS, respectivement nommés de la manière suivante `ns1.ucexchange.com` et `ns2.ucexchange.com`.

2.2 Le DNS primaire

2.2.1 Zone de résolution directe

Pour la mise en place du DNS primaire, j'ai créé un conteneur LXC sous ubuntu. J'y ai donc installé Bind9 avec la commande suivante : `sudo apt install bind9`. Puis, j'ai configuré les fichiers de configuration de bind9 qui se trouvent tous dans le répertoire `/etc/bind/`. Pour commencer, il faut déclarer une zone. Pour ce faire, il faut éditer le fichier `named.conf.local`. Voici la configuration que j'ai mis dans ce fichier :

```
zone "ns1.ucexchange.com" {  
    type master;  
    file "/etc/bind/db.ns1.ucexchange.com";  
};
```

FIGURE 1 – Déclaration de la zone de résolution directe

Cette déclaration permet de dire au bind9 où il doit chercher le fichier de zone de résolution directe. Ce fichier se trouve dans le répertoire `/etc/bind/` et s'appelle `db.ns1.ucexchange.com` que nous allons voir ci-dessous :

```

$TTL      86400
@         IN      SOA      ns1.ucexchange.com. root.localhost. (
                                1           ; Serial
                                604800      ; Refresh
                                86400       ; Retry
                                2419200     ; Expire
                                86400      ) ; Negative Cache TTL
;
         IN      A        10.242.220.2
         IN      NS       ucexchange.com.
www      IN      A        10.242.220.2
@        IN      NS       localhost.
ucexchange.com. IN      MX      10 ucexchange.com.

```

FIGURE 2 – Configuration de la zone de résolution directe

Voici donc le fichier de configuration pour la résolution directe de mon premier DNS. Je vais vous expliquer les lignes de manière détaillée ci-après :

- **\$TTL 86400** définit le Time To Live par défaut pour les enregistrements dans cette zone en secondes. Cela signifie que les informations contenues dans ces enregistrements ne seront pas mises à jour plus souvent que toutes les 86400 secondes (24 heures).
- **@ IN SOA ns1.ucexchange.com. root.localhost.** définit le début de l'enregistrement de début de zone (SOA). Le caractère @ représente le nom de domaine lui-même, ici **ucexchange.com**. **SOA** indique que cet enregistrement est un enregistrement **SOA** et **ns1.ucexchange.com.** est le nom du serveur d'origine pour cette zone. **root.localhost.** est l'adresse e-mail du responsable technique pour cette zone.
- **IN NS ucexchange.com.** définit un enregistrement **NS** (Name Server) pour le nom de domaine @, indiquant que le serveur de nom **ucexchange.com** est responsable de la résolution des noms pour ce domaine.
- **www IN A 10.242.220.2** définit une adresse IPv4 pour le nom de domaine **www**.
- **ucexchange.com. IN MX 10 ucexchange.com.** définit un enregistrement **MX** (Mail Exchange) pour le nom de domaine "ucexchange.com." indiquant que le serveur de messagerie "ucexchange.com" est responsable de la gestion des courriels pour ce domaine.

2.2.2 Zone de résolution inverse

Pour la zone de résolution inverse, j'ai utilisé le même principe que pour la zone de résolution directe. Mais il faut déclarer la zone et son fichier. Voici comment j'ai déclaré la zone de résolution inverse dans le fichier `named.conf.local` :

```
zone "10.in-addr.arpa" {  
    type master;  
    file "/etc/bind/ns1.ucexchange.inv";  
};
```

FIGURE 3 – Déclaration de la zone de résolution indirecte

Comme avant, je viens définir dans quel fichier sera enregistré ma configuration de zone indirecte.

La spécificité de la résolution indirecte est qu'on va donc mettre à l'envers les adresses IP dans les fichiers de configuration comme vous pouvez le constater ci-dessous :

```
;   
; BIND reverse data file for local loopback interface   
;   
$TTL      604800   
@         IN      SOA      ns1.ucexchange.com. root.localhost. (   
                                1           ; Serial   
                                604800      ; Refresh   
                                86400      ; Retry   
                                2419200    ; Expire   
                                604800 )   ; Negative Cache TTL   
;   
@         IN      NS       ucexchange.com.   
2.220.242 IN      PTR      ucexchange.com.
```

FIGURE 4 – Configuration de la zone de résolution indirecte

- le terme `IN PTR` signifie que c'est une zone de résolution indirecte en elle permet de faire l'inverse que la zone de résolution directe. Cette dernière va permettre de transformer un nom en adresse IP. Tandis que la zone de résolution inverse va permettre de transformer une adresse IP en nom.

2.2.3 Test du premier DNS

Nous pouvons tester le dns primaire, en prenant une machine test que j'ai crée exprès pour l'occasion. Je me suis mis dans le même réseau que le DNS primaire et quand j'effectue la commande `nslookup ns1.ucexchange.com` j'ai bien le retour attendu :

```
root@test-client-stras:~# nslookup ns1.ucexchange.com
Server:          10.242.220.2
Address:         10.242.220.2#53

Name:   ns1.ucexchange.com
Address: 10.242.220.2
```

FIGURE 5 – Retour de la commande `nslookup ns1.ucexchange.com`

Nous constatons que je récupère bien l'adresse IP du DNS primaire avec son nom associé.

2.3 Le DNS secondaire

2.3.1 Zone de résolution directe

Le but du DNS secondaire et qu'il hérite du premier DNS. Donc j'ai laissé une configuration assez minimale pour le DNS secondaire qui est `ns2.ucexchange.com`. J'ai donc redéclaré la zone exactement de la même manière que pour le premier DNS :

```
zone "ns2.ucexchange.com" {
    type master;
    file "/etc/bind/db.ns2.ucexchange.com";
};
```

FIGURE 6 – Déclaration de la zone de résolution directe

Pour la zone de résolution directe du deuxième DNS, j'ai laissé par défaut les valeurs, car comme je l'ai expliqué plutôt, le DNS secondaire hérite du DNS primaire :

```
; BIND reverse data file for empty rfc1918 zone
;
; DO NOT EDIT THIS FILE - it is used for multiple zones.
; Instead, copy it, edit named.conf, and use that copy.
;
$TTL      86400
@         IN      SOA      ns2.ucexchange.com. root.localhost. (
                                1           ; Serial
                                604800      ; Refresh
                                86400       ; Retry
                                2419200    ; Expire
                                86400      ) ; Negative Cache TTL
;
;         IN      A        10.242.220.3
@         IN      NS       localhost.
```

FIGURE 7 – Configuration du DNS secondaire

L'explication des lignes de configuration du premier DNS est toujours valable pour le DNS secondaire car c'est la même chose.

2.3.2 Zone de résolution inverse

Pour la zone de résolution inverse du DNS secondaire, j'ai repris exactement les mêmes paramètres que pour le premier DNS car ici, le but n'était pas de faire une résolution inverse très compliqué.

2.3.3 Test du DNS secondaire

Comme pour le DNS primaire nous pouvons effectuer la commande `nslookup ns2.ucexchange.com` pour voir si le DNS secondaire est bien en place. Voici le résultat :

```
root@test-client-metz:~# nslookup ns2.ucexchange.com
Server:      10.242.220.3
Address:     10.242.220.3#53

Name:   ns2.ucexchange.com
Address: 10.242.220.3
```

FIGURE 8 – Test du DNS secondaire

3 le DHCP

3.1 Explication du DHCP

Un serveur DHCP (Dynamic Host Configuration Protocol) est utilisé pour attribuer des adresses IP à des ordinateurs et d'autres dispositifs sur un réseau. Il joue un rôle clé dans la configuration automatique des paramètres réseau pour les ordinateurs et autres dispositifs connectés au réseau.

Lorsqu'un ordinateur ou un autre dispositif se connecte à un réseau, il envoie une requête DHCP pour demander une adresse IP. Le serveur DHCP répond en attribuant une adresse IP disponible à ce dispositif, ainsi que d'autres paramètres de configuration réseau tels que le masque de sous-réseau, la passerelle par défaut et les serveurs DNS.

L'utilisation d'un serveur DHCP simplifie la gestion des adresses IP sur un réseau car il automatise la distribution des adresses IP, ce qui évite les erreurs de configuration manuelle et rend plus facile le suivi de l'utilisation des adresses IP sur le réseau.

En plus de cela, un serveur DHCP permet de configurer de manière centralisée les paramètres réseau pour les dispositifs sur le réseau. Cela signifie que vous pouvez facilement apporter des modifications aux paramètres réseau pour de nombreux dispositifs en une seule étape, plutôt que de devoir les configurer manuellement sur chaque dispositif individuellement.

J'ai choisi d'utiliser le logiciel `isc-dhcp-server`, encore une fois présent sur Linux. Pour pouvoir le configurer il faut l'installer via la commande suivante :
`apt-get install isc-dhcp-server`

4 Mise en place du DHCP

Une fois installé la configuration est assez simple, il suffit de déclarer plusieurs paramètres pour que le serveur DHCP fonctionne. Par exemple, le serveur DNS, les adresses de gateway, les adresses IP à attribuer, etc.

Voici la configuration que j'ai pu faire, à chaque fois en déclarant des plages d'adresses IP différentes pour les différents Vlans de chaque sites (Strasbourg et Metz).

```
default-lease-time 600;
max-lease-time 7200;
subnet 10.242.210.0 netmask 255.255.255.0 {
    option subnet-mask 255.255.255.0;
    option routers 10.242.210.254;
    range 10.242.210.10 10.242.210.253;
}

subnet 10.242.220.0 netmask 255.255.255.0 {
    option subnet-mask 255.255.255.0;
    option routers 10.242.220.254;
    range 10.242.220.10 10.242.220.253;
}

subnet 10.242.230.0 netmask 255.255.255.0 {
    option subnet-mask 255.255.255.0;
    option routers 10.242.230.254;
    range 10.242.230.10 10.242.230.253;
}

subnet 10.242.110.0 netmask 255.255.255.0 {
    option subnet-mask 255.255.255.0;
    option routers 10.242.110.254;
    range 10.242.110.10 10.242.110.253;
}

subnet 10.242.120.0 netmask 255.255.255.0 {
    option subnet-mask 255.255.255.0;
    option routers 10.242.120.254;
    range 10.242.120.10 10.242.120.253;
}

subnet 10.242.130.0 netmask 255.255.255.0 {
    option subnet-mask 255.255.255.0;
    option routers 10.242.130.254;
    range 10.242.130.10 10.242.130.253;
}
```

FIGURE 9 – Configuration du serveur DHCP

Nous y voyons donc plusieurs paramètres. Voici quelques explications :

- `default-lease-time` : permet de déclarer le temps d'attribution d'une IP
- `max-lease-time` : permet de déclarer le temps maximum d'attribution d'une IP
- `option routers` : permet de déclarer les adresses de gateway
- `option subnet-mask` : permet de déclarer le masque de sous-réseau
- `range` : permet de déclarer les plages d'adresses IP à attribuer

4.0.1 test du serveur DHCP

Pour pouvoir vérifier que mon serveur DHCP fonctionne, j'ai créé des VM en utilisant le réseau du serveur DHCP, et nous pouvons constater que mes VM obtiennent bien une adresse IP automatiquement :

```
root@test-client-stras:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0@if67: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether fe:55:6f:fc:90:28 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.242.220.13/24 brd 10.242.220.255 scope global dynamic eth0
        valid_lft 317sec preferred_lft 317sec
    inet6 fe80::fc55:6fff:fefc:9028/64 scope link
        valid_lft forever preferred_lft forever
```

FIGURE 10 – Test du serveur DHCP

Nous avons bien une adresse IP en 10.242.220.XX

5 Le serveur WEB

Pour la mise en place du serveur Web j'ai tout simplement utilisé Apache2. Pour l'installer il suffit d'utiliser la commande suivante `apt-get install apache2`. Une fois installé, j'ai laissé par défaut la page car le but était ici de pouvoir effectuer une requête `ping`. Mais aussi, le but était de pouvoir atteindre le serveur web avec le DNS de l'entreprise. Donc nous pouvons faire un `ping` vers le nom de domaine du serveur et nous avons bien une réponse comme le démontre la capture d'écran ci-dessous :

```
root@test-client-metz:~# ping www.ns2.ucexchange.com
PING www.ns2.ucexchange.com (10.242.220.3) 56(84) bytes of data.
64 bytes from 10.242.220.3 (10.242.220.3): icmp_seq=1 ttl=64 time=0.021 ms
64 bytes from 10.242.220.3 (10.242.220.3): icmp_seq=2 ttl=64 time=0.069 ms
64 bytes from 10.242.220.3 (10.242.220.3): icmp_seq=3 ttl=64 time=0.079 ms
64 bytes from 10.242.220.3 (10.242.220.3): icmp_seq=4 ttl=64 time=0.065 ms
^C
--- www.ns2.ucexchange.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3065ms
rtt min/avg/max/mdev = 0.021/0.058/0.079/0.023 ms
```

FIGURE 11 – Ping vers le serveur Web

Nous avons bien une réponse suite aux ping vers le serveur. J'ai choisi d'héberger mon serveur Apache sur la même VM que la où est héberger le DNS secondaire. J'ai donc utilisé le DNS secondaire pour le nom de domaine. C'est pourquoi nous avons en retour l'adresse IP de mon DNS secondaire et comme nom de domaine mon deuxième DNS.

6 Conclusion

En conclusion, le serveur DHCP, le serveur Web Apache et la configuration de DNS avec Bind9 sont des composants importants d'un réseau informatique. Le serveur DHCP simplifie la gestion des adresses IP sur un réseau en attribuant automatiquement des adresses IP aux dispositifs connectés. Le serveur Web Apache permet de publier et de gérer des sites Web tandis que la configuration de DNS avec Bind9 garantit la résolution efficace des noms de domaines en adresses IP.

Cette SAE m'a permis d'apprendre pas mal de choses sur la gestion de DNS et un serveur DHCP dont je ne connaissais pas l'existence. J'ai pu aussi découvrir comment le cœur d'un réseau informatique fonctionnait au sein d'une entreprise présente sur plusieurs sites. Avec mes camarades nous avons même réussi à interconnecter les 3 parties, pour que la partie LAN puisse prendre des adresses IP depuis mon serveur DHCP, et pour que la partie FAI puisse faire communiquer le site de Strasbourg avec le site de Metz.