Domain Name System

Table des matières

I - Les différents types de serveurs DNS **3**

1. Serveur primaire 3
2. Serveur récursif 4
3. Serveur cache 5

II - Mise en place d'un DNS avec bind9 **6**

1. Mise en place d'un serveur DNS local minimaliste avec BIND9 6

III - Annexes **8**

1. Le fichier /etc/hosts 8
2. Exercice : Lecture du /etc/hosts 8

Solutions des exercices **9**

Crédits des ressources **10**

Contenus annexes **11**

I Les différents types de serveurs DNS

1. Serveur primaire

Un serveur primaire est un serveur qui fait autorité sur sa sa zone. Il s'agit d'un serveur contenant la correspondance entre adresses IP et noms de domaine pour une zone donnée. Il ne renverra donc sur aucun autre serveur s'il reçoit une requête.

Cependant, s'il reçoit une requête ne concernant pas sa zone, il ne saura pas y répondre.

Pour chaque domaine, il ne peut exister qu'un unique serveur primaire.

Quelques tests pour comprendre

On va faire quelques tests avec la commande `nslookup` et le serveur DNS de wikipedia qui fait évidemment autorité sur la zone "wikipedia.org".

Pour faire appel au DNS de wikipedia, on utilisera l'adresse suivante : **ns0.wikimedia.org**

Premièrement, faisons une requête au DNS de la FDN (80.67.169.12) pour le nom de domaine wikipedia.org :

[Essayez chez vous !]

```
nslookup wikipedia.org 80.67.169.12

#Reponse:
Server:    80.67.169.12
Address:  80.67.169.12#53

Non-authoritative answer:
Name:  wikipedia.org
Address: 91.198.174.192
Name:  wikipedia.org
Address: 2620:0:862:ed1a::1
```

Le serveur n'a pas autorité et il l'affiche dans sa réponse. On comprend que ce DNS n'est pas primaire pour la zone wikipedia.org.

Testons maintenant le serveur DNS de wikipedia :

```
nslookup wikipedia.org ns0.wikimedia.org

#Reponse:
Server:    ns0.wikimedia.org
Address:  208.80.154.238#53

Name:  wikipedia.org
Address: 91.198.174.192
Name:  wikipedia.org
Address: 2620:0:862:ed1a::1
```

Cette fois-ci le DNS ne répond pas qu'il ne fait pas autorité donc on peut en conclure qu'il est bien primaire pour la zone wikipedia.org.

Essayons un dernier test : demandons au DNS de wikipedia l'adresse correspondant au nom de domaine icann.org :

```
nslookup icann.org ns0.wikimedia.org

#Réponse:
Server:    ns0.wikimedia.org
Address:  208.80.154.238#53
```

```
6
7** server can't find icann.org: REFUSED
8
```

Le serveur étant primaire, il ne peut répondre que pour sa propre zone et ne peut donc répondre positivement aux requêtes concernant d'autres zones.

Serveur secondaire

Pour chaque serveur primaire, il est conseillé d'avoir au moins un serveur secondaire qui prendra le relais en cas de panne du serveur primaire. Il contient donc les mêmes informations que le serveur primaire.

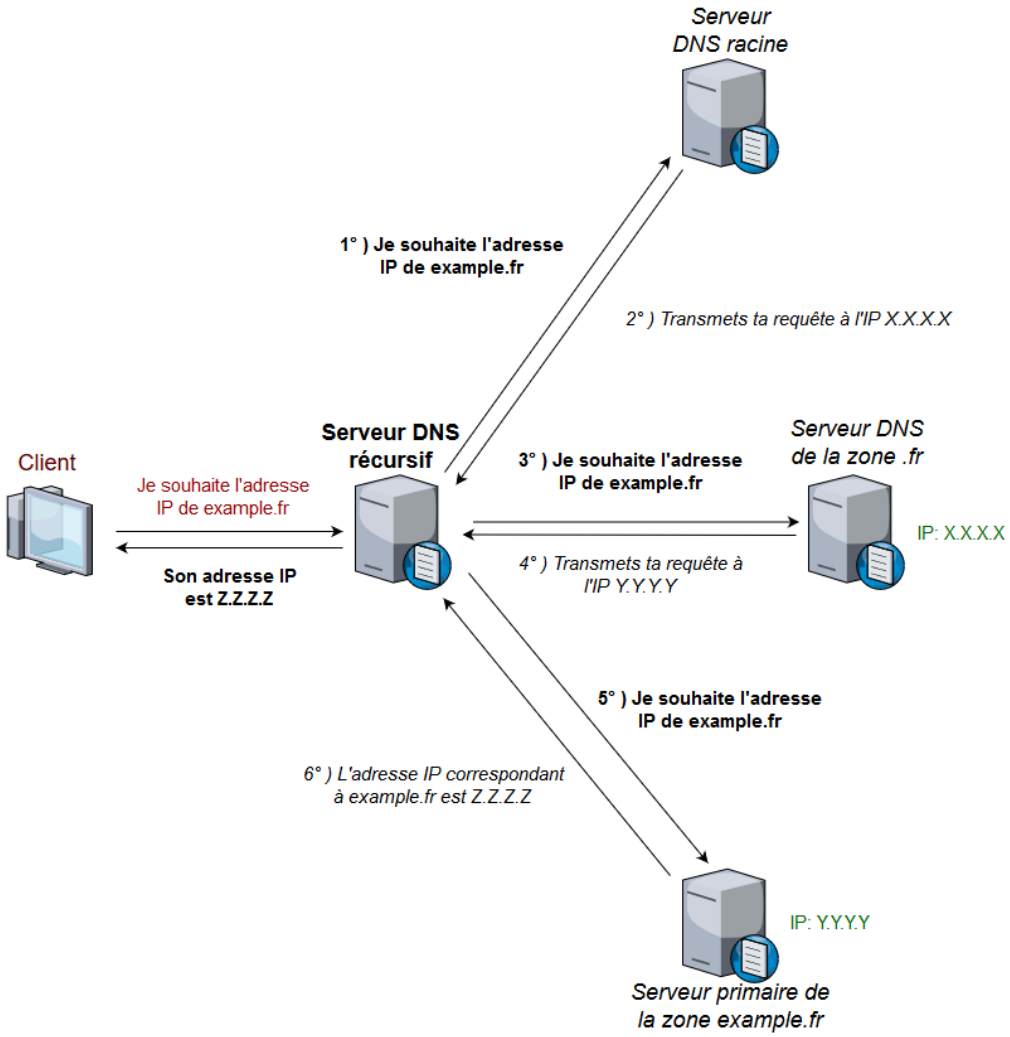
2. Serveur récursif

Ce type de serveur est celui auquel on adresse le plus souvent nos requêtes en premier lieu. Il ne contient pas de réponse mais sait comment en trouver une à coup sûr si le nom de domaine que vous lui entrez a bien une IP correspondante.

Son fonctionnement est assez simple :

- il va faire remonter votre requête à la racine
- la racine lui transmettra l'IP du serveur gérant le domaine de premier niveau du nom de domaine de la requête
- Le serveur récursif va transmettre la requête à ce serveur.
- Ce serveur vous répondra soit l'IP que vous cherchez, soit l'IP du serveur gérant le domaine de second niveau
- Et ainsi de suite jusqu'à trouver le serveur primaire contenant la réponse à votre requête.

Voici un petit schéma modélisant le trajet d'une requête :



Un système d'arbre offrant une efficacité dans la récursivité

On voit ici que toute la hiérarchisation en domaines et sous-domaines prend tout son sens dans la recherche récursive. Celle-ci permet d'assurer un résultat certain et en un temps minimum grâce à l'arborescence du système.

3. Serveur cache

Ce type de serveur est assez simple et se combine en général avec le serveur récursif. Il va simplement conserver pendant un certain temps les réponses données par le serveur auquel il est lié. Ensuite, à chaque réponse qui sera reçue, on consultera d'abord les réponses qu'il a en mémoire avant de demander au serveur récursif.

Cette solution est notamment utilisé sur des DNS recevant un grand nombre de requêtes comme ceux des fournisseurs d'accès internet (F.A.I.).

II Mise en place d'un DNS avec bind9

1. Introduction

BIND (*Berkeley Internet Name Daemon*) est utilisé par 80% des DNS actuellement. Il a été développé par 4 étudiants de l'Université de Berkeley dans les années 80.

BIND9 est la dernière version qui avait été réécrite totalement pour supprimer les importantes failles de sécurité des versions précédentes.

On va installer BIND9 pour déployer un serveur DNS sur notre propre machine.

Pour installer le paquet, il suffit d'entrer `sudo apt-get install bind9` (on conseillera de faire un `sudo apt-get update`)

2. Mise en place d'un serveur DNS local minimaliste avec BIND9

Dans cette partie, on va configurer un serveur local et minimal pour faire de simples manipulations et mieux comprendre le fichier de zone.

Première étape : installation du paquet

```
sudo apt-get install bind9
```

Deuxième étape : modification du fichier

On va éditer le fichier /etc/bind/named.conf.local et y entrer les lignes suivantes :

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

On définit tout d'abord la zone qui est donc exemple.com.

Puis, on dit de le serveur est primaire ("master"). Le type des serveurs secondaires sera "slave".

Enfin, on donne l'emplacement du fichier de zone.

Troisième étape : édition du fichier de zone

On va éditer /etc/bind/db.exemple.com et y entrer les lignes suivantes :

```
$TTL    10800
@       IN      SOA     ns1.exemple.com. root.exemple.com. (
                             1         ; Serial
                         10800         ; Refresh
                         86400         ; Retry
                       2419200         ; Expire
                        604800 )       ; Negative Cache TTL
;
@       IN      NS      ns1.exemple.com.
ns1     IN      A       3.3.3.3
toto    IN      A       4.4.4.4
tata    IN      A       5.5.5.5
```

Pour comprendre un peu mieux ces entrées, je vous invite à lire ce grain si ce n'est pas déjà fait : *Les enregistrements* (cf. p.11)

Quatrième étape : redémarrer le serveur

Pour redémarrer notre DNS, on utilisera la commande suivante :

```
1 sudo /etc/init.d/bind9 restart
```

III Annexes

1. Le fichier /etc/hosts

Ce fichier, existant depuis le début des années 80, a pour rôle d'associer une adresse IP à un nom d'hôte (un nom de domaine n'étant qu'une forme structurée de nom d'hôte). Il est donc présent dans le répertoire etc et est consulté par votre système à chaque fois que vous tenter d'accéder à un nom d'hôte spécifique.

Notons qu'il est consulté avant qu'une requête soit envoyée à un DNS. Ce fichier n'a de valeur que sur notre propre machine. Aucune autre machine de notre réseau (même local) ne prendra en compte ce qui y est inscrit.

De /etc/hosts au DNS

A l'origine, on se partageait le fichier après l'avoir mis à jour mais très vite le fichier devint trop volumineux et il fallut trouver une solution : le Domain Name System. Il permet donc de centraliser le système de noms d'hôtes. Cependant, le système nécessite une structure (permettant une hiérarchisation des noms) qui fut trouvée dans les noms de domaines.

L'utilisation actuelle du fichier

Ce fichier existe toujours et son utilisation est restreinte à de petits réseaux locaux ainsi qu'à quelques autres cas particuliers.

Sa structure

La structure du fichier est des plus simples : sur chaque ligne, on a une IP et juste à coté le nom d'hôte qui lui correspond. Voici un exemple :

```
1 X.X.X.X monsite.fr
2 Y.Y.Y.Y test
```

Pour aller un peu plus loin ⊕ Complément

http://www.linux-france.org/~mdecore/linux/doc/memo2/node37.html

2. Exercice : Lecture du /etc/hosts

[solution n°1 p. 9]

Sur Linux, par défaut, une adresse est liée au nom d'hôte "localhost", quelle est-elle ?

Solutions des exercices

Solution n°1

[exercice p. 8]

Sur Linux, par défaut, une adresse est liée au nom d'hôte "localhost", quelle est-elle ?

127.0.0.1

On exécute un simple `sudo cat /etc/hosts` et la réponse nous saute aux yeux.

Cette IP est particulière car elle est, par convention, réservée pour renvoyer sur la machine sur laquelle vous êtes. On parle souvent de *localhost* ou de *loopback address*.

Crédits des ressources

p. 5

Attribution - Marc DAMIE

Contenus annexes

1. Les différents types d'enregistrement

On va donc passer en revue les principaux types d'enregistrement qu'on peut entrer dans un fichier de zone.

Notons que pour chacun des enregistrements on peut indiquer un TTL (*Time To Live*) personnalisé. Cela permettra par exemple de mettre des grands temps pour les noms les plus demandés de la zone. Ainsi, l'enregistrement restera longtemps en cache et le serveur recevra moins de requêtes pour ces noms de domaine.

Enregistrements A et AAAA

Ces enregistrements sont assez simples : on lie **un nom de domaine à une adresse IP** (v4 pour le type A et v6 pour le type AAAA).

Voici un exemple :

```
exemple.fr.  IN  A     192.0.2.1
             IN  AAAA  2001:db8:10::1
```

Typiquement, nous utiliserons ces enregistrement pour représenter une machine par exemple le VPS de Quentin ou la raspberry PI de Tobias.

Enregistrements CNAME

Ces enregistrements permettent de faire qu'un **nom de domaine pointe sur un autre**, des **alias**.

```
www          IN  CNAME exemple.fr.
wwwtest      IN  CNAME exemple.fr.
```

Il est souvent pratique d'utiliser ce système pour faire que plusieurs sous-domaines pointent sur une seule adresse. Ici, seul exemple est directement lié à une adresse IP. Si l'adresse IP vient à changer on aura pas à changer ce que pointe les sous-domaines.

Par exemple si le service de pad et celui de chat sont sur la machine vps01 on aura les enregistrements `pad IN CNAME vps01.exemple.fr.` et `chat IN CNAME vps01.exemple.fr.`.

Enregistrements NS

Ces enregistrements ont pour but de définir les **DNS de la zone**. Le premier de la liste étant le serveur primaire.

```
exemple.fr.  IN  NS    ns0
exemple.fr.  IN  NS    ns1
```

On a défini les adresses de manière relative, il résultera de ces lignes que le serveur primaire a pour nom de domaine **ns0.exemple.fr** et le secondaire aura **ns1.exemple.fr**.

Les enregistrements MX

Ces enregistrements permettent de **définir les adresses des serveurs mail**. Le nombre précédent l'adresse sert à définir la priorité (plus le nombre est grand, plus le serveur est prioritaire).

```
exemple.fr.  IN  MX    10 mail.exemple.fr.
@            IN  MX    20 mail2.exemple.fr.
```

Notons qu'avec ces lignes il faudra un enregistrement de type A pour dire les adresses liées à mail.exemple.fr et mail2.exemple.fr.

Quelques autres types un peu plus complexes

Il existe quelques autres types un peu plus complexes tels que SRV, TXT ou SPF.

Pour en savoir un peu plus, je vous redirige vers le site suivant : Wiki Gandi[1]

Commentaire d'un fichier de zone

Exemple

Nous avons repris ci-dessous une partie du **fichier de zone** du domaine **picasoft.net**. Nous allons le commenter.

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;; Configuration des serveurs de nom ;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
@             IN      NS      ns01.picasoft.net.      ; Alice
@             IN      NS      ns02.picasoft.net.      ; Bob

ns01          IN      A       91.224.148.84               ; Alice IPv4
ns01          IN      AAAA    2a03:7220:8080:5400::1      ; Alice IPv6
ns02          IN      A       91.224.148.85               ; Bob IPv4
ns02          IN      AAAA    2a03:7220:8080:5500::1      ; Bob IPv6

;;;;;;;;;;;;;;;
;; Machines ;;
;;;;;;;;;;;;;;;
pica01        IN      A       91.224.148.57
pica01        IN      AAAA    2a03:7220:8080:3900::1

pica02        IN      A       91.224.148.60
pica02        IN      AAAA    2a03:7220:8080:3c00::1

;;;;;;;;;;;;;;;
;; Services ;;
;;;;;;;;;;;;;;;
; Services sur pica01
www               IN      CNAME   pica01.picasoft.net.    ; Site web
week.pad          IN      CNAME   pica01.picasoft.net.    ; Etherpad week
doc               IN      CNAME   pica01.picasoft.net.    ; Serveur Web de
documents techniques

; Services sur pica02
pad               IN      CNAME   pica02.picasoft.net. ; Etherpad
team              IN      CNAME   pica02.picasoft.net. ; Mattermost
wiki              IN      CNAME   pica02.picasoft.net. ; Wiki
```

Nous commençons par indiquer les serveurs qui permettrons de gérer les requêtes DNS (dans la partie **configuration des serveurs de nom**), nous utilisons pour cela les enregistrements de type **NS**, le serveur primaire est ainsi `ns01.picasoft.net` et le secondaire `ns02.picasoft.net`. Il faut bien sûr mettre des enregistrements **A** (IPv4) et **AAAA** (IPv6) afin de savoir quelle adresse contacter lorsqu'on appelle `ns01.picasoft.net` ou `ns02.picasoft.net`.

Nous définissons ensuite nos machines (**pica01** et **pica02**) grâce à des champs **A** et **AAAA** afin de pouvoir associer une adresse IP à leur nom (dans la partie **machines**).

Enfin nous définissons des alias (enregistrements **CNAME**) afin de lier facilement les services à la machine qui les héberge. Ainsi si le service change de machine il suffit de changer l'alias et si une machine change d'IP il suffit de changer ses enregistrements **A** et **AAAA**, les services qui lui sont associés pointeront alors automatiquement sur la bonne IP.

1. https://wiki.gandi.net/fr/dns/zone