TP 5 DNS

TP RSD

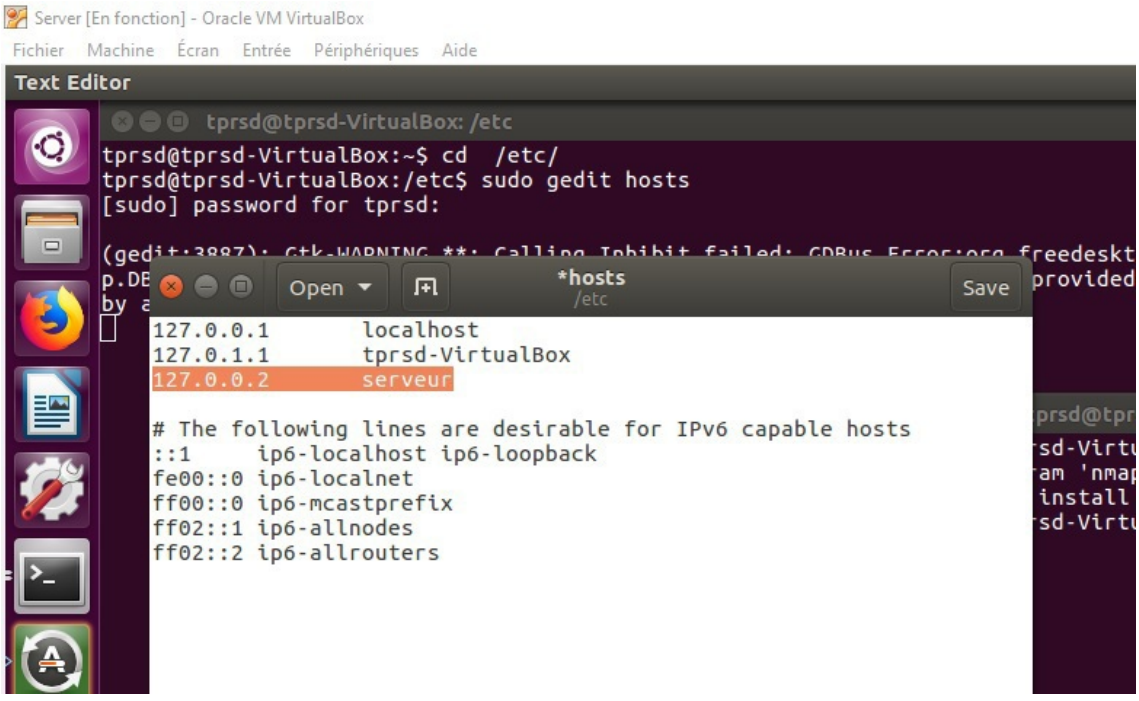
Par Bambrik Ilyas et Amraoui Asma

Table des matières

TP DNS

1. /etc/hosts

- Le fichier */etc/hosts* permet de donner des noms aux machines situées dans le réseau afin de les adresser par leurs noms et au lieu de leurs @IP.
- Ouvrez */etc/hosts* en mode administrateur :
  sudo gedit /etc/hosts
- Ajoutez une entrée pour une machine appelée *serveur* qui correspond à l'adresse IP 127.0.0.2 (localhost2) et sauvegardez le fichier hosts.



Testez par un ping la connexion à la machine *serveur* ou par l'accès à un site web hébergé dans votre machine.

```
tprsd@tprsd-VirtualBox:~$ ping serveur
PING serveur (127.0.0.2) 56(84) bytes of data.
64 bytes from serveur (127.0.0.2): icmp_seq=1 ttl=64 time=0.027 ms
64 bytes from serveur (127.0.0.2): icmp_seq=2 ttl=64 time=0.068 ms
64 bytes from serveur (127.0.0.2): icmp_seq=3 ttl=64 time=0.076 ms
64 bytes from serveur (127.0.0.2): icmp_seq=4 ttl=64 time=0.031 ms
64 bytes from serveur (127.0.0.2): icmp_seq=5 ttl=64 time=0.075 ms
64 bytes from serveur (127.0.0.2): icmp_seq=6 ttl=64 time=0.074 ms
64 bytes from serveur (127.0.0.2): icmp_seq=7 ttl=64 time=0.033 ms
64 bytes from serveur (127.0.0.2): icmp_seq=8 ttl=64 time=0.073 ms
64 bytes from serveur (127.0.0.2): icmp_seq=9 ttl=64 time=0.072 ms
64 bytes from serveur (127.0.0.2): icmp_seq=10 ttl=64 time=0.038 ms
64 bytes from serveur (127.0.0.2): icmp_seq=11 ttl=64 time=0.074 ms
64 bytes from serveur (127.0.0.2): icmp_seq=12 ttl=64 time=0.096 ms
64 bytes from serveur (127.0.0.2): icmp_seq=13 ttl=64 time=0.074 ms
64 bytes from serveur (127.0.0.2): icmp_seq=14 ttl=64 time=0.070 ms
^Z
[1]+  Stopped                 ping serveur
tprsd@tprsd-VirtualBox:~$
```

2. Serveur DNS cache

- Afin d'installer le serveur DNS BIND sur une machine (celui ci est déjà installé sur la machine virtuelle) exécutez la commande suivante :
  sudo apt-get update
  sudo apt-get install bind9 bind9utils bind9-doc
- Exécutez la commande suivante afin d'ouvrir le port 53 (DNS) dans le par-feu :
  sudo ufw allow 53/udp
- Pour commencer, copiez les fichiers de configuration suivants dans un repertoir de votre choix (*mkdir DNSFiles* dans cet exemple):
  mkdir DNSFiles
  cp /etc/resolv.conf /etc/bind/ /home/tprsd/DNSFiles*
- Ouvrez */etc/bind/named.conf.options* en tant qu'administrateur :
  sudo gedit /etc/bind/named.conf.options
- Afin de permettre notre serveur DNS BIND9 de répondre à des requêtes DNS en tant que serveur cache, il est nécessaire d'ajouter les commandes suivantes :


```
// pour ecouter les requetes DNS entrantes sur le port 53 sur les
// interfaces de notre machine
//localhost = 127.0.0.1 et interface de la machine virtuelle =10.0.2.15;
listen-on port 53 { 127.0.0.1; 10.0.2.15; };

// ################################

// autorise la resolution des requetes entrentes de toute machine
// appartenant aux réseaux 127.0.0.0/24 pou 10.0.2.0/8
allow-query {  127.0.0.0/24; 10.0.2.0/8; };

// ################################
// defini 8.8.8.8 et 8.8.4.4 (Google DNS server) comme
// forwarders des requetes que la machine locale ne peut pas
// resourde localement (cache)
forwarders { 8.8.8.8; 8.8.4.4; };

// ################################
// execution de requete DNS recursive
recursion yes;

```

3. DNS pour zone locale

- Modifiez */etc/bind/named.conf.default-zones* et ajoutez à la fin du fichier la définition d'une zone pour un domaine *exemple.com* :

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

// definition d'un fichier reverse-lookup our la zone (PTR)
zone    "10.0.2.in-addr.arpa" IN {
       type master;
       file "/etc/bind/db.10.0.2";
```

```
};
```

- Créez le fichier de zone */etc/bind/db.exemple.com* :

```
$TTL 604800
@  IN SOA  server.exemple.com.   root.server.exemple.com. (
        2019061501    ; Serial
        604800    ; Refresh
        86400    ; Retry
        2419200    ; Expire
         604800 )  ; Negative Cache TTL

;
@               IN      NS      server.exemple.com.
;
server          IN      A       127.0.0.1
@                IN      A        127.0.0.1
;
```

- *@* est équivalent au nom du domaine (*exemple.com*)
- *@ IN SOA server.exemple.com. root.server.exemple.com.* : @ (référence de *exemple.com*) est représenté pas le serveur DNS *"server.exemple.com."* . root.server.exemple.com. (root@ server.exemple.com) représente l'adresse mail de l'administrateur de cette zone.
- *2019061501 ; Serial* : Version du fichier de la zone (*2019-06-15 num°01*) ;
- *604800 ; Refresh* : Temps après le quel les serveurs DNS secondaires doivent vérifier s'il y a eu un changement dans le fichier de la zone ;
- *86400 ; Retry :* Temps après le quel les serveurs DNS secondaires doivent réessayer si la vérification échoue ;
- *2419200 ; Expire* : Temps après le quel l'information de cette zone sera non valide si aucune opération de vérification n'aboutisse ;
  604800) ; Negative Cache TTL : Duré de cache des réponses négative ;
- *@ IN NS server.exemple.com.* : L'enregistrement NS (NAME SERVER) du domaine *exemple.com.* pointe vers la machine *server.exemple.com.* .
- *server IN A 127.0.0.1* : la machine server (*server.exemple.com.*) se situe dans l'adresse *127.0.0.1*.
- *exemple.com. IN A 127.0.0.1* : la machine *exemple.com.* est située à l'adresse *127.0.0.1*.

Créez le fichier */etc/bind/db.10.0.2* et ajoutez le contenu suivant :

```
$TTL 604800
@  IN SOA  server.exemple.com.   root.server.exemple.com. (
              2019061501    ; Serial
         604800    ; Refresh
          86400    ; Retry
        2419200    ; Expire
         604800 )  ; Negative Cache TTL

@               IN      NS      server.exemple.com.
15              IN      PTR     server.exemple.com.
15              IN      PTR     exemple.com.
```

Le fichier contient la correspondance entre noms de machines dans le domaine et leurs adresses IP

(enregistrement PTR) :

- *15 IN PTR server.exemple.com. :* la machine 15 (10.0.2.*15*) correspond au nom *server.exemple.com.*
- Redémarrez le serveur BIND9 :
  sudo service bind9 restart

Pour tester le serveur DNS, tapez la commande suivante sur le terminale:

```
nslookup server.exemple.com 10.0.2.15
```

Cette commande permet d'interroger le serveur DNS 10.0.2.15 sur le nom server.exemple.com. Le résultat doit être l'adresse IP affectée à la machine server dans le fichier zone db.exemple.com.

Remarque : nslookup

Dans la commande *nslookup*, si vous possédez une autre adresse IP, remplacez 10.0.2.15 par votre adresse.

4. Enregistrement CNAME et Wildcard

- Les enregistrement de type CNAME sont significatif d'alias. Dans ligne 15 du fichier de zone suivant (db.exemple.com) :

```
www IN CNAME server
```

www est déclaré comme alias de server.

- L'enregistrement AAAA décare l'adresse IPv6 d'une machine. Dans ligne 17 du fichier de zone suivant (db.exemple.com) :

PC1 IN AAAA ::1

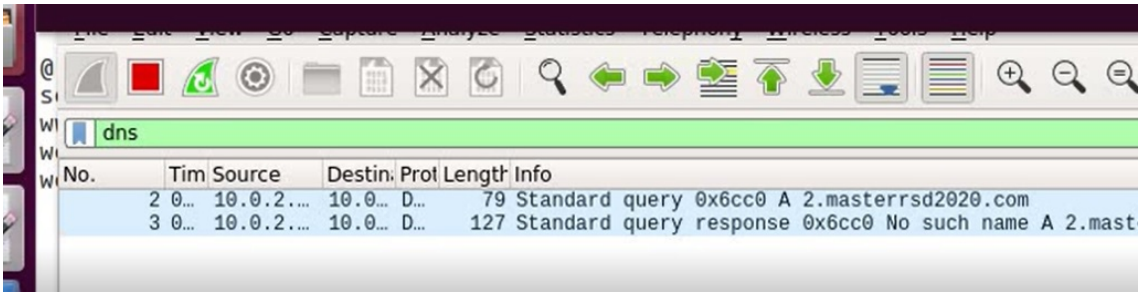
La machine PC1.exemple.com possède l'adresse IPv6 : :1.

```
1  $TTL 604800
2  @  IN SOA  server.exemple.com.   root.server.exemple.com. (
3          2019061501    ; Serial
4          604800    ; Refresh
5          86400    ; Retry
6          2419200    ; Expire
7           604800 )  ; Negative Cache TTL
8
9  ;
10 @               IN      NS      server.exemple.com.
11 ;
12 server          IN      A       127.0.0.1
13 @                IN      A        127.0.0.1
14 ;
15 www              IN      CNAME   server
16 ;
17 PC1              IN      AAAA    ::1
18 ;
```

Testez les commandes et capturez les résultats avec Wireshark (ajoutez le filtre dns ou udp.port==53 sur l'interface loopback) :

```
nslookup www.exemple.com 10.0.2.15
```

```
nslookup PC1.exemple.com 10.0.2.15
```



Si on teste la résolution pour le nom PC3.exemple.com, le résultat doit être une réponse négatif puisque aucune entrée dans *db.exemple.com* ne correspond à nom. Testez la commande suivante et lisez bien la réponse négatif du serveur:

```
nslookup PC3.exemple.com 10.0.2.15
```

```
1  $TTL 604800
2  @  IN SOA  server.exemple.com.   root.server.exemple.com. (
3          2019061501    ; Serial
4          604800    ; Refresh
5          86400    ; Retry
6          2419200    ; Expire
7           604800 )  ; Negative Cache TTL
8
9  ;
10 @               IN      NS      server.exemple.com.
11 ;
12 server          IN      A       127.0.0.1
13 @                IN      A        127.0.0.1
14 ;
15 www              IN      CNAME   server
16 ;
17 PC1              IN      AAAA    ::1
18 ;
19 *          IN      A       127.0.0.5
20 ;
```

Maintenant on ajoute l'enregistrement de type Wildcard (ligne 19) qui signifie que pour une requête qui correspond à un nom de machine inexistant dans le fichier de zone (db.exemple.com), celui ci correspond à l'adresse 127.0.0.5. L'entrée Wildcard se distingue par * au début de la ligne.

Testez les deux commandes suivantes (maintenant le serveur répondra avec la même adresse 127.0.0.5 pour les deux requêtes) :

```
nslookup PC3.exemple.com 10.0.2.15
```

```
nslookup Machine.exemple.com 10.0.2.15
```