TP Administration : Serveurs DHCP/DNS

TP Administration

L'objectif de cette activité est de réaliser la mise en œuvre des serveurs DHCP et DNS.

Les questions ne portent que sur la mise en œuvre de Dnsmasq.

Mise en situation

Vous devez disposer d'un PC possédant un système d'exploitation Linux ou Windows et du logiciel de virtualisation *VirtualBox*. Le système invité sera une installation du **serveur Ubuntu 18.04 LTS**.

Il est conseillé de configurer les serveurs avec une adresse IP statique.

Dnsmasq

Introduction

Dnsmasq est un serveur léger pour fournir les services **DNS**, **DHCP**, **BOOTP** et **TFTP** pour un petit réseau, voire pour un poste de travail.

Il permet d'offrir un service de nommage des machines d'un réseau local privé non intégrées au service DNS d'Internet.

Documentations : http://www.thekelleys.org.uk/dnsmasq/doc.html, pour les options de configuration http://www.thekelleys.org.uk/dnsmasq/docs/dnsmasq-man.html#lbAF et aussi https://doc.ubuntu-fr.org/configuration_serveur_dns_dhcp

Installation

Il suffit d'installer le paquet **dnsmasq** :

```
$ sudo apt-get install dnsmasq
```

Question 1. Installer `dnsmasq`.

Le service est contrôlé par **systemctl** :

```
$ systemctl [enable|disable|start|stop|restart|reload|status] dnsmasq
```

Vérifier l'état du serveur :

```
$ systemctl status dnsmasq
 dnsmasq.service - dnsmasq - A lightweight DHCP and caching DNS server
   Loaded: loaded (/lib/systemd/system/dnsmasq.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2020-02-02 08:47:58 UTC; 1min 22s ago
 Main PID: 1830 (dnsmasq)
    Tasks: 1 (limit: 1108)
```

```
   CGroup: /system.slice/dnsmasq.service
           |-1830 /usr/sbin/dnsmasq -x /run/dnsmasq/dnsmasq.pid -u dnsmasq -7 /etc/dnsmasq.d
               ,.dpkg-dist,.dpkg-old,.dpkg-new --local-service --trust-anchor=.,19036,8,2,49
               aac1

févr. 02 08:47:58 serveur systemd[1]: Starting dnsmasq - A lightweight DHCP and caching DNS
    server...
févr. 02 08:47:58 serveur dnsmasq[1808]: dnsmasq: syntax check OK.
févr. 02 08:47:58 serveur dnsmasq[1830]: started, version 2.79 cachesize 150
févr. 02 08:47:58 serveur dnsmasq[1830]: compile time options: IPv6 GNU-getopt DBus i18n IDN
     DHCP DHCPv6 no-Lua TFTP conntrack ipset auth DNSSEC loop-detect inotify
févr. 02 08:47:58 serveur dnsmasq[1830]: reading /etc/resolv.conf
févr. 02 08:47:58 serveur dnsmasq[1830]: using nameserver 127.0.0.53#53
févr. 02 08:47:58 serveur dnsmasq[1830]: read /etc/hosts - 10 addresses
févr. 02 08:47:58 serveur systemd[1]: Started dnsmasq - A lightweight DHCP and caching DNS
    server.
```

Afficher la version installée :

```
$ dnsmasq -v
Dnsmasq version 2.79 Copyright (c) 2000-2018 Simon Kelley
```

Question 2. Vérifier l'installation de **dnsmasq**.

Configuration

La configuration de **dnsmasq** est réalisée dans le fichier **/etc/dnsmasq.conf**.

Une modification de ce fichier nécessitera un redémarrage du service **dnsmasq**. Un contrôle de l'état du service est ensuite logiquement nécessaire.

```
$ sudo systemctl restart dnsmasq

$ sudo systemctl status dnsmasq

$ journalctl -xe --unit=dnsmasq

# ou en mode suivi :
$ journalctl -f -xe --unit=dnsmasq
```

Configuration DHCP

La configuration **DHCP** de **dnsmasq** est réalisée dans le fichier **/etc/dnsmasq.conf**.

Il est possible de spécifier l'interface d'écoute des requêtes **DHCP** (sinon toutes les interfaces seront utilisées) :

```
# remarque : l'interface de loopback est ajoutée automatiquement
#interface=eth0
```

Exemple pour des **adresses dynamiques** :

On définit la plage d'adresses dynamiques et la durée du bail l'option `dhcp-range` :

```
dhcp-range=192.168.0.100,192.168.0.150,12h
```

Exemple pour des adresses « **semi-statiques** » (fixes) :

On peut associer des adresses IP à des adresses MAC directement avec l'option `dhcp-host` :

```
# identifiée par l'adresse MAC
dhcp-host=11:22:33:44:55:66,192.168.0.66

# on peut ajouter en plus un nom et une durée de bail
dhcp-host=11:22:33:44:55:66,watson,192.168.0.66,45m
```

Il est possible aussi d'utiliser le fichier `/etc/ethers` en précisant alors l'option `read-ethers` dans `/etc/dnsmasq.conf` :

```
read-ethers
```

Exemple d'un `/etc/ethers` :

```
$ cat /etc/ethers
00:02:05:00:00:01 192.168.0.50
00:02:05:00:00:02 192.168.0.51
```

Ensuite, il est possible de définir le masque de réseau et la passerelle par défaut :

```
# le masque
dhcp-option=1,255.255.255.0
# ou en plus lisible :
dhcp-option=option:netmask,255.255.255.0

# la passerelle par défaut
dhcp-option=3,192.168.0.254
# ou en plus lisible :
dhcp-option=option:router,192.168.0.254
```

Il existe aussi les options `dhcp-option=option:ntp-server,X.X.X.X` et `dhcp-option=option:dns-server,Y.Y.Y.Y` ...

La liste des adresses qui seront attribuées par le serveur se trouve ici :

```
dhcp-leasefile=/var/lib/misc/dnsmasq.leases
```

```
$ cat /var/lib/misc/dnsmasq.leases
1611199040 00:01:6c:d1:25:3b 192.168.52.48 bts-sn 01:00:01:6c:d1:25:3b
```

Il est possible d'ajouter l'option `dhcp-authoritative` si **dnsmasq** est définitivement le seul serveur DHCP sur le réseau. L'option `log-dhcp` permet d'activer la journalisation qui se situera dans `/var/log/syslog`.

Il faut ensuite redémarrer le service `dnsmasq` :

```
$ sudo systemctl restart dnsmasq
```

```
$ systemctl status dnsmasq
```

On réalise quelques tests :

Les tests sont à réaliser côté **client** !

Problème potentiel : Il est possible que d'autres serveurs DHCP soient présents sur le réseau et répondent aux requêtes du client.

Solution n°1 :

Il sera probalement nécessaire de les ignorer en les indiquant dans le fichier `/etc/dhcp/dhclient.conf` avec l'option `reject`.

Sur le réseau de la salle de BTS SN, il y a 2 serveurs DHCP : `reject 192.168.52.85;` et `reject 192.168.52.42;`

```
$ sudo cat /etc/dhcp/dhclient.conf | grep -vE "^[# ]"

option rfc3442-classless-static-routes code 121 = array of unsigned integer 8;

send host-name = gethostname();
request subnet-mask, broadcast-address, time-offset, routers,
  domain-name, domain-name-servers, domain-search, host-name,
  dhcp6.name-servers, dhcp6.domain-search, dhcp6.fqdn, dhcp6.sntp-servers,
  netbios-name-servers, netbios-scope, interface-mtu,
  rfc3442-classless-static-routes, ntp-servers;

timeout 300;

# pour les tests :
reject 192.168.52.85;
reject 192.168.52.42;
```

Le **client DHCP** :

```
$ ifconfig
enp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
...

$ sudo dhclient -v enp2s0
Internet Systems Consortium DHCP Client 4.4.1
Copyright 2004-2018 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/enp2s0/00:01:6c:d1:25:3b
Sending on  LPF/enp2s0/00:01:6c:d1:25:3b
Sending on  Socket/fallback
DHCPDISCOVER on enp2s0 to 255.255.255.255 port 67 interval 3 (xid=0x2318cb15)
DHCPOFFER from 192.168.52.85 rejected by rule 192.168.52.85 mask 255.255.255.255.
DHCPOFFER of 192.168.52.48 from 192.168.52.204
DHCPREQUEST for 192.168.52.48 on enp2s0 to 255.255.255.255 port 67 (xid=0x15cb1823)
DHCPNACK from 192.168.52.85 rejected by rule 192.168.52.85 mask 255.255.255.255.
DHCPACK of 192.168.52.48 from 192.168.52.204 (xid=0x2318cb15)
bound to 192.168.52.48 -- renewal in 16596 seconds.
```

Solution n°2 :

Utiliser l'option `-s <adresse serveur>` du client DHCP pour sélectionner votre serveur DHCP :

Côté serveur, l'option `dhcp-authoritative` peut être ajoutée au fichier `/etc/dnsmasq.conf` pour déclarer que c'est le seul serveur DHCP sur le réseau.

```
$ ifconfig
enp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
...

$ sudo dhclient -v -s 192.168.52.204 enp2s0
Internet Systems Consortium DHCP Client 4.4.1
Copyright 2004-2018 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/enp2s0/00:01:6c:d1:25:3b
Sending on  LPF/enp2s0/00:01:6c:d1:25:3b
Sending on  Socket/fallback
DHCPDISCOVER on enp2s0 to 255.255.255.255 port 67 interval 3 (xid=0x2318cb15)
DHCPOFFER of 192.168.52.48 from 192.168.52.204
DHCPREQUEST for 192.168.52.48 on enp2s0 to 255.255.255.255 port 67 (xid=0x15cb1823)
DHCPACK of 192.168.52.48 from 192.168.52.204 (xid=0x2318cb15)
bound to 192.168.52.48 -- renewal in 16596 seconds.
```

L'allocation de l'adresse est consignée dans le fichier `dhclient.leases` :

```
$ cat /var/lib/dhcp/dhclient.leases
lease {
  interface "enp2s0";
  fixed-address 192.168.52.48;
  option subnet-mask 255.255.255.0;
  option routers 192.168.52.42;
  option dhcp-lease-time 86400;
  option dhcp-message-type 5;
  option domain-name-servers 8.8.8.8;
  option dhcp-server-identifier 192.168.52.42;
  renew 3 2021/01/20 21:32:52;
  rebind 4 2021/01/21 09:11:25;
  expire 4 2021/01/21 12:11:25;
}
```

Question 3. Configurer le serveur DHCP pour qu'il fournisse une plage d'adresse IP dynamiques (voir avec le professeur).

```
$ cat /etc/dnsmasq.conf | grep -vE "^[# ]"
```

Question 4. Tester.

Question 5. Configurer le serveur DHCP pour qu'il fournisse une adresse « semi-statique ».

```
$ cat /etc/dnsmasq.conf | grep -vE "^[# ]"
```

Question 6. Tester.

Il est possible de capturer l'échange des trames entre le serveur et le client.

Les numéros de port utilisés ici sont ceux utilisés par le protocole **BOOTP** (pris en charge par DHCP) : **67** pour le serveur et **68** pour le client en **UDP**.

```
$ cat /etc/services | grep -i "bootp" | grep -i "udp"
bootps    67/udp
bootpc    68/udp
```

Pour capturer les trames, on utilise l'outil `tcpdump` (en filtrant sur les numéros de port), ici côté serveur :

```
$ sudo tcpdump -tne -v -i enp0s3 port 67 or port 68
tcpdump: listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
00:01:6c:d1:25:3b > ff:ff:ff:ff:ff:ff, ethertype IPv4 (0x0800), length 342: (tos 0x10, ttl
    128, id 0, offset 0, flags [none], proto UDP (17), length 328)
    0.0.0.0.68 > 255.255.255.255.67: BOOTP/DHCP, Request from 00:01:6c:d1:25:3b, length 300,
         xid 0xba5f34a, Flags [none]
    Client-Ethernet-Address 00:01:6c:d1:25:3b
    Vendor-rfc1048 Extensions
      Magic Cookie 0x63825363
      DHCP-Message Option 53, length 1: Discover
      Hostname Option 12, length 6: "bts-sn"
      Parameter-Request Option 55, length 13:
        Subnet-Mask, BR, Time-Zone, Default-Gateway
        Domain-Name, Domain-Name-Server, Option 119, Hostname
        Netbios-Name-Server, Netbios-Scope, MTU, Classless-Static-Route
        NTP
...
```

Lien : http://tvaira.free.fr/bts-sn/reseaux/fiches/fiche-tcpdump.pdf

Question 7. Retracer les `DHCP-Message` échangés en précisant le sens entre le client et le serveur. Que sont les adresses `ff:ff:ff:ff:ff:ff` et `255.255.255.255` qui apparaissent dans les captures ?

Configuration DNS

dnsmasq met à disposition les entrées du fichier `/etc/hosts` en tant qu'entrées `DNS` (enregistrement `A` ou `AAAA`).

Tout d'abord, il faut s'assurer (côté serveur) que le service `systemd-resolved` n'est pas actif :

```
$ sudo systemctl status systemd-resolved

$ sudo systemctl disable systemd-resolved
$ sudo systemctl stop systemd-resolved

// et DNSStubListener=no
$ cat /etc/systemd/resolved.conf | grep DNSStubListener
#DNSStubListener=yes
```

La configuration DNS de **dnsmasq** est réalisée dans le fichier **/etc/dnsmasq.conf** :

```
port=53
# ne jamais transférer les requêtes A ou AAAA ne contenant pas un nom de domaine complet
domain-needed
# les recherches inversées pour des adresses IP privées (par exemple 192.168.x.x) qui ne se
    trouvent pas dans /etc/hosts ou DHCP recevront une réponse "pas de domaine" plutôt que d
    'être retransmises
bogus-priv
# filtre certaines requêtes périodiques émises par Windows
filterwin2k

# ajoute des domaines locaux ou privés
local=/bts-sn.lan/
# spécifie les domaines DNS pour le serveur DHCP
domain=bts-sn.lan
# dans l'ordre strict du fichier /etc/resolv.conf
#strict-order
# ajoute le nom de domaine à des noms simples dans /etc/hosts et les noms dérivés de DHCP
expand-hosts
```

Ensuite, il suffit d'éditer les noms dans le fichier `/etc/hosts` :

```
$ sudo vim /etc/hosts
192.168.0.1 serveur.bts-sn.lan
192.168.0.50 client1.bts-sn.lan
192.168.0.51 client2.bts-sn.lan
192.168.0.254 routeur.bts-sn.lan
```

Les serveurs DNS à interroger seront listés dans le fichier `/etc/resolv.conf` :

```
nameserver 127.0.0.1
nameserver 8.8.8.8
```

Voir aussi :

```
# nom de domaine local ou privé
domain bts-sn.lan
# domaine de recherche local
search bts-sn.lan
# serveur dns local
nameserver 127.0.0.1
# serveur dns primaire
nameserver 8.8.8.8
```

Les données DNS du serveur **dnsmasq** peuvent être stockées dans des fichiers séparés. Ces fichiers seront alors précisés dans **/etc/dnsmasq.conf** :

```
# Fichier des serveurs DNS
resolv-file=/etc/dnsmasq-dns.conf

# Fichier des enregistrements A et AAAA
addn-hosts=/etc/dnsmasq-hosts.conf
```

Il faut ensuite redémarrer le service `dnsmasq` :

```
$ sudo systemctl restart dnsmasq

$ systemctl status dnsmasq
```

On réalise quelques tests :

```
$ host client1.bts-sn.lan
client1.bts-sn.lan has address 192.168.52.61

$ host 192.168.52.61
61.52.168.192.in-addr.arpa domain name pointer client1.bts-sn.lan.

$ dig A client1.bts-sn.lan
; <<>> DiG 9.11.3-1ubuntu1.13-Ubuntu <<>> A client1.bts-sn.lan
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 58404
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; QUESTION SECTION:
;client1.bts-sn.lan.  IN A

;; ANSWER SECTION:
client1.bts-sn.lan. 0 IN A 192.168.52.61

;; Query time: 0 msec
;; SERVER: 192.168.52.60#53(192.168.52.60)
;; WHEN: Mon Feb 03 13:39:08 UTC 2020
;; MSG SIZE rcvd: 63

$ ping -c 1 client1.bts-sn.lan
PING client1.bts-sn.lan (192.168.52.61) 56(84) bytes of data.
64 bytes from client1.bts-sn.lan (192.168.52.61): icmp_seq=1 ttl=64 time=0.258 ms

--- client1.bts-sn.lan ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.258/0.258/0.258/0.000 ms
```

Question 8. Configurer le serveur DNS pour le domaine "`bts-sn.lan`". Il doit fournir un nom au client et les entrées suivantes :

```
192.168.52.XX www.bts-sn.lan
192.168.52.XX intra.bts-sn.lan
192.168.52.XX serveur.bts-sn.lan

192.168.52.85 bigbrother.bts-sn.lan
192.168.52.42 routeur.bts-sn.lan
```

Question 9. Tester.

Question 10. Réaliser une capture des trames avec l'outil `tcpdump` en filtrant uniquement le protocole DNS (le nom de ce service est `domain`).

Configuration TFTP

La configuration TFTP de `dnsmasq` est réalisée dans le fichier `/etc/dnsmasq.conf` :

```
enable-tftp
tftp-root=/var/ftpd
```

DHCP (*Dynamic Host Configuration Protocol*)

Introduction

DHCP (*Dynamic Host Configuration Protocol*) désigne un protocole réseau (RFC 1541 et 2131 essentiellement) dont le rôle est d'assurer la configuration automatique des paramètres IP d'une station, notamment en lui assignant automatiquement une adresse IP et un masque de sous-réseau pour une durée limitée (bail).

DHCP fonctionne sur le modèle client-serveur : un serveur (numéro de port 67), qui détient la politique d'attribution des configurations IP, envoie une configuration donnée pour une durée donnée à un client (numéro de port 68) donné (typiquement, une machine qui vient de démarrer). Le serveur va servir de base pour toutes les requêtes DHCP (il les reçoit et y répond), aussi doit-il avoir une configuration IP fixe. Le protocole DHCP s'appuie entièrement sur BOOTP en reprenant le mécanisme de base et le format des messages. DHCP est une extension de BOOTP.

Dans un réseau IP, on peut donc n'avoir qu'une seule machine avec adresse IP fixe : le serveur DHCP.

En résumé

- DHCP (*Dynamic Host Configuration Protocol*) permet d'automatiser la configuration TCP/IP des machines du réseau, quel que soit leur système d'exploitation.
- L'utilisation de DHCP simplifie l'administration système en regroupant en un seul point la configuration de tout un réseau (adresses dynamiques, semi-statiques, masque de sous-réseaux, DNS, passerelle par défaut, ...).
- Le service DHCP peut aussi servir à renvoyer la configuration des démarrages par réseau (serveur de *boot*, fichiers de démarrage réseau, ...).
- Il est nécessaire de configurer le serveur DHCP avec une adresse IP statique.

DHCP fonctionne avec IPv4 mais il fonctionne aussi avec IPv6 et il est alors appelé DHCPv6. Toutefois, en IPv6, les adresses peuvent être autoconfigurées sans DHCP.

Serveur `isc-dhcp-server`

L'*Internet Software Consortium* développe un serveur DHCP pour le monde du logiciel libre. C'est le serveur DHCP le plus répandu et celui qui respecte au mieux les RFCs. L'une des principales innovations de la version 3 est la possibilité de mettre à jour un DNS dynamiquement en fonction des adresses IP fournies par le serveur DHCP.

Il est distribué par le paquet **`isc-dhcp-server`**.

Installation

Il est nécessaire d'installer le paquet **`isc-dhcp-server`** car seuls les paquets clients DHCP sont installés par défaut :

```
$ sudo apt-get install isc-dhcp-server
```

Le service est contrôlé par `systemctl` :

```
$ systemctl [enable|diable|start|stop|restart|relaod|status] isc-dhcp-server
```

Vérifier l'état du serveur :

```
$ sudo systemctl status isc-dhcp-server
```

Configuration

Le fichier `/etc/dhcp/dhcpd.conf` contient la configuration du serveur. Il est composé de plusieurs sections, chacune limitée par des accolades.

```
subnet 192.168.0.0 netmask 255.255.255.0
{
    default-lease-time 28800;
    max-lease-time 86400;
    range 192.168.0.90 192.168.0.248;
    option routers 192.168.0.254;
    option domain-name-servers 192.168.0.254, 8.8.8.8;
}
```

Exemple pour des adresses dynamiques :

```
subnet 10.0.0.0 netmask 255.0.0.0
{
  option broadcast-address 10.255.255.255; # adresse de diffusion
  range 10.0.0.100 10.0.0.250; # plage d'adresses dynamiques
}
```

Exemple pour des adresses semi-statiques :

```
deny unknown-clients; # rejete les clients inconnus
subnet 10.0.0.0 netmask 255.0.0.0
{
  option broadcast-address 10.255.255.255; # adresse de diffusion
  host machine1
  {
    hardware ethernet 01:01:02:ae:34:c4; # adresse MAC
    fixed-address 10.0.0.2; # adresse IP fixe
  }
}
```

L'utilisation de DHCP permet de fournir une configuration complète de tout un réseau (adresses dynamiques, semi-statiques, masque de sous-réseaux, DNS, passerelle par défaut, ...). Quelques options :

- **`option routers 10.0.0.1;`** pour indiquer la passerelle par défaut (elle sera ajoutée à la table de routage)
- **`option domain-name-servers 10.0.0.1, 10.0.0.6;`** pour fournir les serveurs DNS (ils seront ajoutés dans `/etc/resolv.conf`)
- **`option domain-name "intra.net";`** pour fournir les domaines de recherche (ils seront ajoutés dans `/etc/resolv.conf`)

Documentation : https://doc.ubuntu-fr.org/isc-dhcp-server

DNS (*Domain Name System*)

Introduction

Le DNS (*Domain Name System* ou système de noms de domaine) est un service permettant de traduire un nom de domaine en adresses IP de la machine portant ce nom (RFC 882/883 en 1983).

DNS utilise le protocole de transport UDP et le port 53. La taille maximale des paquets utilisée est de 512 octets.

Le type d'enregistrement de ressource RR (*Resource Record*) est codé sur 16 bits. Les principaux enregistrements définis sont les suivants :

— **A record** (*Address record*) qui fait correspondre un nom d'hôte à une adresse IPv4 de 32 bits distribués sur quatre octets. **AAAA** pour IPv6.
— **CNAME record** (*Canonical Name record*) qui permet de faire d'un domaine un alias vers un autre. Cet alias hérite de tous les sous-domaines de l'original.
— **PTR record** (*PoinTer Record*) qui associe une adresse IP à un enregistrement de nom de domaine (aussi dit « *reverse* » car il fait le contraire du A record ou AAAA).

Le système des noms de domaines consiste en une hiérarchie dont le sommet est appelé la racine (représentée par un point `.`). Dans un domaine, on peut créer un ou plusieurs sous-domaines ainsi qu'une délégation pour ceux-ci (les informations relatives à ce sous-domaine sont enregistrées sur un autre serveur).

Les domaines se trouvant immédiatement sous la racine sont appelés domaine de premier niveau (TLD : *Top Level Domain*). Les noms de domaines ne correspondant pas à une extension de pays sont appelés des domaines génériques (gTLD), par exemple `.org` ou `.com`. S'ils correspondent à des codes de pays (fr, be, ch...), on les appelle ccTLD (*country code TLD*).

On entend par **FQDN** (*Fully qualified domain name*) ou Nom de domaine pleinement qualifié un nom de domaine écrit de façon absolue, y compris tous les domaines jusqu'au domaine de premier niveau (TLD), il est ponctué par un point final. Dans une réseau TCP/IP, une adresse FQDN sera l'association entre le nom de la machine et le domaine auquel elle appartient.

La norme prévoit qu'un élément d'un nom de domaine (appelé label) ne peut dépasser 63 caractères, un FQDN ne pouvant dépasser 255 caractères.

Les hôtes n'ont qu'une connaissance limitée du système des noms de domaine. Quand ils doivent résoudre un nom, ils s'adressent à un ou plusieurs serveurs de noms dits **récursifs**, c'est-à-dire qui vont parcourir la hiérarchie DNS et faire suivre la requête à un ou plusieurs autres serveurs de noms pour fournir une réponse.

Quand un serveur DNS récursif doit trouver l'adresse IP de `www.lasalle84.org`, un processus itératif démarre pour consulter la hiérarchie DNS. Ce serveur demande aux serveurs DNS appelés **serveurs racine** quels serveurs peuvent lui répondre pour la zone `org`. Parmi ceux-ci, notre serveur va en choisir un pour savoir quels serveurs sont capables de lui répondre pour la zone `lasalle84.org`. C'est un de ces derniers qui pourra lui donner l'adresse IP de `www.lasalle84.org`. S'il se trouve qu'un serveur ne répond pas, un autre serveur de la liste sera consulté.

Les serveurs récursifs fournissent des réponses qui ne sont pas nécessairement à jour, à cause du **cache** mis en place. On parle alors de réponse ne faisant pas autorité (*non-authoritative answer*). L'ensemble des

serveurs primaires et secondaires font **autorité** pour un domaine, c'est-à-dire que la réponse ne fait pas appel à un autre serveur ou à un cache.

Les **serveurs racine** sont gérés par douze organisations différentes (2 européennes, 1 japonaise et 9 américaines). Sept de ces serveurs sont en réalité distribués dans le monde grâce à la technique *anycast* (plus de 200 serveurs répartis dans 50 pays du monde) et neuf disposent d'une adresse IPv6. Il existe 13 autorités de nom appelées de `a` à `m.root-servers.net`. Le serveur `k` reçoit par exemple de l'ordre de 20 000 requêtes par seconde.

En résumé

— Le service DNS (*Domain Name System*) est utilisé pour associer les adresses IP aux noms complets (**FQDN**, *Fully Qualified Domain Name*) des machines (et inversement).
— Le DNS est une base de données distribuée, chaque domaine et sous domaine (appelés **zones**) étant gérés par un serveur DNS différent (un serveur peut gérer plusieurs zones). De plus, les serveurs sont organisés entre eux de façon hiérarchique.
— Une **zone** est gérée par un et un seul serveur DNS principal, et peut être répliquée sur un ou plusieurs serveurs secondaires.
— Chaque serveur DNS contient, pour chaque zone qu'il gère, les fichiers de la base de données permettant de convertir un nom de machine en adresse IP et inversement, ainsi que les noms et adresses des autres serveurs DNS de la zone et des serveurs de mail.

Installation

Le serveur de nom fourni avec Ubuntu Linux est **BIND** (*Berkeley Internet Name Domain*). **BIND** est composé, entre autre, du démon **`/usr/sbin/named`** et de la commande **`/usr/sbin/rndc`**.

Il suffit d'installer le paquet `bind9` :

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

Le paquet **dnsutils** fournit des outils très pratiques pour tester le service DNS : `sudo apt-get install dnsutils`

Vérifier l'état du serveur :

```
$ sudo systemctl status bind9
```

Configuration

Les fichiers de configuration de `bind9` sont stockés dans `/etc/bind/`.

Il lit sa configuration dans le fichier **`/etc/bind/named.conf`**, et stocke ses informations dans le répertoire **`/var/cache/bind/`**.

```
zone "xxx.esimed" {
    type master;
    file "/etc/bind/db.esimed.xxx";
};
```

```
zone "0.168.192.in-addr-arpa" {
    type master;
    file "/etc/bind/db.esimed.xxx.rev";
};
```

Les fichiers de zones contiennent les enregistrements constituant les différents noms de machines et serveurs du domaine.

Les différents types d'enregistrements les plus courants sont :

- **A** : *Address*, un nom de machine associé à une adresse IP.
- **CNAME** : *Canonical NAME*, un alias sur un nom de machine.
- **MX** : *Mail eXchange*, noms des serveurs de mails du domaine.
- **NS** : *Name Server*, noms des serveurs DNS.
- **SOA** : *Start Of Authority*, informations à propos de cette zone.
- et pour la résolution inverse **PTR** : *PoinTeR*, une adresse IP associée à un nom de machine.

```
$TTL 86400
@   IN  SOA ns.xxx.lasalle. root.xxx.lasalle. (
            12345 ; Version
            21600 ; Refresh secondaires
            3600 ; Attente après demande erronee
            604800 ; TTL max dans les caches des DNS secondaires
            86400 ; TTL min dans les caches
        )
    IN  NS  ns.xxx.lasalle.
    IN  MX  10 mail.xxx.lasalle.

@   IN  A   192.168.0.2
ns  IN  A   192.168.0.2
mail    IN  A   192.168.0.2
server IN  A   192.168.0.2
client IN  A   192.168.0.3
www   IN CNAME server
```

Documentation : https://doc.ubuntu-fr.org/bind9