Administration Systèmes et Serveurs

Claude Duvallet

Université du Havre
UFR des Sciences et Techniques
25 rue Philippe Lebon
BP 540
76058 Le Havre Cedex
Courriel : Claude.Duvallet@gmail.com

Plan de la présentation

Network File System (NFS)

Network File System (NFS)

- Protocole développé par Sun Microsystems.
- Permet à un ordinateur d'accéder à des fichiers via un réseau.
- Fait partie de la couche application du modèle OSI.
- Système de fichiers en réseau permettant de partager des données principalement entre systèmes UNIX.
- Des versions existent pour Macintosh ou Microsoft Windows.
- Compatible avec sur la plupart des systèmes.

Les différentes versions

- Les versions 1 et 2 sont non sécurisées, prévues pour fonctionner sur UDP.
- La version 3 est étendue pour prendre en charge TCP.
- La gestion de la sécurité reste élémentaire dans la version 3 et souffre d'importantes lacunes.
- La version 4 du protocole marque une rupture totale avec les versions précédentes :
  - L'ensemble du protocole est repensé, et les codes sont réécrits.
  - Il s'agit d'un système de fichiers objet.

Network File System version 4 (1/2)

- Une gestion totale de la sécurité :
  - Négociation du niveau de sécurité entre le client et le serveur,
  - Sécurisation simple, support de kerberos5, certificats SPKM et LIPKEY2,
  - Chiffrement des communications est rendu possible.
- Support accru de la montée en charge :
  - Réduction du trafic par groupement de requêtes (compound),
  - Délégation (le client gère le fichier en local).
- Systèmes de maintenances simplifiés :
  - Migration : le serveur NFS est migré de la machine A vers la machine B de manière transparente pour le client,
  - Réplication : le serveur A est répliqué sur la machine B.

Network File System version 4 (2/2)

- Reprise sur incidents :
  - La gestion de la reprise sur incident est intégrée du côté client et du côté serveur.
- Compatibilité :
  - NFSv4 peut être utilisé sous Unix et sous MS-Windows,
  - Il est disponible sur Mac depuis MacOS X Lion (10.7)3.
- Support de plusieurs protocoles de transports (TCP, RDMA).

$\Rightarrow$ NFSv4 est incompatible avec NFSv4.

$\Rightarrow$ NFSv4 n'est pas prévu pour fonctionner avec UDP.

Installation et configuration

- Installation des packages :
  `sudo apt-get -y --force-yes install nfs-kernel-server`

- Configuration :
  - Modification du fichier `/etc/exports`.
  - Syntaxe :
    `<dossier partagé> <hôte1>(<options>) <hôte2>(<options>)...`
  - Exemple :

```
/dossier/à/partager/ 192.168.0.0/24(rw,all_squash,anonuid=1000,
                                    anongid=1000,sync)
/dossier/numero02/  master(rw) trusty(rw,no_root_squash)
```

⇒ N'oubliez pas de relancer le service après la modification du fichier `exports` : `sudo service nfs-kernel-server reload`

- Pour vérifier que l'export a bien eu lieu, taper sur le serveur NFS la commande : `showmount -e`

Les options du fichier exports (1/2)

- `<dossier partagé>` : chemin du dossier partagé.
- `<hôte>` : indique qui peut accéder au partage.
  - une adresse IP : on indique simplement l'adresse IP de la machine pouvant accéder à ce partage.
  - un nom d'hôte : on indique le nom complet de l'hôte (nom DNS).
  - un nom de groupe réseau NIS (NIS netgroup) qui s'indique sous la forme `@<netgroup>`.
  - un domaine avec un joker qui indique les machines d'un domaine ou sous-domaine.
    Par exemple : *.ubuntu.lan.
  - un intervalle d'adresses IP avec le masque de sous-réseau.
    Par exemple : 192.168.0.0/24 ou 192.168.0.*

Les options du fichier exports (2/2)

- `<options>` : ce sont les options de partage.
  - rw : permet la lecture et l'écriture sur un partage pour l'hôte défini. Par défaut, les partages sont en mode lecture (ro).
  - async : permet au serveur NFS de violer le protocole NFS et de répondre aux requêtes avant que les changements effectués par la requête aient été appliqués sur l'unité de stockage.
  - sync : est le contraire de async. Le serveur NFS respecte le protocole NFS.
  - root_squash : force le mapping de l'utilisateur root vers l'utilisateur anonyme (option par défaut).
  - no_root_squash : n'effectue pas de mapping pour l'utilisateur root.
  - all_squash : force le mapping de tous les utilisateurs vers l'utilisateur anonyme.
  - anonuid : indique au serveur NFS l'UID de l'utilisateur anonyme.
  - anongid : indique au serveur NFS le GID de l'utilisateur anonyme.

Le client

- Installation du client :

```
sudo apt-get install nfs-common
```

- Montage d'un répertoire (sur une seule ligne) :

```
sudo mount -t nfs4 adresse_ip:/dossier/partage
                                        /dossier/de/montage
```

- Ajout d'une ligne dans `/etc/fstab` (sur une seule ligne) :

```
adresse_ip:/dossier/partage /dossier/de/montage nfs4
                                        defaults,user,auto 0 0
```

La commande `mount`

- Elle permet d'accéder à des points de montage locaux ou réseaux (NFS, Samba, etc.).
- Sa syntaxe générale est la suivante :

  ```
  mount [options] périphérique répertoire
  ```

- Le périphérique peut être situé sur un autre serveur (Exemple : partage NFS) ou sur le même serveur (autre partition).
- On associe donc un périphérique à un répertoire afin de pouvoir y accéder.

La commande `mount`

- Plusieurs options sont disponibles :
  - `-t type` : permet de spécifier le type de système de fichiers du périphérique (vfat, ntfs, ext4, reiserfs, nfs4, iso9660, auto, etc.)

exemples :
  - `mount -t nfs4 serveur:/nfs/export /montage/local` : permet d'accéder par la suite au partage NFS.
  - `mount -t ext4 /dev/sda2 /home` permet d'associer la seconde partition de votre disque dur (sda) au répertoire home.

remarques : Ces mêmes montages peuvent être spécifiés dans le fichier /etc/fstab afin de les rendre automatiques au démarrage du système.

  - `-r` ou `-o ro` : Le système de fichiers sera monté en lecture seule.
  - `-w` ou `-o rw` : Le système de fichiers sera monté en lecture/écriture. C'est l'option par défaut.
- La commande `umount` permet de démonter un périphérique à partir de son nom ou du nom du répertoire sur lequel il était monté.

Exercice

- Sur un serveur, mettre en place deux partages :
  - Partager le répertoire `/exports/public/` de façon à ce que toutes les machines de votre salle puissent y accéder.
  - Partager le répertoire `/exports/a104-XX/` de façon à ce que la machine de votre voisin puisse y accéder.
- Coté client :
  - Tester vos partages avec la commande `mount`.
  - Mettez en place les montages automatiques des répertoires partagés.

Network Information Service (NIS)

Network Information Service (NIS)

- Son but est de distribuer les informations contenues dans des fichiers de configuration contenant par exemple les noms d'hôte (`/etc/hosts`), les comptes utilisateurs (`/etc/passwd`), etc. sur un réseau.
- Un serveur NIS stocke et distribue donc les informations administratives du réseau, qui se comportent ainsi comme un ensemble cohérent de comptes utilisateurs, groupes, machines, etc.
- À l'origine, NIS est sorti sous le nom de « Yellow Pages » (YP) ou Pages jaunes mais le nom étant déposé par la compagnie britannique British Telecom, Sun a renommé son protocole NIS.
- Les commandes NIS commencent toutes par yp.
- NIS est réputé pour être faible en termes de sécurité.

Composition et fonctionnement

- NIS comporte un serveur, une bibliothèque d'accès client et des commandes d'administration.
- Le serveur NIS génère des cartes (aussi appelé maps) stockées dans des fichiers de base de données (en général DBM ou GDB) à partir des fichiers de configuration.
- Le client récupère les informations en interrogeant le serveur à partir d'appels RPC.

Les commandes

Quelques commandes de base du côté des postes clients :

- `yppasswd` : Cette commande permet de changer le mot de passe d'un utilisateur NIS.
- `ypcat` : Cette commande permet d'afficher un fichier NIS. Exemple : `ypcat passwd`.
- `ypwhich` : Cette commande permet de connaître le nom du serveur NIS utilisé.
- `domainname nomdudomaine` : Cette commande permet de (re)définir le domaine NIS.

Installation et configuration (1/2)

- Installation des packages :

```
sudo apt-get -y --force-yes install portmap nis
```

  Bien définir le même nom de domain sur le serveur et le client.
- Configuration (Coté serveur) :
  - Ajout de la ligne suivante dans le fichier `/etc/hosts.allow`.

```
portmap ypserv ypbind : "Liste d'adresses IP"
```

  - Editer le fichier `/etc/default/nis` et modifier la ligne NISSERVER :

```
NISSERVER=master
```

  - Editer le fichier `/etc/yp.conf` et ajouter la ligne :

```
ypserver 127.0.0.1
```

  - Editer le fichier `/etc/ypserv.securenets` :
    - Ajouter des ligne de la forme pour limiter les accès :

```
netmask adresse_ip_ou_reseau
```

    - Commenter la ligne contenant `0.0.0.0` : elle autorise tous les accès.
  - Construction de la base pour la première fois :

```
sudo /usr/lib/yp/ypinit -m
```

Installation et configuration (2/2)

- Configuration (Coté client) :
  - Il faut aussi installer le package `nis` sur le client :
  - Ajout de la ligne suivante dans le fichier `/etc/hosts.allow` :
    ```
    portmap : "adresse IP du serveur"
    ```
  - Éditer le fichier `/etc/passwd` afin d'ajouter la ligne suivante :
    ```
    +::::::
    ```
  - Éditer le fichier `/etc/group` afin d'ajouter la ligne suivante :
    ```
    +:::
    ```
  - Éditer le fichier `/etc/shadow` afin d'ajouter la ligne suivante :
    ```
    +::::::::
    ```
  - Éditer le fichier `/etc/yp.conf` afin d'ajouter la ligne suivante :
    ```
    domain nom_du_domaine server nom_du_serveur
    ```
    Sur le serveur, c'est indiqué dans `/etc/defaultdomain`.
  - Dans le fichier `/etc/hosts` ajouter l'adresse IP du serveur :
    ```
    adresse_ip nom_du_serveur
    ```
- Redémarrer le serveur ainsi que le client.

Exercice

- Configurer un client et un serveur.
  - Sur le serveur, après avoir configuré NIS avec le nom de domaine `ubuntu-nis`, ajoutez un compte utilisateur `nisuser`.
  - Sur le client, configuré avec le même nom de domaine.
- Tenter de vous connecter sur le client avec le compte `nisuser`.

Dynamic Host Configuration Protocol (DHCP)

Dynamic Host Configuration Protocol (DHCP)

- C'est un protocole de configuration dynamique des hôtes.
- Service de la couche applicative au sein de l'architecture TCP/IP.
- Il permet aux ordinateurs clients l'obtention automatique d'une configuration réseau (adresse IP, passerelle, DNS, etc.).
- Il évite la configuration de chaque ordinateur manuellement.
- Les ordinateurs configurés pour utiliser DHCP n'ont pas le contrôle de leur configuration réseau : ils la reçoivent du serveur DHCP.
- C'est complétement transparent pour l'utilisateur.
- De façon générale, les adresses IP peuvent être attribuées de deux manières :
  - statique : en configurant le réseau directement sur l'ordinateur,
  - dynamique : en utilisant un serveur DHCP qui attribue les adresses en fonction de son fichier de configuration.

Installation

- Pour installer le serveur DHCP, il faut installer le paquet `isc-dhcp-server` :
  `sudo apt-get install isc-dhcp-server`
- Il faudra changer la configuration par défaut en éditant le fichier `/etc/dhcp/dhcpd.conf`.
- Dans ce fichier, on définit l'ensemble des options globalement ou par réseau.
- Vous aurez également besoin d'éditer le fichier `/etc/default/isc-dhcp-server` pour spécifier les interfaces que `dhcpd` (le démon de `isc-dhcp-server`) devra écouter.
- Par défaut, le démon `dhcpd` écoute l'interface eth0.

$\Rightarrow$ Les interfaces réseaux de votre serveur doivent être configurées obligatoirement en adresses IP statiques.

Exemple de configuration

- La configuration la plus fréquente est d'assigner aléatoirement une adresse IP.

```
default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option domain-name-servers 192.168.1.1, 192.168.1.2;
option domain-name "ubuntu.lan";
option ntp-servers 192.168.1.254;

subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.10 192.168.1.100;
    range 192.168.1.150 192.168.1.200;
}
```

Exemple de configuration : explications

- Le serveur DHCP assignera au client une adresse IP comprise entre 192.168.1.10 et 192.168.1.100 ou entre 192.168.1.150 et 192.168.1.200 pour une durée de 600 secondes.
- Le client peut spécifier une période de temps spécifique, dans ce cas, le temps d'allocation maximum est de 7200 secondes.
- Le serveur va également informer le client qu'il doit utiliser :
  - un masque de sous réseau à 255.255.255,
  - une adresse de multi-diffusion à 192.168.1.255,
  - une adresse de routeur/passerelle à 192.168.1.254,
  - des serveurs DNS à 192.168.1.1 et 192.168.1.2,
  - un suffixe DNS ubuntu.lan,
  - un serveur de temps (NTP).

Configuration : Adresses IP fixes uniquement

- Dans ce cas, l'adresse IP que reçoit le client est toujours la même.
- Pour cela il suffit d'ajouter une directive host dans la définition du subnet.
- Pour chaque client, il faut donner son adresse fixe en fonction de son adresse MAC.

```
deny unknown-clients;

subnet 192.168.1.0 netmask 255.255.255.0 {
    host client1 {
        hardware ethernet 08:00:27:15:01:81;
        fixed-address 192.168.1.20;
    }
    host client2 {
        hardware ethernet 00:JJ:YU:38:AC:45;
        fixed-address 192.168.1.21;
    }
}
```

Plusieurs interfaces (1/2)

- Pour que le serveur écoute sur certaines interfaces, il faut les spécifier dans `/etc/default/isc-dhcp-server` :
  ```
  INTERFACES="eth0 eth1"
  ```
- Dans ce cas l'écoute se fait sur eth0 et eth1.
- Soient 3 réseaux :
  - Le réseau internet.
  - Le réseau local 192.168.1.* réservé aux serveurs (web, FTP, messagerie, etc.).
  - Le réseau local 192.168.2.* réservé aux clients (réseau local partagé).
- Il y a 4 autres machines sur les réseaux :
  - 192.168.1.2 (nommée ftp),
  - 192.168.1.3 (nommée web),
  - 192.168.1.4 (nommée mail) et
  - 192.168.2.2 (nommée portable).

Plusieurs interfaces (2/2)

- Aucune machine inconnue ne se verra attribuer une adresse IP par DHCP.
- Toutes les machines des réseaux ont la possibilité de démarrer par PXE.
- La machine serveur DHCP est aussi le routeur/pare-feu/NAT.
- Il fait aussi office de serveur DNS du domaine ubuntu.lan.
- Les interfaces sur lesquelles le serveur démarre doivent avoir une adresse quand le service dhcp démarre.
- On leur attribuera les adresses 192.168.1.1 et 192.168.2.1.

Configuration du serveur : /etc/dhcp/dhcpd.conf

```
### RÉSEAU #####
## Nom du serveur DHCP
server-name "dns.ubuntu.lan";

## Mode autoritaire (autoritaire)
authoritative;

## Masque de sous-réseau
option subnet-mask 255.255.255.0;

### DOMAINE ###
## Nom du domaine
option domain-name "ubuntu.lan";

## Adresse IP du serveur DNS
option domain-name-servers XXX.XXX.XXX.XXX;

## Type de mise à jour du DNS (aucune)
ddns-update-style none;

### TEMPS DE RENOUVÈLEMENT DES ADRESSES ###
default-lease-time 3600;
max-lease-time 7200;
```

Configuration du serveur : /etc/dhcp/dhcpd.conf

```
### Sécurité ###
## refus(deny)/autorise(allow) les clients inconnus
deny unknown-clients;

### PXE ### Permet le boot réseau pour TFTP
allow bootp;
allow booting;

## déclaration pour le sous-réseau 192.168.1.*
subnet 192.168.1.0 netmask 255.255.255.0 {
  # Spécification d'un domaine différent de celui par défaut
  option domain-name "ubuntu.lan";
  option broadcast-address 192.168.1.255;
  ## routeur par défaut
  option routers 192.168.1.1;
  ## Plage d'attribution d'adresses
  range 192.168.1.6 192.168.1.7;
  filename "pxelinux.0";
  # définit le serveur qui servira le fichier "pxelinux.0"
  next-server 192.168.2.1;
  # évalue si l'adresse est déjà attribuée
  ping-check = 1;
}
```

Configuration du serveur : /etc/dhcp/dhcpd.conf

```
## Déclaration sous réseau 192.168.2.*
subnet 192.168.2.0 netmask 255.255.255.0 {
  option domain-name "ubuntu.lan";
  option broadcast-address 192.168.2.255;
  option routers 192.168.2.1;
  range 192.168.2.2 192.168.2.3;
  ping-check = 1;
  filename "pxelinux.0";
  next-server 192.168.2.1;
}

#### Configuration des hôtes avec IP fixée ####
# hôte FTP
host ftp {
  hardware ethernet 00:0f:75:af:eb:44;
  fixed-address 192.168.1.2;
}
```

Configuration du serveur : /etc/dhcp/dhcpd.conf

```
# hôte WEB
host web {
  hardware ethernet 00:02:0d:31:d1:cc;
  fixed-address 192.168.1.3;
}
# hôte mail
host mail {
  hardware ethernet 00:02:55:d2:d1:cc;
  fixed-address 192.168.1.4;
}
# hôte PORTABLE
host portable {
  hardware ethernet 00:0e:af:31:d1:cc;
  fixed-address 192.168.2.2;
}
```

Vérifications

- Les messages d'erreurs sont le fichier suivant :
  `tail /var/log/syslog`
- Pour que les log soient enregistrés dans un autre fichier (par exemple `/var/log/dhcpd.log`) :
  - Dans le fichier `/etc/dhcp/dhcpd.conf`, il faut ajouter la ligne suivante :
    ```
    log-facility local7;
    ```
  - Il faut créer le fichier `/var/log/dhcpd.log` avec comme propriétaire `syslog` (droits en lecture/écriture) et comme groupe `adm` (droits en lecture).
    ```
    sudo touch /var/log/dhcpd.log
    sudo chown syslog:adm /var/log/dhcpd.log
    sudo chmod 0640 /var/log/dhcpd.log
    ```
  - Puis ajouter ceci dans le fichier `/etc/rsyslog.d/50-default.conf` :
    ```
    local7.*                /var/log/dhcpd.log
    ```
  - et relancer le daemon `syslog` :
    ```
    sudo restart rsyslog
    ```

Les baux

- Par défaut, le fichier `/var/lib/dhcp/dhcpd.leases` donne des informations sur les baux actuellement distribués par le serveur.
- On y retrouve des informations essentielles comme l'adresse IP distribué à une adresse MAC, le nom de la machine qui a fait cette demande DHCP, l'heure de début et de fin du bail.
- `/var/lib/dhcp/dhcpd.leases`

```
lease 192.168.2.128 {
  starts 2 2012/07/31 20:24:28;
  ends 3 2012/08/01 01:24:28;
  ...
  hardware ethernet 01:11:5b:12:34:56;
  ...
  client-hostname "machine01";
}
```

Exercice

- Configurer un serveur DHCP pour qu'il distribue dynamiquement des adresses entre 192.168.1.5 et 192.168.2.10.

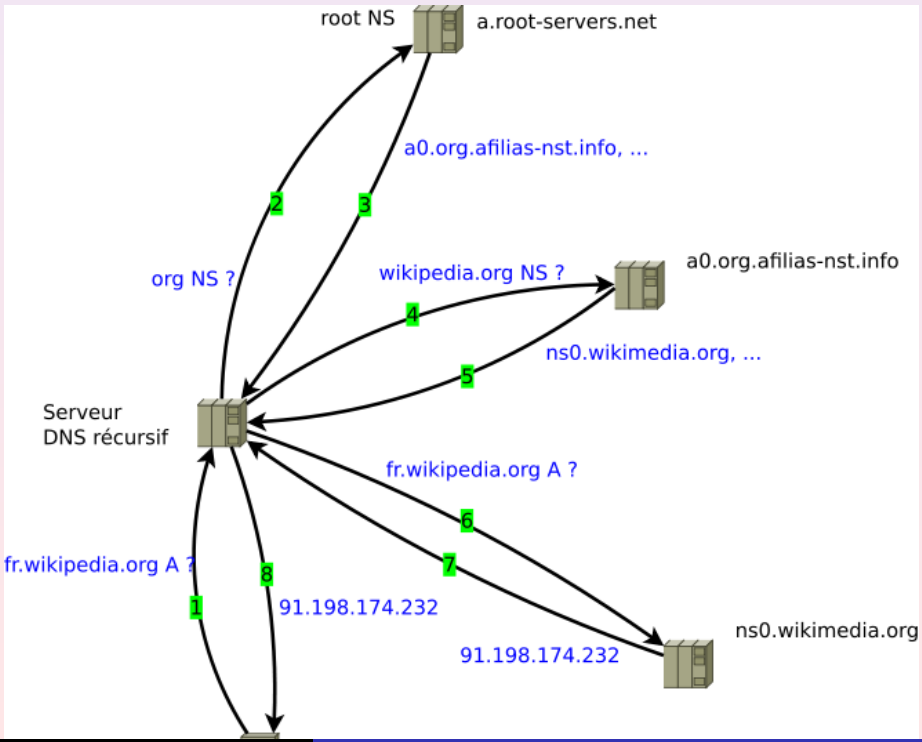
Domain Name System (DNS)

Domain Name System (1/2)

- Le système de noms de domaine (ou DNS, Domain Name System) est un service permettant d'établir une correspondance entre une adresse IP et un nom de domaine.
- Les réseaux informatiques sont composés d'ordinateurs qui communiquent entre eux à l'aide d'adresses numériques uniques, appelées adresses IP.
- Chaque adresse IP identifie un ordinateur (clients et serveurs).
- L'adresse numérique facilite grandement le traitement pour l'ordinateur.
- Or, pour un être humain, il est beaucoup plus difficile de se souvenir d'une série de chiffres que d'une suite de lettres et de mots.

Domain Name System (2/2)

- Le système des noms de domaine a été inventé pour palier à ce problème :
  - il fait correspondre une adresse alphanumérique (www.google.fr) à une adresse IP numérique (88.191.119.240).
- Cette correspondance, appelée résolution, s'effectue à l'intérieur d'un serveur spécialisé dans la résolution de noms de domaine :



Installation des packages

- Installation des packages :

```
sudo apt-get -y --force-yes install dnsutils bind9
```

Plusieurs scénarios de configuration (1/4)

- Serveur cache.
- Serveur maître.
- Serveur esclave.
- Serveur hybride.
- Serveur furtif.
- Serveur récursif/non récursif.

Plusieurs scénarios de configuration (2/4)

- Serveur cache :
  - BIND9 effectue les requêtes DNS et se rappel de la réponse pour la prochaine requête. Les réponses DNS sont en cache.
  - Cette méthode est utile pour une connexion internet lente.
  - Cela augmente la bande passante et réduit également le temps de latence.
- Serveur maître :
  - BIND9 est utilisé pour contenir les enregistrements DNS d'un nom de domaine enregistré.
  - Un ensemble d'enregistrements DNS pour un nom de domaine est appelé une "zone".
- Serveur esclave :
  - Un serveur esclave est utilisé en complément d'un serveur maître, en lui servant de copie pour sa ou ses zones configurée(s).
  - Ce type de serveur est recommandé pour les "gros" réseaux.
  - Il assure la disponibilité de la zone DNS, même si le serveur maître est hors ligne.

Plusieurs scénarios de configuration (3/4)

- Serveur hybride :
  - Un serveur BIND9 peut être configuré à la fois :
    - comme serveur cache et comme serveur maître,
    - comme serveur cache et serveur esclave,
    - ou même serveur cache, serveur maître et esclave.
  - Il suffit de combiner les différentes configurations.
- Serveur récursif/non récursif :
  - Les serveurs BIND9 sont par défaut récursifs.
  - Ils interrogent tour à tour les serveurs DNS nécessaires jusqu'à obtenir la réponse, et la transmettre à leur client.
  - Dans le cas contraire, le serveur DNS délègue la résolution du nom de domaine à un autre serveur DNS.

Plusieurs scénarios de configuration (4/4)

- Serveurs furtifs :
  - Il existe deux autres configurations fréquentes pour un serveur DNS :
    - Serveur furtif maître et serveur furtif esclave.
    - Identiques aux serveurs maître et esclave, mais avec une organisation différente : ils ne sont visibles qu'à l'intérieur du domaine.
  - Exemple : soient 3 serveurs DNS nommés A, B et C.
    - A est un serveur maître, B et C sont des esclaves.
    - Si votre domaine est configuré pour utiliser A et B comme serveurs de noms, alors C est un serveur furtif esclave.

    $\Rightarrow$ Serveur esclave qui ne sera pas interrogé depuis Internet.
    - Si votre domaine est configuré pour utiliser B et C comme serveurs de noms, alors A est un serveur furtif maître.

    $\Rightarrow$ Édition de la zone ou ajout sur A, mais les ordinateurs (depuis Internet) interrogent seulement B et C.
    - Dans les deux cas, le serveur furtif n'est pas interrogé depuis internet. Il peut ainsi être réservé pour une utilisation locale.

Les types d'enregistrements DNS (1/3)

- Enregistrement de type A (Address) : il fait correspondre une adresse IP à un nom de machine.

```
www       IN    A       1.2.3.4
```

- Enregistrement de type CNAME (Alias) : Utilisé pour créer un alias depuis un enregistrement de type A.
- Il est possible de créer un enregistrement de type CNAME qui pointe vers un autre enregistrement CNAME, mais ceci double le nombre de requêtes qui seront faîtes au serveur de noms.

```
mail      IN    CNAME   www
www       IN    A       1.2.3.4
```

Les types d'enregistrements DNS (2/3)

- Enregistrement MX (Mail Exchange) : Utilisé pour définir vers quel serveur de la zone un courriel à destination du domaine doit être envoyé, et avec quelle priorité.
- Cet enregistrement doit pointer vers un enregistrement de type A, et non un alias CNAME.
- Il peut y avoir plusieurs enregistrements MX si il existe plusieurs serveurs de messagerie sur le domaine.

```
            IN    MX  10    mail.ubuntu.lan.
mail        IN    A       1.2.3.4
```

Dans cet exemple, "10" représente la préférence si jamais d'autres serveurs de courriels sont présents dans la même zone. La valeur est comprise entre 0 et 65535.

Les types d'enregistrements DNS (3/3)

- Enregistrement NS (Name Server) : Utilisé pour définir quels serveurs répondent pour cette zone.
- Cet enregistrement doit pointer vers un enregistrement de type A, non pas vers un enregistrement de type CNAME.
- C'est ici que le serveur maître et les esclaves sont définis.
- Les serveurs furtifs sont intentionnellement omis.

```
                IN      NS      ns.ubuntu.lan.
[...]
ns              IN      A       1.2.3.4
```

Configuration de Bind9

- Les fichiers de configuration de BIND9 sont stockés sous : `/etc/bind/`
- La configuration de BIND9 est effectuée dans les fichiers suivants :
  - `/etc/bind/named.conf` : ce fichier contient la configuration générale de Bind.
  - `/etc/bind/named.conf.options` : ce fichier contient la configuration générale de Bind.
  - `/etc/bind/named.conf.local` : ce fichier contient toutes les options (récursivité, forwarding, etc.), les ACL, les vues, la gestion des logs, etc.

Configuration pour un seul ordinateur (1/2)

- BIND est configuré pour ne répondre qu'aux requêtes du PC sur lequel il est installé.
- Il se charge lui même de la résolution de noms, sans passer par les serveurs DNS de votre FAI.
- Editer le fichier `/etc/bind/named.conf.options`, positionner l'option `listen-on` sur l'interface réseau interne `127.0.0.1`.

$\Rightarrow$ BIND ne sera plus accessible depuis l'extérieur.

- Modifier la ligne du fichier `/etc/bind/named.conf.options` qui ressemble à ceci :

```
listen-on-v6 { any; };
```

- Afin qu'elle ressemble à cela :

```
listen-on-v6 { ::1; };
```

Configuration pour un seul ordinateur (2/2)

- Commenter l'option `forwarders`. Il suffit de mettre un # devant chaque ligne :

```
#// forwarders {
#//  0.0.0.0;
#// };
```

- Si votre carte réseau est configurée pour utiliser DHCP, décommenter la ligne du fichier `/etc/dhcp/dhclient.conf` :

```
prepend domain-name-servers 127.0.0.1;
```

- Si, au contraire, elle est configurée avec une adresse IP statique, modifier le fichier `/etc/resolv.conf` afin que toutes les requêtes passent par BIND :

```
nameserver 127.0.0.1
```

- Redémarrer bind :

```
sudo service bind9 restart
```

Configuration pour un serveur cache (1/2)

- Le serveur BIND9 est configuré par défaut en tant que serveur cache.
- Il suffit simplement d'ajouter les serveurs DNS de votre prestataire Internet.
- Décommentez et éditez les lignes suivantes dans `/etc/bind/named.conf.options` :

```
[...]
forwarders {
     0.0.0.0;
};
[...]
```

ou 0.0.0.0 est l'adresse IP du ou des serveurs DNS de votre prestataire Internet.

- Redémarrez le démon BIND9 :

```
sudo service bind9 restart
```

Configuration pour un serveur cache (2/2)

- Si le package dnsutils a été installé, il est possible de tester la nouvelle configuration en utilisant : `dig -x 127.0.0.1`
- Si tout fonctionne bien, vous devriez voir apparaître une sortie similaire à :

```
; <<>> DiG 9.9.5-3ubuntu0.6-Ubuntu <<>> -x 127.0.0.1
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 5303
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;1.0.0.127.in-addr.arpa. IN PTR

;; Query time: 10 msec
;; SERVER: 127.0.1.1#53(127.0.1.1)
;; WHEN: Thu Jan 07 10:33:41 CET 2016
;; MSG SIZE  rcvd: 40
```

- La commande `dig` peut être utilisée pour interroger d'autres domaines : `dig google.com`
- Si vous "diggez" un même domaine plusieurs fois, vous devriez voir apparaître une énorme diminution du temps, entre la première et la deuxième requête.

Configurer un serveur maître (1/8)

- BIND9 va être configuré comme serveur maître pour le domaine `ubuntu.lan`.
- Remplacez simplement `ubuntu.lan` par votre propre nom de domaine si vous souhaitez un autre DNS.
- Pour ajouter une zone, et faire de BIND9 un serveur maître, éditer le fichier `named.conf.local` :

```
[...]
zone "ubuntu.lan" {
    type master;
    file "/etc/bind/db.ubuntu.lan";
};
[...]
```

- Utiliser le fichier d'une zone existante comme modèle :

```
sudo cp /etc/bind/db.local /etc/bind/db.ubuntu.lan
```

Ce fichier contiendra les enregistrements de votre zone.

Configurer un serveur maître (2/8)

- Editer le nouveau fichier pour la zone (`/etc/bind/db.ubuntu.lan`),
- Changer localhost par le FQDN (Fully Qualified Domain Name ou Nom de domaine pleinement qualifié) de votre serveur, en laissant le point "." supplémentaire à la fin.
- Changer `127.0.0.1` par l'adresse IP du serveur de nom et `root.localhost` par une adresse courriel valide, mais avec un point "." à la place de l'arobase "@".
- Laisser également le point à la fin.

Configurer un serveur maître (3/8)

- Créer un enregistrement de type hôte A pour le serveur de nom `ns.ubuntu.lan` :

```
;
; BIND data file for local eth0 interface
;
$TTL 604800
@ IN SOA ns.ubuntu.lan admin.ubuntu.lan (
      2 ; Serial
604800 ; Refresh
  86400 ; Retry
2419200 ; Expire
604800 ) ; Negative Cache TTL
;
@ IN NS ns.ubuntu.lan.
NS IN A 192.168.0.11
```

- Le numéro de série doit être incrémenté à chaque changement dans le fichier de zone.
- En cas de multiples changements, une seule incrémentation suffit.

Configurer un serveur maître (4/8)

- Il est fréquent d'utiliser la date d'édition de la zone comme numéro de série, au format américain. Exemple : 2010122710 = incrémentation 10 du 27 décembre 2010).
- Il est maintenant possible d'ajouter des enregistrements DNS à la suite de la zone .
- Une fois les changements dans le fichier de zone effectués, il faut redémarrer BIND9 pour qu'ils prennent effet :
  `sudo service bind9 restart`
- Maintenant que notre fichier de zone est configuré et que les adresses IP sont résolues, une zone de recherche inversée est requise.
- Une zone de recherche inversée permet au DNS de convertir une adresse en nom.

Configurer un serveur maître (5/8)

- Editer `/etc/bind/named.conf.local` et ajouter les lignes suivantes :

```
zone "1.168.192.in-addr.arpa" {
    type master;
    notify no;
    file "/etc/bind/db.192";
};
```

- Remplacer 1.168.192 par les trois premiers octets (si vous êtes en classe C) de votre réseau dans l'ordre inversé.
- Remplacer également le nom du fichier de zone db.192 par le nom approprié.

Configurer un serveur maître (6/8)

- Créer maintenant le fichier db.192 depuis un fichier existant :
  `sudo cp /etc/bind/db.127 /etc/bind/db.192`
- Editer le fichier `/etc/db.192` et changer comme nous l'avons fait précédemment le nom de domaine et l'adresse courriel :

```
$TTL 604800
@ IN SOA ns.ubuntu.lan admin.ubuntu.lan (
      3 ; Serial
604800 ; Refresh
  86400 ; Retry
2419200 ; Expire
604800 ) ; Negative Cache TTL
;
@ IN NS ns.ubuntu.lan.
11 IN PTR ns.ubuntu.lan.
```

- Le numéro de série de la zone de recherche inversée (PTR) nécessite d'être incrémenté à chaque changement.

Configurer un serveur maître (7/8)

- Les enregistrements PTR (PoinTeR) servent à la résolution inverse des noms.
- Dans `resolv.conf`, il faut écrire la ligne `nameserver 127.0.0.1`.
- Pour chaque enregistrement A ajouté dans `/etc/bind/db.ubuntu.lan`, il faut créer un enregistrement PTR dans `/etc/bind/db.192`.
- Après avoir créé le fichier de la zone de recherche inversée, redémarrez BIND9 : `sudo service bind9 restart`

Configurer un serveur maître (8/8)

- Il doit maintenant être possible de faire un ping sur `ubuntu.lan` et la requête doit être résolue : `ping ubuntu.lan`
- L'utilitaire `named-checkzone` (inclus dans le package BIND9) peut également être utilisé :
  `named-checkzone ubuntu.lan /etc/bind/db.ubuntu.lan`
  et
  `named-checkzone ubuntu.lan /etc/bind/db.192`
- Pour tester la recherche inversée, l'utilitaire dig peut être utilisé :
  `dig 1.168.192.in-addr.arpa. AXFR`
- Vous devriez voir en sortie "console", la résolution de `1.168.192.in-addr.arpa.` par votre serveur de nom.