Introduction à la sécurité TP 1 - Retour à l'essentiel

Maciej Korczynski & Simon Fernandez

Septembre 2020

Objectifs

- Familiarisation avec l'environnement (installation + utilisation de base)
- Installation et configuration du serveur web

Pour une rapide introduction à Linux : http://doc.ubuntu-fr.org/tutoriel/learn_unix_in_10_minutes

Notions nécessaires / Rappels

Media d'installation

Une installation d'un système d'exploitation peut se faire via plusieurs médias:

- CD/DVD
- Clef USB
- PXE (installation par le réseau)

Dans notre cas nous travaillerons sur une machine virtuelle, nous simulerons donc la présence d'un lecteur CD/DVD.

Partitionnement

Un disque dur physique peut être divisé en plusieurs disques logiques, on parle alors de partitionnement. Chaque partition possède son propre format de fichier, les plus connues sont :

- ext2/3/4 (linux)
- FAT32 (old Windows)
- NTFS (Windows)

Il existe des partitions ayant un rôle particulier, c'est le cas de la partition SWAP dans un environnement linux, qui est utilisée par le système comme de la mémoire vive quand la RAM est pleine.

Réseau

Les commandes `ifconfig` et `ip` permet de configurer manuellement les interfaces réseaux.

Exemple d'utilisation de `ifconfig` (voir `man ifconfig` pour plus de détails) :

- `ifconfig` : Liste les interfaces actives
- `ifconfig -a` : Liste toutes les interfaces
- `ifconfig INTERFACE` : information sur l'interface (IP/Masque/MAC..)
- `ifconfig INTERFACE up/down` : active / désactive une interface

`ifconfig` est de plus un plus remplacé par la commande `ip`.

Voici quelques équivalent aux commandes ci-dessus avec `ip` (`man ip` pour plus de détails)

- `ip link` : Liste les interfaces actives
- `ip addr show dev INTERFACE` : information sur l'interface (IP/Masque/MAC..)
- `ip link set INTERFACE up/down` : active / désactive une interface

Libre à vous d'utiliser dans la suite des TPs `ifconfig` ou `ip`

D'autres outils utiles:

- `dhclient INTERFACE` : permet de forcer une demande d'adresse IP aux serveurs DHCP.
- `route -n` permet de lister les « routes du réseaux » (c'est à dire par quelles interfaces/passerelles un paquet doit sortir en fonction de sa destination).

Note : le préfixe `sudo` avant une commande permet d'avoir les droits d'administrateurs. Par exemple pour désactiver l'interface `eth0`, il faut la rajouter avant : `sudo ifconfig eth0 down`

Pour une configuration persistante, on peut:

- Modifier `/etc/resolv.conf` : contient la configuration permettant de trouver le serveur DNS (la conversion URL - IP), ou de manuellement spécifier l'IP à utiliser pour certains domaines. Voir man `resolv.conf`.
- Pour les Ubuntu récents (18.04 & suivants): utiliser l'outil `netplan`
- Pour les vieux Ubuntu: éditer `/etc/network/interfaces`

Exemple de fichier `/etc/resolv.conf`

```
domain e.ujf-grenoble.fr
search e.ujf-grenoble.fr
nameserver 193.54.238.51
```

Réseau et machine virtuelle

Nous travaillerons sur une machine virtuelle (VM), nous devons donc simuler dans la machine une interface réseau. Il existe trois modes fréquemment utilisés :

- **NAT** (Network Address Translation) : dans ce mode la VM posséde une addresse non routable dans le réseau de la machine hôte. La machine hôte se chargera alors du routage entre la VM et son réseau. La VM n'est pas accessible directement depuis l'extérieur dans ce mode. Afin qu'elle le soit, il faut configurer une redirection de port entre la machine hôte et la VM.
- **Accès par pont (bridge)** : ce mode créera un pont entre l'interface de la VM et l'interface de la machine hôte. La VM sera vue par le réseau comme une machine comme les autres, et pourra donc posséder sa propre adresse IP et être accessible depuis le réseau. Cette configuration dans VirtualBox se fait dans Périphériques : Cartes réseau...
- **Réseau privé hôte (host-only network)**: VirtualBox va créer un réseau virtuel sur lequel seront connectées les VM et la machine hôte. Les machines pourront donc communiquer entre elles, sans jamais passer par le reste du réseau. Pour cela, dans VirtualBox, il faut créer un réseau virtuel, dans Files > Host Network Manager, puis configurer la VM

Package

Les paquets (packages) sont des fichiers permettant l'installation de programme. Les gestionnaires de paquets ont pour but de les gérer. Ils en existent plusieurs, parmis eux :

- `dpk` (Debian PacKage) : Pour installer les fichiers `.deb`
- `apt` (Advanced Packaging Tool) : Pour installer des logiciels depuis les dépôts APT (Ubuntu et ses déclinaisons)
- `rpm` (Red Hat Package Manager) : Pour Red Hat / Fedora / ...
- `Click-Package` :Pour les applications Ubuntu mobiles (en cours de développement)

Pour gérer un paquet venant d'un fichier .deb :

- `dpkg -i PACKAGE_NAME` : Installe le paquet
- `dpkg -r PACKAGE_NAME` : Désinstalle le paquet
- `dpkg -l` : Liste les paquets installés
- `dpkg -L PACKAGE_NAME` : Liste les fichiers utilisés par le programme

La liste de dépôts APT se trouve dans le fichier `/etc/apt/sources.list` . Les commandes couramment utilisées pour gérer ces dépôts sont :

- `apt update` : Mise à jour du cache des dépôts

- `apt upgrade` : Mise à jour des paquets
- `apt search NAME` : Rechercher le nom d'un paquet
- `apt install PACKAGE_NAME` : Installer un paquet
- `apt-get remove PACKAGE_NAME` : Désinstaller un paquet

Gestion des utilisateurs

Les fichiers contenant les informations sur les utilisateurs sont

- `/etc/passwd` : Liste des utilisateurs, sous la forme `login:password:uid:gid:comment:shell`
- `/etc/group` : Liste des groupes, sous la forme `groupname:password:gid:members`

Généralement les mots de passes ne sont pas stocker dans ces fichiers (un 'x' se trouve alors dans le champs password), mais les empreintes (hash) des mots de passe sont stockées dans le fichier `/etc/shadow` (voir man shadow), qui n'est accessible que par les utilisateurs ayant les droits d'administrateurs.

Les commandes les plus courrantes pour la gestion des utilisateurs et des groupes sont :

- `groupadd` : Ajoute un groupe
  - `groupadd -g GROUPE_ID GROUPE_NAME`
- `useradd` : Ajoute un utilisateur
  - `useradd -d USER HOME_DIR -u USER_ID -g GROUPE_ID -s /bin/bash -m USER_NAME`
- `userdel` : Supprime un utilisateur
  - `userdel USER_NAME`
- `passwd` : Change le mot de passe d'un utiliateur
  - `passwd USER_NAME`
- `id` : Information sur l'utilisateur
  - `id USER_NAME`
- `usermod` : Modifie un utilisateur
- `usermod -G LIST_GROUPE USER_NAME` : change les groupes d'un utilisateur (LIST_GROUPE sous la forme "Groupe1,Groupe2,..")

Apache2

Apache2 est un serveur web open-source populaire et très utilisé. La configuration principale se trouve dans le fichier `/etc/apache2/apache2.conf`, qui fait appel à d'autre fichiers de configuration (via la directive `Include`), afin de répartir cette configuration.

Un serveur apache peut gérer plusieurs sites web, on alors parle de Virtual Host pour chaque site.

Le(s) site(s) web se trouve(nt) généralement dans le répertoire `/var/www/`. Lors de l'installation, un site par défaut se créer (accessible depuis http://localhost:80).

Ils existent plusieurs façons de faire cohabiter plusieurs sites sur une même machine :

- Virtual host basé sur le nom de domaine
- Virtual host basé sur l'adresse IP
- Virtual host basé sur le port

Dans la suite nous allons voir comment mettre en place un site en fonction de son port : le site "site1" sur le port 8080. Par défaut apache2 écoute sur le port 80 (et 443 pour les connections sécurisées). La liste des ports écoutés se trouve dans le fichier `/etc/apache2/ports.conf`. L'écoute d'un port est spécifiée par la directive `Listen`.

Ainsi, pour que le serveur écoute le port 8080, il faut rajouter `Listen 8080` dans le fichier `ports.conf`.

Chaque site possède un fichier de configuration se trouvant dans `/etc/apache2/sites-avaible/`

Exemple de fichier de configuration (très) simple se trouvant dans `/etc/apache2/sites-avaible/site1` :

```
<VirtualHost *:8080>
        DocumentRoot    /var/www/site1
</VirtualHost>
```

- `<VirtualHost *:8080>` : `*` correpond à l'addresse IP à écouter (`*` représente toute les adresses), et 8080 le port
- `DocumentRoot /var/www/site1` : correspond à l'emplacement du site sur le serveur, c'est le dossier contenant les pages et le contenu.

Une manière simple de mettre en place une authentification sur un site web est d'utiliser `htpasswd` afin de gérer les utilisateurs depuis un fichier.

- `htpasswd -c /chemin/vers/fichier/.htpasswd USER` : créer le fichier avec l'utilisateur `USER`
- `htpasswd /chemin/vers/fichier/.htpasswd USER` : créer ou modifier l'utilisateur `USER`
- `htpasswd -d /chemin/vers/fichier/.htpasswd USER` : supprimer l'utilisateur USER

Il est recommandé que `/chemin/vers/fichier/` ne fasse pas partie de l'arborescence du site web.

Une fois le fichier `.htpasswd` créé, il faut renseigner au fichier de configuration du site (dans le dossier `sites-available`), la recherche de la méthode d'authentification.

```
<VirtualHost *:8080>
        DocumentRoot    /var/www/site1

        <Directory /var/www/site1>
            AuthUserFile /chemin/vers/fichier/.htpasswd
            AuthName "Welcome to Hell!"
            AuthType Basic
            Require valid-user
        </Directory>
</VirtualHost>
```

Note : une idée communément répandue veut l'utilisation de fichier `.htaccess` (et pas `.htpasswd`) pour la gestion des règles. Seulement cette utilisation dans un cas général est déconseillée et ne devrait être employée que dans des cas particuliers (voir https://httpd.apache.org/docs/2.2/howto/htaccess.html#when)

Pour activer la configuration d'un site se trouvant dans `/etc/apache2/sites-available/`: `sudo a2ensite site1`
Pour le désactiver : `sudo a2dissite site1`

On peut vérifier qu'un site a été activé par la présence d'un fichier portant son nom dans le répertoire `/etc/apache2/sites-enabled/`

Il ne faut pas oublier de redémarrer le serveur apache2 après avoir activé/désactivé un site pour que les changements soient prisent en compte.

- `sudo systemctl start|restart|stop apache2`

Travail demandé / Assignment

Setup

- Téléchargez `Ubuntu` en version Serveur.
- Créez une nouvelle machine virtuelle depuis virtualbox avec pour nom `CySec-NAME`.
  - Choisissez de créer un disque d'amorçage VDI, dynamiquement alloué, de 10Go.
- Lors du premier lancement, choisissez l'ISO Ubuntu comme source de media.
- Installez Ubuntu et suivez les instructions de l'installeur. Laissez les options par défaut, et ne demandez l'installation d'aucun service.
- Configurez la carte réseau de la machine virtuelle en NAT.

Exercices

- Donnez l'adresse IP de la machine hôte ainsi que de la machine virtuelle sur le réseau `Host-only`

Bonus : Trouvez comment changer l'addresse IP et l'adresse MAC d'une interface avec la commande `ifconfig` ou `ip address`

- Créez deux groupes d'utilisateur.
- Créez un utilisateur par groupe, plus un troisième qui devra se trouver dans les deux groupes.

Bonus : Trouvez la commande permettant de supprimer directement le répertoire d'un utilisateur quand on supprime l'utilisateur (voir `userdel`)

- Mettez à jour le cache de dépôt APT et installez `apache2`.
- Retirez le site par défaut d'apache (nom du site: `default`) et configurez Apache2 avec deux sites web (une simple page html sera suffisant par site).
  - Un site accessible publiquement
  - Un site accessible uniquement par certains utilisateurs (utilisez `.htpasswd`).

Bonus : Que sont les Multi-Processing Modules (MPM) ? Citez des exemples de MPM et expliquer leurs différences.

Pour ceux qui finissent plus tôt (Bonus)

- Trouvez un moyen de faire communiquer (avec ping par exemple), deux machines virtuelles, venant de deux pcs différents, avec ces contraintes :
  - Utilisez une connexion par pont à la place du NAT
  - N'utilisez pas l'IP et le sous réseau fourni par le DHCP (utilisez un autre sous-réseau)