

## Titre du TP : Sécurité Unix et Gestion des Utilisateurs

### Introduction :

Ce TP vise à explorer les principes de sécurité Unix et les bonnes pratiques de gestion des utilisateurs. Les exercices couvriront la séparation des privilèges, l'utilisation de comptes utilisateur, les mises à jour et correctifs, ainsi que la configuration d'un pare-feu pour assurer la sécurité réseau.

### Exercice 1 : Séparation des Privilèges

Identifiez les fichiers système critiques sur votre système.

Vérifiez les permissions associées à ces fichiers.

Comparez les permissions des fichiers système avec ceux des fichiers utilisateurs.

### Correction:

Fichiers critiques: Ce sont les fichiers essentiels pour le démarrage, la configuration, et la gestion du système et des utilisateurs :

Le répertoire **/etc/** qui contient plusieurs fichiers qui contiennent des informations sur les utilisateurs, groupes etc.. **/etc/passwd /etc/hosts /etc/sudoers /etc/groups**

Le répertoire **/boot** contient les fichiers nécessaires au démarrage du système, y compris le noyau Linux

Le répertoire **/bin** qui contient des exécutable

Le répertoire **/var/log** qui contient des logs

Les droits des fichiers créés par utilisateurs , **rwxr** - le propriétaire peut lire écrire exécuter, le groupe associé peut lire exécuter, et toutes les utilisateurs peuvent lire , **executer**.

```
drwxr-xr-x@ 7 vladislavyaromiy staff 224 Oct 15 16:02 supplyChain-monorepo
drwxr-xr-x@ 9 vladislavyaromiy staff 288 Feb 14 08:51 sysDauphine
drwxr-xr-x 27 vladislavyaromiy staff 864 Feb 7 2023 theme-1
drwxr-xr-x 11 vladislavyaromiy staff 352 Mar 3 2023 theme-2-g-5-y-23
drwxr-xr-x 8 vladislavyaromiy staff 256 Mar 14 2023 theme3
drwxr-xr-x 11 vladislavyaromiy staff 352 May 22 2023 tp-js-s4-22-23
drwxr-xr-x 5 vladislavyaromiy staff 160 May 18 2023 tp2_tp3_yaromiy_mattot
drwxr-xr-x 52 vladislavyaromiy staff 1664 May 3 2023 tp_erreurs_yaromiy_mattot
drwxr-xr-x@ 7 vladislavyaromiy staff 224 Jan 8 2023 uni
drwxr-xr-x 15 vladislavyaromiy staff 480 Jul 8 2023 upselling
drwxr-xr-x@ 4 vladislavyaromiy staff 128 Nov 26 16:44 wordlists
```

Maintenant on va comparer avec les fichiers critiques du système:

```
drwxr-xr-x  27 root  wheel      804 Dec 15 15:43 pam.d
-rw-r--r--   1 root  wheel    8460 Dec 15 15:43 passwd
-rw-r--r--   1 root  wheel     75 Dec 15 15:43 paths
drwxr-xr-x   6 root  wheel     192 Jan 18 09:34 paths.d
```

Le propriétaire peut lire et modifier le fichier mais ne peut pas l'exécuter. Les membres du groupe et les autres utilisateurs peuvent uniquement lire le fichier et ne peuvent ni le modifier ni l'exécuter.

On remarque aussi que le fichier passwd est en mode read pour tous les utilisateurs par défaut et en tant qu'administrateur on pourra le modifier .

```
-r--r-----  1 root  wheel    1563 Dec 15 15:43 sudoers
```

Ce définit les règles permettant à certains utilisateurs ou groupes d'exécuter des commandes en tant que superutilisateur (root) ou un autre utilisateur en utilisant sudo , on remarque que le propriétaire du fichier a le droit de le lire uniquement. Les membres du groupe ont le droit de lire le fichier uniquement. Les autres utilisateurs (ni le propriétaire, ni les membres du groupe) n'ont aucun droit sur le fichier.

## Exercice 2 : Gestion des Comptes Utilisateurs

Créez un nouvel utilisateur avec des privilèges limités.

Configurez le groupe sudo pour autoriser cet utilisateur à exécuter des commandes avec des privilèges élevés.

Limitez l'utilisation du compte root aux tâches d'administration critiques.

### Correction:

Creation d'un user : **sudo useradd -m newuser**

-m : Cette option crée le répertoire personnel de l'utilisateur dans /home.

Création de mdp pour nouveau user: **sudo passwd newuser**

Par défaut, cet utilisateur ne disposera pas de droits d'administration. Pour donner des privilèges plus élevés on devra modifier le groupe sudo:

**sudo usermod -aG sudo newuser**

Pour limiter le root aux tâches d'administration critiques on peut utiliser les ACL:

**setfacl -m u:newuser:r-x fichier\_administratif**

Cette commande donne à newuser des droits de lecture et exécution mais pas d'écriture sur fichier\_administratif.

### Exercice 3 : Mises à Jour et Correctifs

Vérifiez la liste des mises à jour disponibles sur votre système.

Effectuez les mises à jour nécessaires en utilisant le gestionnaire de paquets approprié.

Vérifiez que les correctifs de sécurité sont bien appliqués.

#### Correction:

Cette commande met à jour la liste des paquets et versions disponibles en utilisant APT (Advanced Packaging Tool)

**sudo apt update**

Pour vérifier la liste des updates disponibles sans installation:

**apt list --upgradable**

Pour installer les mises à jours

**sudo apt upgrade**

### Exercice 4 : Configuration d'un Pare-feu

Utilisez iptables pour configurer une règle permettant le trafic entrant sur le port SSH.

Vérifiez que la règle est correctement appliquée en testant la connexion SSH depuis un autre ordinateur.

#### Correction:

Tout d'abord il faut configurer ssh et installer openssh-server sur votre serveur

Once you have installed an OpenSSH server,

```
sudo apt-get install openssh-server
```

you will need to configure it by editing the `sshd_config` file in the `/etc/ssh` directory.



`sshd_config` is the configuration file for the OpenSSH server. `ssh_config` is the configuration file for the OpenSSH client. Make sure not to get them mixed up.

First, make a backup of your `sshd_config` file by copying it to your home directory, or by

```
sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.factory-defaults
sudo chmod a-w /etc/ssh/sshd_config.factory-defaults
```

Creating a read-only backup in `/etc/ssh` means you'll always be able to find a known-good

Once you've backed up your `sshd_config` file, you can make changes with any text editor,

```
sudo gedit /etc/ssh/sshd_config
```

runs the standard text editor in Ubuntu 12.04 or more recent. For older versions replace "su" (see the suggestions in the rest of this page), you can apply them by saving the file then do

```
sudo restart ssh
```

If you get the error, "Unable to connect to Upstart", restart ssh with the following:

```
sudo systemctl restart ssh
```

**J'ai utilisé la connexion via mot de passe , pour cela on doit modifier `sshd_config` pour avoir:**

```
ubuntu
GNU nano 6.2 /etc/ssh/sshd_config

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 22
#AddressFamily any
ListenAddress 0.0.0.0
ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

[ Read 122 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^_ Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-E Redo
```

Après on doit permettre le trafic entrant sur le port 22 utilisé par SSH

**iptables -A INPUT -p tcp -i eth0 --dport ssh -j ACCEPT**

Pour vérifier que le règle a été bien appliqué :

```
# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination           ctstate ESTABLISHED
ACCEPT     all  --  anywhere              anywhere              tcp dpt:ssh
ACCEPT     tcp  --  anywhere              anywhere              tcp dpt:ssh
```

Il faut nécessairement sauvegarder les règles car la configuration est perdue lors de redémarrage

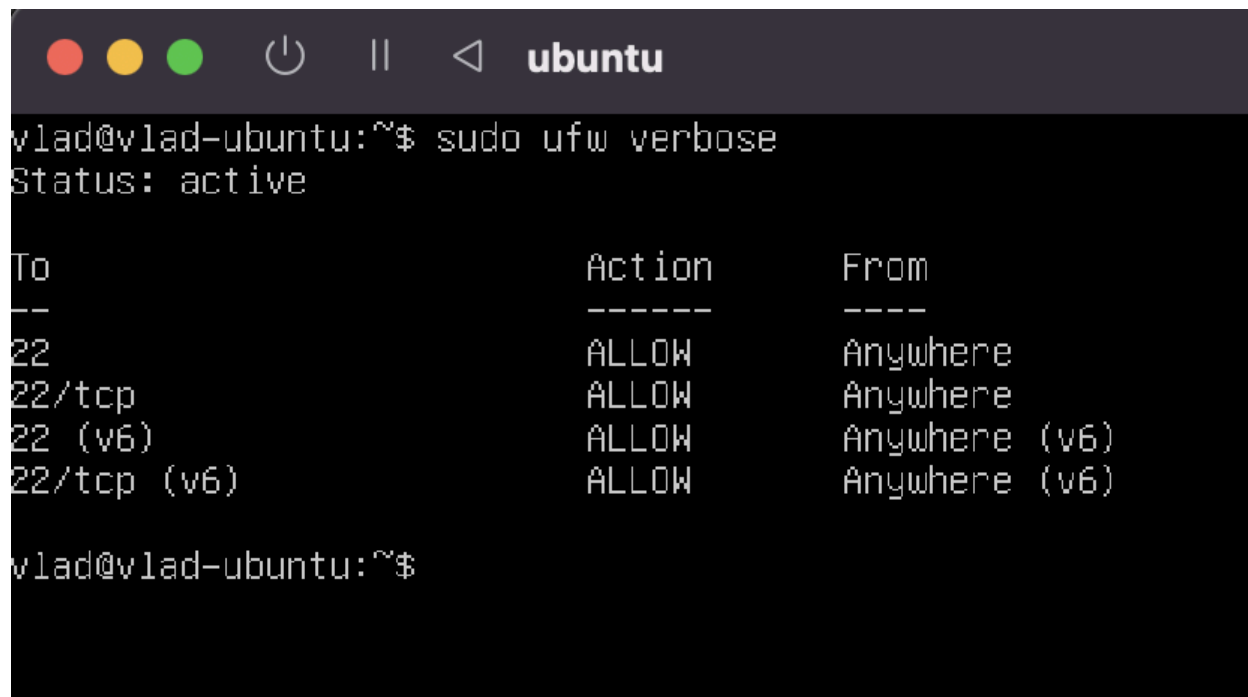
**sudo iptables-save > /etc/iptables/rules.v4**

**Après j'ai utilisé ufw qui est un système firewall qui facilite l'ouverture des connections (L'UFW s'appuie sur les iptables)**

**On fait:**

```
sudo ufw allow ssh
sudo ufw allow 22
sudo ufw enable
```

**Et finalement sudo ufw verbose pour voir les règles applique**



```
vlad@vlad-ubuntu:~$ sudo ufw verbose
Status: active

To Action From
--
22 ALLOW Anywhere
22/tcp ALLOW Anywhere
22 (v6) ALLOW Anywhere (v6)
22/tcp (v6) ALLOW Anywhere (v6)

vlad@vlad-ubuntu:~$
```

Pour se connecter depuis l'autre ordinateur:

**ssh user@ipaddress**

```
*** System restart required ***  
Last login: Wed Feb 14 14:19:18 2024  
vlad@vlad-ubuntu:~$ ls  
main.py  
vlad@vlad-ubuntu:~$ ls  
main.py  
vlad@vlad-ubuntu:~$ ls  
main.py  
vlad@vlad-ubuntu:~$ █
```

**On est connecté !**