**Hardening d’OS**

OS 🡪 >Ensemble de programmes :

* Shells (CLI/GUI)
* Kernel (noyau)
* Init (system)
* Bootlooder
* Drivers
* Gestionnaires de services
* Gestion de logs (journal -linux /gestionnaires d’évènements -windows)
* Heures

Lancer un programme « qu’est-ce qui se passe » :

* Depuis un dossier – Environnement d’exécution (runtime)
* Programme entièrement copié en mémoire ( RAM ) par l’OS
* Scheduling
* rwx( users : permission )

Différents espaces de stockages :

* Disque durs ( vitesse : 100m/s ) ( taille : 1to )
* Mémoires ( vitesse : 3g/s ) ( 16Go )
* Cache ( vitesse : 100g/s ) ( taille : 16Mo )
* Registre ( vitesse : cycle CPU GHz ) ( taille : 64b )

Hardening : Durcissement

* Users
* Pings
* Journaux

TP

Identifiant qui lance le programme 🡪 numéro que l’on voit

Part I : User Management

* Lister tous les utilisateurs créés sur la machine : **cut -d : -f1 /etc/passwd**
* Lister tous les groupes d’utilisateurs : **cut -d : -f1 /etc/group**
* Déterminer la liste des groupes dans lesquels se trouvent notre utilisateur : **groups nom-utilisateur**
* Lister les processus actuellement en cours d’exécution lancés par root : **ps -u root**
* Lister les processus actuellement en cours d’exécution lancés par un utilisateur : **ps -u nom\_utilisateur / ps -u $USER**
* Déterminer le Hash du mot de passe de root**: sudo cat /etc/shadow | grep root**
* Déterminer le Hash du mot de passe d’un utilisateur**: sudo cat /etc/shadow | grep «nom\_utillisateur »**
* Déterminer la fonction de hachage qui a été utilisé : SHA-512 cela dépend de la valeur trouver avec le hash précédemment ici j’ai $6$.
* Déterminer pour l’utilisateur root shell par défaut / chemin vers son répertoire personnel **: grep ‘^root’ /etc/passwd**



**/root** : Le répertoire personnel de root

**/bin/bash**: Le shell par défaut de root

* Déterminer pour notre utilisateur son shell par défaut / chemin vers son répertoire personnel **: grep ‘^LucasM’ /etc/passwd**



**/home/lucasM** : Le répertoire de mon utilisateur

**/bin/bash** : Le shell par défaut

* Accéder au fichier sudoers : **sudo visudo 🡪 LucasM ALL=(ALL) ALL**

**2. User Creation and Configuration**

**-** Créer un utilisateur :

* doit s'appeler meow
* ne doit appartenir QUE à un groupe nommé admins
* ne doit pas avoir de répertoire personnel utilisable
* ne doit pas avoir un shell utilisable

[LucasM@localhost ~]$ **sudo groupadd admins :** ajout groupe admin

[LucasM@localhost ~]$ **sudo useradd -G admins -M -s /usr/sbin/nologin meow :** permet de créer l’utilisateur meow dans le groupe admins sans répertoire personnel et shell utilisable

[LucasM@localhost ~]$ **groups meow :** affiche le groupe de notre utilisateur

meow : meow admins

[LucasM@localhost ~]$ **ls -l /home/meow** : Vérifier le répertoire personnel

ls: cannot access '/home/meow': No such file or directory

[LucasM@localhost ~]$ **grep '^meow' /etc/passwd** : Vérifier le shell de l’utilisateur

meow:x:1002:1003::/home/meow:/usr/sbin/nologin

- Configuration sudoers :

* ajouter une configuration sudoers pour que l'utilisateur meow puisse exécuter seulement et uniquement les commandes ls, cat, less et more en tant que votre utilisateur

[LucasM@localhost ~]$ **sudo su -s /bin/bash - meow**

Last login: Mon Feb 3 12:56:23 CET 2025 on pts/0

su: warning: cannot change directory to /home/meow: No such file or directory

[meow@localhost LucasM]$ sudo -u LucasM ls

[sudo] password for meow:

[meow@localhost LucasM]$ **sudo -u LucasM ls**

[meow@localhost LucasM]$ **sudo -u LucasM cat /etc/passwd**

root:x:0:0:root:/root:/bin/bash

bin:x:1:1:bin:/bin:/sbin/nologin

daemon:x:2:2:daemon:/sbin:/sbin/nologin

adm:x:3:4:adm:/var/adm:/sbin/nologin

lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin

sync:x:5:0:sync:/sbin:/bin/sync

shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown

halt:x:7:0:halt:/sbin:/sbin/halt

mail:x:8:12:mail:/var/spool/mail:/sbin/nologin

operator:x:11:0:operator:/root:/sbin/nologin

games:x:12:100:games:/usr/games:/sbin/nologin

ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin

nobody:x:65534:65534:Kernel Overflow User:/:/sbin/nologin

systemd-coredump:x:999:997:systemd Core Dumper:/:/sbin/nologin

dbus:x:81:81:System message bus:/:/sbin/nologin

tss:x:59:59:Account used for TPM access:/:/usr/sbin/nologin

sssd:x:998:996:User for sssd:/:/sbin/nologin

sshd:x:74:74:Privilege-separated SSH:/usr/share/empty.sshd:/usr/sbin/nologin

chrony:x:997:995:chrony system user:/var/lib/chrony:/sbin/nologin

lucasM:x:1000:1000::/home/lucasM:/bin/bash

LucasM:x:1001:1001::/home/LucasM:/bin/bash

meow:x:1002:1003::/home/meow:/usr/sbin/nologin

[meow@localhost LucasM]$ **sudo -u LucasM less /var/log/syslog**

/var/log/syslog: No such file or directory

[meow@localhost LucasM]$ **sudo -u LucasM more /etc/hostname**

* ajouter une configuration sudoers pour que les membres du groupe admins puisse exécuter seulement et uniquement la commande apt en tant que root

commande ajoutée au fichier visudo : **%admins ALL=(ALL) NOPASSWD: /usr/bin/dnf**

[meow@localhost LucasM]$ **sudo dnf install**

usage: dnf install [-c [config file]] [-q] [-v] [--version] [--installroot [path]] [--nodocs]

[--enableplugin [plugin]] [--disableplugin [plugin]] [--releasever RELEASE

[--setopt SETOPTS] [--skip-broken] [-h] [--allowerasing] [-b | --nobest] [

[-R [minutes]] [-d [debug level]] [--debugsolver] [--showduplicates] [-e E

[--obsoletes] [--rpmverbosity [debug level name]] [-y] [--assumeno] [--ena

[--disablerepo [repo] | --repo [repo]] [--enable | --disable] [-x [package

[--disableexcludes [repo]] [--repofrompath [repo,path]] [--noautoremove] [

[--color COLOR] [--refresh] [-4] [-6] [--destdir DESTDIR] [--downloadonly]

[--comment COMMENT] [--bugfix] [--enhancement] [--newpackage] [--security]

[--advisory ADVISORY] [--bz BUGZILLA] [--cve CVES]

[--sec-severity {Critical,Important,Moderate,Low}] [--forcearch ARCH]

PACKAGE [PACKAGE ...]

dnf install: error: the following arguments are required: PACKAGE

* ajouter une configuration sudoers pour que votre utilisateur puisse exécuter n'importe quel commande en tant root, sans avoir besoin de saisir un mot de passe

commande insérée dans le fichier visudo : **LucasM ALL=(ALL) ALL**

* prouvez que ces 3 configurations ont pris effet (vous devez vous authentifier avec le bon utilisateur, et faire une commande sudo qui doit fonctioner correctement)

Part II : Files and permissions

1. **Listing POSIX permissions**

* Fichier qui contient la liste des utilisateurs :

[LucasM@localhost ~]$ **ls -l /etc/passwd**

-rw-r--r--. 1 root root 1025 Feb 3 11:19 /etc/passwd

* Fichier qui contient la liste des hashes des mots de passes des utilisateurs :

[LucasM@localhost ~]$ **ls -l /etc/shadow**

----------. 1 root root 872 Feb 3 12:56 /etc/shadow

* Fichier de configuration du serveur OpenSSH :

[LucasM@localhost ~]$ **ls -l /etc/ssh/sshd\_config**

-rw-------. 1 root root 3667 Apr 18 2024 /etc/ssh/sshd\_config

* Répertoire personnel de l’utilisateur root :

[LucasM@localhost ~]$ **ls -ld /root**

dr-xr-x---. 3 root root 147 Feb 3 13:46 /root

* Répertoire personnel de mon utilisateur :

[LucasM@localhost ~]$ **ls -ld /home/LucasM**

drwx------. 3 LucasM LucasM 98 Feb 3 13:46 /home/LucasM

* Le programme systemctl :

[LucasM@localhost ~]$ **ls -l $(which systemctl)**

-rwxr-xr-x. 1 root root 305680 Apr 8 2024 **/usr/bin/systemctl**

* Le programme ls :

[LucasM@localhost ~]$ **ls -l $(which ls)**

-rwxr-xr-x. 1 root root 140872 Apr 20 2024 **/usr/bin/ls**

1. **Extended attributes**

* Lister tous les programmes qui ont le bit SUID activé :

[LucasM@localhost ~]$ **find / -perm -4000 -type f 2>/dev/null**

/usr/bin/chage

/usr/bin/gpasswd

/usr/bin/newgrp

/usr/bin/mount

/usr/bin/umount

/usr/bin/su

/usr/bin/crontab

/usr/bin/passwd

/usr/bin/sudo

/usr/sbin/unix\_chkpwd

/usr/sbin/pam\_timestamp\_check

/usr/sbin/grub2-set-bootflag

* Rendre le fichier de configuration du serveur OpenSSH immuable :

[root@localhost ~]# **chattr +i /etc/ssh/sshd\_config**

Vérification

[root@localhost ~]# **lsattr /etc/ssh/sshd\_config**

**----i----------------- /etc/ssh/sshd\_config**

**nano /etc/ssh/sshd\_config**

**Une image contenant texte, Appareils électroniques, capture d’écran, Police

Description générée automatiquement**

1. **Protect a file using permissions**

[root@localhost ~]# su - LucasM

Last login: Mon Feb 3 15:59:11 CET 2025 on pts/1

[LucasM@localhost ~]$ **mkdir ~/public\_folder**

[LucasM@localhost ~]$ **chmod 777 ~/public\_folder**

[LucasM@localhost ~]$ **echo "Fichier Secret" > ~/public\_folder/dont\_readme.txt**

[LucasM@localhost ~]$ **chmod 600 ~/public\_folder/dont\_readme.txt**

[LucasM@localhost ~]$ **cat ~/public\_folder/dont\_readme.txt**

Fichier Secret 🡪 mon utilisateur peut lire

[LucasM@localhost ~]$ **echo "Nouveau ligne add" >> ~/public\_folder/dont\_readme.txt**

[LucasM@localhost ~]$ **cat ~/public\_folder/dont\_readme.txt**

Fichier Secret

Nouveau ligne add

**MEOW ne peut pas y toucher**

[LucasM@localhost ~]$ **sudo -u meow cat ~/public\_folder/dont\_readme.txt**

[sudo] password for LucasM:

cat: /home/LucasM/public\_folder/dont\_readme.txt: **Permission denied**

**ROOT peut y toucher**

[LucasM@localhost ~]$ **sudo cat ~/public\_folder/dont\_readme.txt**

Fichier Secret

Nouveau ligne add

Part III. Networking

* 1. Listenning ports
* Déterminer les programmes qui écoutent sur TCP :

[LucasM@localhost ~]$ **ss -tlnp**

State Recv-Q Send-Q Local Address:Port Peer Address:Port Process

LISTEN 0 128 0.0.0.0:22 0.0.0.0:\*

LISTEN 0 128 [::]:22 [::]:\*

Deux processus écoutent sur le port TCP 22 :

* **0.0.0.0 :22** 🡪 Accepte les connexions sur toutes les interfaces IPV4
* **[ ::]** 🡪Accepte toutes les connexions sur toutes les interfaces IPV6
* Déterminer les programmes qui écoutent sur UDP :

[LucasM@localhost ~]$ **ss -ulnp**

State Recv-Q Send-Q Local Address:Port Peer Address:Port Process

UNCONN 0 0 127.0.0.1:323 0.0.0.0:\*

UNCONN 0 0 [::1]:323 [::]:\*

Deux processus écoutent sur des ports UDP :

* **127.0.0.1 :323** en (IPV4)
* **[ ::1] :323** en (IPV6)

Fermer tout les ports à part ssh :

**sudo firewall-cmd --zone=public --remove-service=dhcpv6-client**

**sudo firewall-cmd --zone=public --remove-service=cockpit**

**sudo firewall-cmd --reload**

Part IV. Storage and Partition

1. Existing partitions

* Déterminer la liste des partitions du système :

[LucasM@localhost ~]$ **sudo fdisk -l**

Disk /dev/nvme0n1: 20 GiB, 21474836480 bytes, 41943040 sectors

Disk model: VMware Virtual NVMe Disk

Units: sectors of 1 \* 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disklabel type: dos

Disk identifier: 0xd699b39e

Device Boot Start End Sectors Size Id Type

/dev/nvme0n1p1 \* 2048 2099199 2097152 1G 83 Linux

/dev/nvme0n1p2 2099200 41943039 39843840 19G 8e Linux LVM

Disk /dev/mapper/rl-root: 17 GiB, 18249416704 bytes, 35643392 sectors

Units: sectors of 1 \* 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/rl-swap: 2 GiB, 2147483648 bytes, 4194304 sectors

Units: sectors of 1 \* 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

* Identifier la partition qui est montée sur /

[LucasM@localhost ~]$ **findmnt /**

TARGET SOURCE FSTYPE OPTIONS

/ **/dev/mapper/rl-root** xfs rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,n

1. Mount Options

* Déterminer les options de montage de la partition :

[LucasM@localhost ~]$ **mount | grep ' on / '**

/dev/mapper/rl-root on / type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)

Explication :

* **rw** : Le système de fichiers est monté en mode lecture-écriture.
* **relatime** : Le système met à jour les timestamps d'accès uniquement si cela est nécessaire, afin d'améliorer les performances.
* **seclabel** : Sécurité SELinux activée, ce qui permet l'utilisation des étiquettes de sécurité.
* **attr2** : Utilisation du format d'attributs étendus 2, améliorant la gestion des métadonnées.
* **inode64** : Support des inodes 64 bits pour une plus grande capacité d'adressage des fichiers.
* **logbufs=8** : Le nombre de tampons de journalisation utilisés est 8. Cela affecte les performances et la gestion du journal de système de fichiers.
* **ogbsize=32k** : Taille de chaque tampon de journalisation, ici 32 Ko.
* **noquota** : Les quotas de disque ne sont pas activés pour cette partition.
* Monter une partition de type tmpfs sur le dossier /tmp

[LucasM@localhost ~]$ **sudo mount -t tmpfs -o noexec tmpfs /tmp**

[LucasM@localhost ~]$ **mount | grep /tmp**

tmpfs on /tmp type tmpfs (rw,noexec,relatime,seclabel,inode64)

[LucasM@localhost ~]$ **echo -e '#!/bin/bash\n echo TP Hardening OS !' > /tmp/hardening.sh**

[LucasM@localhost ~]$ **chmod 777 /tmp/hardening.sh**

[LucasM@localhost ~]$ **/tmp/hardening.sh**

-bash: /tmp/hardening.sh: **Permission denied**

Part V. OpenSSH Server

1. Basics

* afficher l’identifiant du processus serveur openSSH en cours d’execution :

[LucasM@localhost ~]$ **ps aux | grep '[s]shd'**

root 755 0.0 0.2 15860 9216 ? Ss 12:34 0:00 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

root 1367 0.0 0.3 19248 12032 ? Ss 12:38 0:00 sshd: LucasM [priv]

LucasM 1381 0.0 0.1 19444 7156 ? R 12:38 0:00 sshd: LucasM@pts/0

PID du serveur : 755

* Changer le port d’écoute du serveur OpenSSH :

Dans le fichier **/etc/ssh/sshd\_config**

Port : 2231

Configuration d'une authentification par clé :

***ssh-keygen***

***ssh-copy-id -i ~/.ssh/id\_*ed25519.pub -p 2341 username@server*\_ip***

Fichier conf du serveur***ssh /etc/ssh/sshd\_*config**

**PubKeyAuthentication yes**

Désactiver la connexion par password :

Fichier **/etc/ssh/sshd*\_config***

***PasswordAuthentication no***

Désactiver la connexion root *:*

Fichier ***/etc/ssh/sshd\_*config**

**PermitRootLogin no**

Mise en place d'une connexion SSH via certificat :

Génération de la clé SSH du serveur

**ssh-keygen -t rsa -b 4096 -f /etc/ssh/ssh*\_host\_*rsa*\_key -N ""***

*Génération de la paire de clés du CA hôte :*

***ssh-keygen -t rsa -b 4096 -f ~/host\_*ca -N ""**

Signature de la clé hôte avec le CA :

**sudo ssh-keygen -s ~/host*\_ca -I rocky\_*tp -h -n localhost -V +52w /etc/ssh/ssh*\_host\_*rsa*\_key.pub***

Déplacement et configuration du CA hôte sur le serveur :

***sudo mkdir -p /etc/ssh/certs***

***sudo mv ~/host\_*ca.pub /etc/ssh/certs/**

**sudo chmod 644 /etc/ssh/certs/host*\_ca.pub***

Ajout de la configuration du serveur SSH :

***HostCertificate /etc/ssh/ssh\_*host*\_rsa\_*key-cert.pub**

**TrustedUserCAKeys /etc/ssh/certs/host*\_ca.pub***

Redémarrage du service SSH :

***sudo systemctl restart ssh***

Copie de la clé publique du CA vers le client :

***scp -P 2222 /etc/ssh/certs/host\_*ca.pub LucasM@localhost:/tmp/**

Déplacement et permission de la clé CA sur le client

**mv /tmp/host*\_ca.pub ~/.ssh/***

***chmod 644 ~/.ssh/host\_*ca.pub**

Ajout du CA aux hôtes connus du client

**echo "@cert-authority \* $(cat ~/.ssh/host*\_ca.pub)" >> ~/.ssh/known\_*hosts**

Test de la connexion SSH

**ssh -p 2222 LucasM@localhost**

Idées pour sécuriser le serveur SSH

- Mise en place d'une MFA

- Mettre en place une whitelist pour les postes distants autorisés à se connecter au serveur

- Limiter l'accès au système de fichiers pour les utilisateurs