-D'abord j'ai decide de créer un utilisateur qui peut exécuter n'importe qu'elle commande comme le root sans être le root , ceci dans le but de donner une certaine flexibilité que ce soit pour la creation d'utilisateurs ou autre

→ pour ce faire j'ai modifie le fichier sudoers via visudo :

```
root@arashi-VirtualBox:/home/arashi# visudo
```

Puis j'ai ajoute la ligne suivante :

```
root    ALL=(ALL:ALL) ALL
arashi  ALL=(ALL:ALL) ALL
```

➔ J'aurai pu aussi ajouter arashi (c'est mon compte utilisateur de la VM) dans le group admin ou dans le groupe sudoers ayant tous les droits mais j'ai prefere cette approche pour ne pas me compliquer la vie (après par mesure sécurité bien sur je restreindrai l'acces a cette utilisateur)

Creation de l'utilisateur maintenant :

➔ Je cree son repertoire de travail d'abord :

```
arashi@arashi-VirtualBox:~$ sudo mkdir /home/user1
```

➔ J'ajoute le nouvel utilisateur :

➔
```
arashi@arashi-VirtualBox:~$ sudo useradd -d /home/user1 -s /bin/bash user1
```

➔ On lui affecte son password :

```
arashi@arashi-VirtualBox:~$ passwd user1
```

➔ Puis on se connecte avec cet utilisateur :

```
user1@arashi-VirtualBox:~$
```

➔ Lorsque l'on ouvre /etc/passwd on nous souligne que le fichier est unwritable :

```
[ File '/etc/passwd' is unwritable ]
```

➔ Je reviens a mon compte arashi qui peut exécuter n'importe quelle commande via sudo et je modifie le fichier etc/passwd puis j'enregistre :

```
user1:x:1001:1003::/home/user1:/sbin/nologin
```

➔ Lorsque je reessaye de me connecter voici ce qui s'affiche :

```
arashi@arashi-VirtualBox:~$ su - user1
Password:
This account is currently not available.
```

On conlut donc qu'on modifiant le fichier /etc/passwd et plus précisément la ligne de l'utilisateur cible , on peut lui blocker l'acces au compte (le shell defini est un nologin)

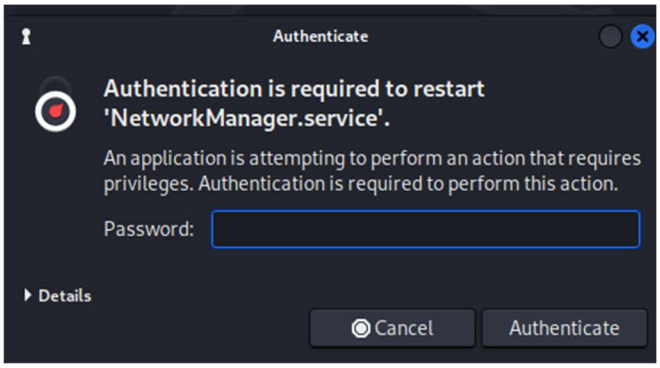
SSH :

➔ Le numero de port ssh est de 22 (c'est le port par defaut mais on peut toujours changer ceci soit par des traduction via le pat et le port forwarding autre par mesure de sécurité)

(j'ai change de machine virtuelle entre temps suite a des problèmes de reseau)

1-2 – Restriction du Root :

➢ Le service réseau, dans les systèmes d'exploitation utilisant Systemd, est généralement géré par le service NetworkManager , avec l'utilisateur user1 je vais essayer de redémarrer ce service :



On me dit la que cette action requiert les privileges du root et que je dois m'authentifier en temps que root pour pouvoir redémarrer le service reseau

➢ Lorsque j'essaie de faire la même action mais cette fois ci avec sudo la , j'ai le message suivant qui indique que mon user1 ne figure pas dans le groupe sudoers pour pouvoir réaliser cette action avec les permissions du root :

```
user1@Arashi:~$ sudo systemctl restart NetworkManager
[sudo] password for user1:
user1 is not in the sudoers file.
```

➢ Donc pour pouvoir redemarer le service avec sudo , il faut d'abord modifier le fichier de configuration /etc/sudoers avec la commande visudo et ceci en tant que root bien sur car ce fichier n'est accessible qu'au root (ou par un tiers utilisateurs dans sudoers et pouvant excercer cette commande)
1er cas : donner user1 uniquement le droit de performer la commande systemctl restart NetworkManager en tant que root , on ajoute la ligne suivant dans /etc/sudoers :

```
user1   ALL=(ALL:ALL) /bin/systemctl restart NetworkManager
```

Voyons ci ca marche maintenant :

```
user1@Arashi:~$ sudo systemctl restart NetworkManager
[sudo] password for user1:
user1@Arashi:~$
```

C'est bon

- 2eme cas : donner au user 1 tous les droits du root comme on l'a déjà fait pour le compte Arashi au début , mais ce n'est pas securise

1-3 Securite des mots de passe :

- On compare les lignes du root est de user1 dans /etc/passwd :

```
root:x:0:0:root:/root:/usr/bin/zsh
nm-openvpn:x:129:131:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
user1:x:1001:1001::/home/user1:/bin/bash
```

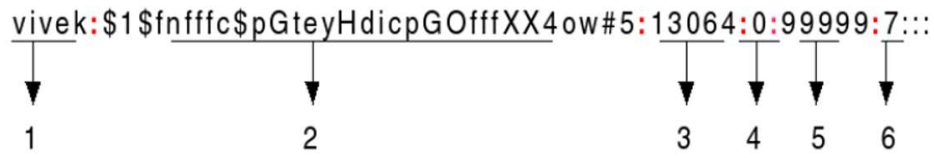
On voit qu'il y'a un «x » dans le champ password (chose qui est normal car les mots de passe des utilisateurs ont été deplace du fichier /etc/password vers /etc/shadow ou ils y sont stockes sous forme de HASH

- Dans /etc/shadow on voit les lignes suivantes :

```
root:$y$j9T$/joqe8uzLGhJd2fDDh/.G0$Mv596zS3MPNm7tU83R8JYKJxm3uK9QrWzyeMRvsScv9:20214:0:99999:7:::
user1:$y$j9T$Ee3KvkEuyA9o6ZP/4kCuu/$de8pn7NI3mQPVJdPFcp.9/p7eCdWf4Tjls5FFBO1MQ6:20214:0:99999:7:::
```

Voila comment on explique les champs :

Exemple:



Format :

1. **nom_de_utilisateur**
2. **mot_de_passe:**
3. **derniere_modification_MotDePasse**
4. **minimum de jours requis entre les changements de mot de passe**
5. **maximum de jours pendant lesquels le mot de passe est valide**
6. **nombre de jours avant l'expiration du mot de passe**
7. **nombre de jours après l'expiration du mot de passe**
8. **date d'expiration du compte**

- Maintenant on va modifier le mdp du user1 pour voir le changement de sa ligne dans /etc/shadow :

```
user1:$y$j9T$n96g4dBZ/wNq17qs6Ltcx.$3To/oBAElR.P62rk4ulk3QU7MTQO2dkC.TCIzgLTqe2:20214:0:99999:7:::
user1:$y$j9T$Ee3KvkEuyA9o6ZP/4kCuu/$de8pn7NI3mQPVJdPFcp.9/p7eCdWf4Tjls5FFBO1MQ6:20214:0:99999:7:::
```

On observe bien la difference du hash du mdp

- On modifie maintenant la date de validite

```
user1:$y$j9T$n96g4dBZ/wNq17qs6Ltcx.$3To/oBAElR.P62rk4ulk3QU7MTQO2dkC.TCIzgLTqe2:20214:0:99999:7:::0:
```

2-3 Expiration de mot de passe :

- On modifie la date de validite du mdp du user1 et on voit le changement au niveau de sa ligne dans /etc/shadow :

```
┌──(root㉿Arashi)-[~user1]
└─# chage -M 90 "user1"

┌──(root㉿Arashi)-[~user1]
└─# grep -E "user1" /etc/shadow
user1:$y$j9T$n96g4dBZ/wNq17qs6Ltcx.$3To/oBAElR.P62rk4ulk3QU7MTQO2dkC.TCIzgLTqe2:20214:0:90:7:::0:
```

2-2-Utilisation de l'utilitaire « John The Ripper »

- On va créer le fichier de mdp password.txt avec la commande unshadow ayant pour paramètres les fichier /etc/passwd et /etc/shadow :

```
┌──(arashi㉿Arashi)-[~]
└─$ sudosudo unshadow /etc/passwd /etc/shadow > password.txt
```

Voici comment est organise le fichier password.txt

```
_rpc:!:110:65534::/run/rpcbind:/usr/sbin/nologin
Debian-snmp:!:111:109::/var/lib/snmp:/bin/false
redis:!:112:111::/var/lib/redis:/usr/sbin/nologin
usbmux:!:113:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
mosquitto:!:114:114::/var/lib/mosquitto:/usr/sbin/nologin
redsocks:!:115:115::/var/run/redsocks:/usr/sbin/nologin
stunnel4:!*:991:991:stunnel service system account:/var/run/stunnel4:/usr/sbin/nologin
sshd:!:116:65534::/run/sshd:/usr/sbin/nologin
dnsmasq:!:999:65534:dnsmasq:/var/lib/misc:/usr/sbin/nologin
sslh:!:117:118::/nonexistent:/usr/sbin/nologin
postgres:!:118:119:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
avahi:!:119:120:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
_gvm:!:120:122::/var/lib/openvas:/usr/sbin/nologin
speech-dispatcher:!:121:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/false
inetsim:!:122:124::/var/lib/inetsim:/usr/sbin/nologin
geoclue:!:123:125::/var/lib/geoclue:/usr/sbin/nologin
lightdm:!:124:126:Light Display Manager:/var/lib/lightdm:/bin/false
statd:!:125:65534::/var/lib/nfs:/usr/sbin/nologin
saned:!:126:128::/var/lib/saned:/usr/sbin/nologin
polkitd:!*:989:989:User for polkitd:/:/usr/sbin/nologin
rtkit:!:127:129:RealtimeKit,,,:/proc:/usr/sbin/nologin
colord:!:128:130:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
nm-openvpn:!:129:131:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
nm-openconnect:!:130:132:NetworkManager OpenConnect plugin,,,:/var/lib/NetworkManager:/usr/sbin/nologin
arashi:$y$j9T$prwKEzorsBQWnRYaGBino/$H2BX2/fgax.JzbQjfKbxKx78vTlbflUG9pJqk6/An3/:1000:1000:Arashi,,,:/home/arashi:/usr/bin/zsh
user1:$y$j9T$n96g4dBZ/wNq17qs6Ltcx.$3To/oBAElR.P62rk4ulk3QU7MTQO2dkC.TCIzgLTqe2:1001:1001::/home/user1:/bin/bash
```

- On lance maintenant la commande l'utilitaire john the ripper pour cracker les mots de passes de password.txt avec comme parametre la wordlists rockyou connu pour ce type de tache :

```
└─$ john password.txt /usr/share/wordlists/rockyou.txt
Warning: only loading hashes of type "tripcode", but also saw type "descrypt"
Use the "--format=descrypt" option to force loading hashes of that type instead
Warning: only loading hashes of type "tripcode", but also saw type "pix-md5"
Use the "--format=pix-md5" option to force loading hashes of that type instead
Warning: only loading hashes of type "tripcode", but also saw type "mysql"
Use the "--format=mysql" option to force loading hashes of that type instead
Warning: only loading hashes of type "tripcode", but also saw type "oracle"
Use the "--format=oracle" option to force loading hashes of that type instead
Warning: only loading hashes of type "tripcode", but also saw type "Raw-SHA1"
Use the "--format=Raw-SHA1" option to force loading hashes of that type instead
Warning: only loading hashes of type "tripcode", but also saw type "LM"
Use the "--format=LM" option to force loading hashes of that type instead
Warning: only loading hashes of type "tripcode", but also saw type "Raw-SHA1-AxCrypt"
Use the "--format=Raw-SHA1-AxCrypt" option to force loading hashes of that type instead
Warning: only loading hashes of type "tripcode", but also saw type "bfegg"
Use the "--format=bfegg" option to force loading hashes of that type instead
Warning: invalid UTF-8 seen reading /usr/share/wordlists/rockyou.txt
Warning: only loading hashes of type "tripcode", but also saw type "dynamic=md5($p)"
Use the "--format=dynamic=md5($p)" option to force loading hashes of that type instead
Warning: only loading hashes of type "tripcode", but also saw type "cryptoSafe"
Use the "--format=cryptoSafe" option to force loading hashes of that type instead
```

- Malheureusement ou plutôt heureusement aucun mdp n'a pu être cracke par john the ripper :

```
└─$ john --show password.txt
0 password hashes cracked, 0 left
```

3-Desactivation des services inutiles

3-1 Verification des services reseau actifs

- On identifie tous les ports actifs :

```
└─$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp        0      0 10.0.2.15:bootpc        10.0.2.2:bootps         ESTABLISHED
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type       State         I-Node   Path
unix  3      [ ]         STREAM     CONNECTED     10667
unix  3      [ ]         STREAM     CONNECTED     8120
unix  3      [ ]         STREAM     CONNECTED     9604     @/tmp/.X11-unix/X0
unix  3      [ ]         STREAM     CONNECTED     10400
unix  3      [ ]         STREAM     CONNECTED     8894     /run/systemd/journal/stdout
unix  3      [ ]         STREAM     CONNECTED     8886
unix  2      [ ]         DGRAM      CONNECTED     163706
unix  3      [ ]         STREAM     CONNECTED     10616
unix  3      [ ]         STREAM     CONNECTED     9693     /run/user/1000/bus
unix  3      [ ]         STREAM     CONNECTED     26144
unix  3      [ ]         STREAM     CONNECTED     10699    /run/user/1000/bus
unix  3      [ ]         STREAM     CONNECTED     8938     /run/systemd/journal/stdout
unix  3      [ ]         STREAM     CONNECTED     9603
```

- Identification des ports TCP/UDP ouverts :

```
└─$ netsnetstat -atp

(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
```

```
└─$ netsnetstat -aup
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
udp        0      0 10.0.2.15:bootpc        10.0.2.2:bootps         ESTABLISHED -
```

- On observe les correspondances Port< - - > Service

```
└─$ sudo cat /etc/services
# Network services, Internet style
#
# Updated from https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml .
#
# New ports will be added on request if they have been officially assigned
# by IANA and used in the real-world or are needed by a debian package.
# If you need a huge list of used numbers please install the nmap package.

tcpmux          1/tcp                           # TCP port service multiplexer
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
```

- On liste les ports ouverts et leurs services correspondants :

```
┌──(arashi㉿Arashi)-[~]
└─$ sudosudo netstat -tulnp

Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name

┌──(arashi㉿Arashi)-[~]
```

Et si j'accede a mon navigateur web par exemple (on est cense avoir le port 443 avec https qui doit apparaitre)

```
 will not be shown, you would have to be root to see it all.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 10.0.2.15:44380         mrs08s20-in-f3.1e1:http ESTABLISHED 88624/firefox-esr
tcp        0      0 10.0.2.15:41792         adsl-235-28-192-81:http ESTABLISHED 88624/firefox-esr
```

Et voila c'est bien ce qu'on attendais (44380 les port respective de https et http

3-2 Verification des services demares :

- Pour lister tous les services et leur état d'activation au démarrage, utilise :

```
└─$ systsystemctl list-unit-files --type=service

UNIT FILE                                  STATE           PRESET
accounts-daemon.service                    enabled         enabled
apache-htcacheclean.service                disabled        disabled
apache-htcacheclean@.service               disabled        disabled
apache2.service                            disabled        disabled
apache2@.service                           disabled        disabled
apparmor.service                           disabled        disabled
apt-daily-upgrade.service                  static          -
apt-daily.service                          static          -
atftpd.service                             indirect        disabled
auth-rpcgss-module.service                 static          -
autovt@.service                            alias           -
avahi-daemon.service                       disabled        disabled
blueman-mechanism.service                  disabled        disabled
bluetooth.service                          disabled        disabled
capsule@.service                           static          -
colord.service                             static          -
configure-printer@.service                 static          -
console-getty.service                      disabled        disabled
console-setup.service                      enabled         enabled
container-getty@.service                   static          -
cron.service                               enabled         enabled
```