TD1

Les exercices ci-dessous sont à réaliser dans votre clone de la machine virtuelle vm-passoire.

2. Mots de passe

Cet exercice permet de comprendre la vulnérabilité que représente un mot de passe trop simple.

- Utiliser unshadow pour préparer un fichier mypasswd : sudo unshadow /etc/passwd /etc/shadow > mypasswd
- Lancer john sur le fichier mypasswd. Il est possible de restreindre la recherche à l'utilisateur etu via l'option -users:etu.
- Retrouver les fichiers mis à jour dans le compte etu ; expliquer les droits des nouveaux fichiers.
- Ajouter l'utilisateur boule, de mot-de-passe bill. Relancer john pour qu'il retrouve le mot de passe de boule tout en mesurant le temps nécessaire.
- Sur passoire, vérifier les droits des fichiers /etc/passwd et /etc/shadow. Corriger-les ci-besoin.
- Changer le mot de passe de l'utilisateur etu (retenir le mot de passe choisi !).

3. Configuration des droits : utilisateurs

Cet exercice permet de comprendre les droits des fichiers et des groupes.

decimal	droit
1	--x
2	-w-
3	-wx
4	r--
5	r-x
6	rw-
7	rwx
4000	suid
2000	guid
1000	sticky-bit

u —— user g —— group o —— other

- Créer deux utilisateurs remus et romulus sur passoire. On suppose que leurs répertoires courants sont /home/remus et /home/romulus.
- En tant qu'utilisateur remus, créer un fichier update.sh dans /home/remus, contenant l'instruction suivante : date >> /home/remus/log.txt
- Ajouter les droits d'exécution à ce fichier.

```
chmod +x <filename>
```

- Lancer en tant que remus update.sh. Que constatez-vous ?
- Adapter les droits des fichiers update.sh et log.txt de sorte que l'utilisateur romulus puisse également exécuter update.sh.

- Vérifier en vous connectant en tant que romulus et en lançant la commande udpdate.sh de remus.
- En tant que remus, enlever les droits de lecture, d'écriture et d'exécution pour 'other' sur le fichier log.txt. Refaire les étapes 4 et 6. Que constatez-vous ?

- En tant que etu, créer un groupe rome et y ajouter les utilisateurs remus et romulus.
- En tant que remus, changer les groupes des fichiers update.sh et log.txt de sorte qu'ils appartiennent au groupe rome.
- Recommencer les étapes 4 et 6. Que constatez-vous ?
- Faire en sorte que tous les fichiers dorénavant créés par remus appartiennent au groupe rome. Tester à nouveau l'étape 6.
- Faire en sorte que ce comportement soit le comportement par défaut des utilisateurs remus et romulus. ❓

4. Configuration des droits : setuid

Cet exercice permet de comprendre le droit spécial dit *setuid* et de corriger la machine virtuelle passoire.

- Comparer les droits des fichiers /usr/local/bin/readfile et /usr/local/bin/readfile-s. Le code de ce programme peut être retrouvé sur la page précédente.

```
stat /usr/local/bin/readfile /usr/local/bin/readfile-s | grep Access
```

- Utiliser ces scripts pour visualiser les fichiers /etc/passwd, /etc/shadow, /etc/group, /etc/gshadow. Expliquer les droits de ces fichiers et les résultats obtenus.

	/etc/passwd	/etc/shadow	/etc/group	/etc/gshadow
/usr/local/bin/readfile	oui	non	oui	non
/usr/local/bin/readfile-s	oui	oui	oui	oui

cat

https://linuxconfig.org/how-to-use-special-permissions-the-setuid-setgid-and-sticky-bits

http://www.linux-france.org/article/memo/node19.html#461

- Sur la machine virtuelle passoire, ajouter le setuid bit au script update.sh de l'utilisateur remus (cf. exercice 3). En tant que utilisateur etu, lancer le script. Que constatez-vous ? Pourquoi ?
- Expliquer les commandes suivantes : on pourra s'aider de man 2 stat (et du fichier /usr/include/linux/stat.h)
- - find / -user root -perm /u+s 2> /dev/null

    chercher à partir de la racine? —— /

    find the file whose owner is root —— -user root

    find the file with setuid

    2> /dev/null —— make stdeer redirect to /dev/null

    On peut rediriger les messages d'erreur vers le ``trou noir'' (le périphérique /dev/null)

    ❓ -perm /u+s pourquoi on ajoute « /«
- - find / -type f \ (-perm -2000 -o -perm -4000 \ ) -print 2 > /dev/null

    - \ (-perm -2000 -o -perm -4000 \ ) —— find the file with setuid or setgid

    - 2000 —— setgid only for floder, in this case we have indicated that it's file (-type f)
- - find / -type f -perm /u-s -perm /g+s -print 2> /dev/null

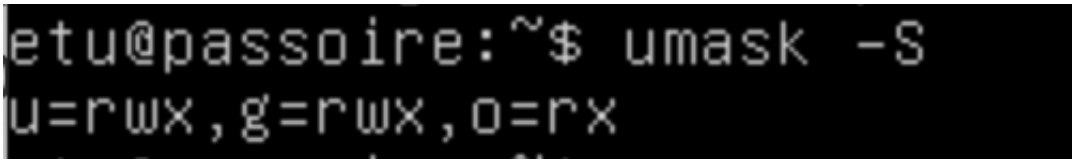
    - find the file with setgid but without setuid
- Estimer l'intérêt de ce bit selon les exécutables trouvés et au regard de l'usage de la machine. Enlever le *setuid bit* aux commandes qui ne sont pas utiles dans notre contexte.

5. Configuration des droits : umask

Cet exercice permet de comprendre les droits par défaut des fichiers et de corriger la machine virtuelle passoire.

- Afficher le masque par défaut. Créer un fichier (avec la commande touch) et constater les droits qui lui sont attribués.

```
1 umask -S #check default rights while user creates a new file
```



```
etu@passoire:~$ cd /home/etu/
etu@passoire:~$ touch test_umask
etu@passoire:~$ ls -l
total 8
drwxrwxrwx 2 etu etu 4096 Sep 17 15:23 bin
-rw-rw-r-- 1 etu etu 2140 Sep 19 14:13 mypasswd
-rw-rw-r-- 1 etu etu    0 Oct  1 09:27 test_umask
```

Il semble que le droit "x" ne peut être mis pour un nouveau fichier.

- Changer la valeur du masque de sorte que les fichiers créés aient aucun droits pour le groupe ni les autres utilisateurs. Vérifier en créant un fichier temporaire.

```
umask 077 //aucun droits pour le groupe ni les autres utilisateurs
```

```
etu@passoire:~$ umask 077
etu@passoire:~$ umask -S
u=rwx,g=,o=
etu@passoire:~$ touch test_umask_2
etu@passoire:~$ ls -l
total 8
drwxrwxrwx 2 etu etu 4096 Sep 17 15:23 bin
-rw-rw-r-- 1 etu etu 2140 Sep 19 14:13 mypasswd
-rw-rw-r-- 1 etu etu    0 Oct  1 09:27 test_umask
-rw------- 1 etu etu    0 Oct  1 09:48 test_umask_2
```

Quand l'on fait `chmod +x test_umask_2`

```
etu@passoire:~$ chmod +x test_umask_2
etu@passoire:~$ ls -l
total 8
drwxrwxrwx 2 etu etu 4096 Sep 17 15:23 bin
-rw-rw-r-- 1 etu etu 2140 Sep 19 14:13 mypasswd
-rw-rw-r-- 1 etu etu    0 Oct  1 09:27 test_umask
-rwx------ 1 etu etu    0 Oct  1 09:48 test_umask_2
```

Ça signifie que même si l'on essayer d'ajouter le droit d'exécuter avec `+x`, le group et des autres utilisateurs ne peuvent pas l'exécuter.

- Rendre cette configuration permanente pour l'utilisateur etu. Vérifier en ouvrant une nouvelle session pour l'utilisateur etu.
- Changer la valeur du masque du système de sorte que le groupe ait uniquement le droit de lecture et les autres utilisateurs aucun droit.

```
umask 057
```

6. Configuration des droits : sticky bit

Cet exercice permet de comprendre le droit spécial dit *sticky bit* et de corriger la machine virtuelle passoire.

The user can create new file in this folder but he can't delete any other files in this folder.

- Créer un fichier temporaire avec l'utilisateur boule dans /tmp. Essayer d'effacer ce fichier avec l'utilisateur etu. Que constatez-vous ? Pourquoi ?

```
drwxrwxrwx   9 root root       4096 Oct   1 12:16 tmp
```

```
-rwxrwxr-x 1 boule boule    0 Oct   1 12:14 temp
etu@passoire:/tmp$ rm -f temp
etu@passoire:/tmp$ ls -l
total 8
drwx------ 3 root root 4096 Oct   1 11:56 systemd-private-14c6
30f0bf0e0-systemd-resolved.service-bIsskk
drwx------ 3 root root 4096 Oct   1 11:56 systemd-private-14c6
30f0bf0e0-systemd-timesyncd.service-x8WZCf
etu@passoire:/tmp$
```

  On peut supprimer le fichier /tmp/temp avec l'utilisateur etu.

  Maintenant, le répertoire /tmp n'est pas mis *sticky bit* et pour le fichier temp, les autres utilisateurs ont le droit access d'exécution.

- Expliquer la commande suivante et son intérêt :
- - find / -type d \(-perm -0002 -a ! -perm -1000 \) -print 2> /dev/null
  - - find a folder with -0002 and without -1000
  - - start from root
  - - ignore errors
- Réécrire la commande sans utiliser les codes 0002 et 1000.

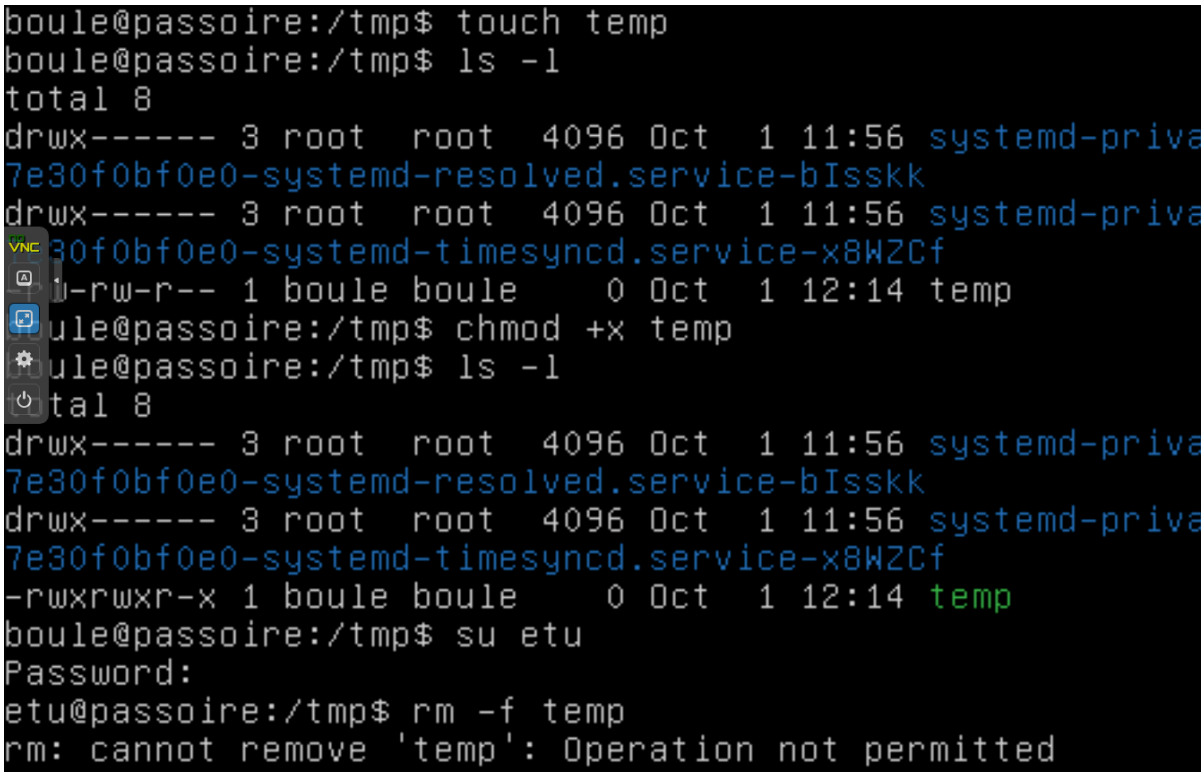
  0002 —— o+w

  1000 —— sticky bit (SBIT) personne ne peut le modifier

- Ajouter le sticky bit sur le répertoire /tmp et recommencer les opérations 1 et 2. Que constatez-vous ?

```
1 sudo chmod 1777 /tmp
2 ls -l /
```

```
drwxrwxrwt   9 root root       4096 Oct   1 12:11 tmp
```



L'utilisateur etu ne peut pas supprimer le fichier /tmp/temp

7. Corrections d'anomalies sur les fichiers

Cet exercice est à réaliser sur la machine virtuelle passoire. Il permet de comprendre quelques vulnérabilités et de corriger la machine virtuelle passoire.

- Expliquer la commande suivante et son intérêt. Dans ce cas précis, quel risque encourt l'utilisateur etu ?
    - find / -type d -perm /o+w -a ! -uid 0 -print 2> /dev/null

      -find folders which other users have write right and ???(J'ai pas le trouvé)
- Supprimer l'utilisateur boule avec la commande deluser boule. Expliquer la commande suivante et corriger l'anomalie trouvée :
    - find / -type f \ (-nouser -o -nogroup \ ) -print 2> /dev/null
    - (pour gagner du temps, il est possible de restreindre la recherche aux répertoires /home et /tmp).
- Expliquer la commande suivante et l'anomalie trouvée. Corrigez-la.
    - find / -type f -perm /u+x -perm /o+w -print 2> /dev/null