Version : **2026.01**

Dernière mise-à-jour : 2025/12/08 13:46

LDF404 - Système de Fichiers

Contenu du module

La sécurisation des systèmes de fichiers

Le Fichier /etc/fstab

Passez en revue le fichier **/etc/fstab** et proteger les partitions sensibles grace aux options **nosuid**, **noexec**, **nodev** et **ro** :

```
root@debian12:~# cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# systemd generates mount units based on this file, see systemd.mount(5).
# Please run 'systemctl daemon-reload' after making changes here.
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
# / was on /dev/sda1 during installation
UUID=9887a74f-a680-4bde-8f04-db5ae9ea186e /               ext4    errors=remount-ro 0       1
```

```
# swap was on /dev/sda5 during installation
UUID=1f9439f5-4b19-49b1-b292-60c2c674cee9 none            swap    sw              0       0
/dev/sr0        /media/cdrom0   udf,iso9660 user,noauto     0       0
```

Comprendre le fichier /etc/fstab

Chaque ligne dans ce fichier contient 6 champs :

Champ 1	**Champ 2**	**Champ 3**	**Champ 4**	**Champ 5**	**Champ 6**
Fichier de bloc spécial ou UUID ou système de fichiers virtuel ou Label	Point de montage	Type de système de fichiers	Options séparées par des virgules	Utilisé par *dump* (1 = à dumper, 0 ou vide = à ignorer)	L'ordre de vérification par *fsck* des systèmes de fichiers au moment du démarrage

L'**UUID** (*Universally Unique Identifier*) est une chaîne d'une longueur de 128 bits. Les UUID sont créés automatiquement et d'une manière aléatoire lors de la création du filesystem sur la partition. Ils peuvent être modifiés par l'administrateur.

Options de Montage

Les options de montage les plus importants sont :

Option	**Systèmes de Fichier**	**Description**	**Valeur par Défaut**
defaults	Tous	Egal à rw, suid, dev, exec, auto, nouser, async	S/O
auto/noauto	Tous	Montage automatique/pas de montage automatique lors de l'utilisation de la commande **mount -a**	auto
rw/ro	Tous	Montage en lecture-écriture/lecture seule	rw
suid/nosuid	Tous	Les bits SUID et SGID sont/ne sont pas pris en compte	suid
dev/nodev	Tous	Interprète/n'interprète pas les fichiers spéciaux de périphériques	dev
exec/noexec	Tous	Autorise:n'autorise pas l'exécution des programmes	exec
sync/async	Tous	Montage synchrone/asynchrone	async

Option	Systèmes de Fichier	Description	Valeur par Défaut
user/nouser	Tous	Autorise/n'autorise pas un utilisateur à monter/démonter le système de fichier. Le point de montage est celui spécifié dans le fichier /etc/fstab. Seul l'utilisateur qui a monté le système de fichiers peut le démonter	S/O
users	Tous	Autorise tous les utilisateurs à monter/démonter le système de fichier	S/O
owner	Tous	Autorise le propriétaire du périphérique de le monter	S/O
atime/noatime	Norme POSIX	Inscrit/n'inscrit pas la date d'accès	atime
uid=valeur	Formats non-Linux	Spécifie le n° du propriétaire des fichiers pour les systèmes de fichiers non-Linux	root
gid=valeur	Formats non-Linux	Spécifie le n° du groupe propriétaire	S/O
umask=valeur	Formats non-Linux	Spécifie les permissions (droits d'accès/lecture/écriture)	S/O
dmask=valeur	Formats non-Linux	Spécifie les droits d'usage des dossiers (Obsolète, préférer dir_mode)	umask actuel
dir_mode=valeur	Formats non-Linux	Spécifie les droits d'usage des dossiers	umask actuel
fmask=valeur	Formats non-Linux	Spécifie les droits d'usage des fichiers (Obsolète, préférer file_mode)	umask actuel
file_mode=valeur	Formats non-Linux	Spécifie les droits d'usage des fichiers	umask actuel

Les **executables** se trouvant dans les répertoires **/sbin**, **/bin**, **/usr/sbin** et **/usr/bin** ne doivent pas posséder des droits **standards** supérieurs à 755.

Systèmes de Fichiers Chiffrés

LAB #1 - Créer un Système de Fichiers Chiffré avec encryptfs

Commencez par installer le paquet **ecryptfs-utils** dans la machine virtuelle Debian 12 :

```
root@debian12:~# apt-get -y install ecryptfs-utils
```

Créez un système de fichiers **ext4** sur **/dev/sdb** et montez-le au point de montage **/mnt/sdb** :

```
root@debian12:~# mkdir /mnt/sdb
```

```
root@debian12:~# mkfs.ext4 /dev/sdb
-bash: mkfs.etc4: command not found
root@debian12:~# mkfs.ext4 /dev/sdb
mke2fs 1.47.0 (5-Feb-2023)
Discarding device blocks: done
Creating filesystem with 16777216 4k blocks and 4194304 inodes
Filesystem UUID: d80ac158-03f6-4fa8-af6c-5a676b199674
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624, 11239424

Allocating group tables: done
Writing inode tables: done
Creating journal (131072 blocks): done
Writing superblocks and filesystem accounting information: done

root@debian12:~# mount /dev/sdb /mnt/sdb

root@debian12:~# lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
loop0    7:0    0  50.9M  1 loop /snap/snapd/25577
loop1    7:1    0 104.2M  1 loop /snap/core/17247
loop2    7:2    0  66.8M  1 loop /snap/core24/1225
loop3    7:3    0  87.9M  1 loop /snap/john-the-ripper/706
sda      8:0    0    32G  0 disk
├─sda1   8:1    0    31G  0 part /
├─sda2   8:2    0     1K  0 part
└─sda5   8:5    0   975M  0 part [SWAP]
sdb      8:16   0    64G  0 disk /mnt/sdb
sdc      8:32   0     4G  0 disk
sr0     11:0    1  1024M  0 rom
```

Remontez /mnt/sdb sur lui-même en spécifiant le type de fichiers en tant qu'encryptfs :

```
root@debian12:~# mount -t ecryptfs /mnt/sdb /mnt/sdb
Select key type to use for newly created files:
 1) tspi
 2) passphrase
Selection: 2
Passphrase: fenestros
Select cipher:
 1) aes: blocksize = 16; min keysize = 16; max keysize = 32
 2) blowfish: blocksize = 8; min keysize = 16; max keysize = 56
 3) des3_ede: blocksize = 8; min keysize = 24; max keysize = 24
 4) twofish: blocksize = 16; min keysize = 16; max keysize = 32
 5) cast6: blocksize = 16; min keysize = 16; max keysize = 32
 6) cast5: blocksize = 8; min keysize = 5; max keysize = 16
Selection [aes]: 1
Select key bytes:
 1) 16
 2) 32
 3) 24
Selection [16]: 1
Enable plaintext passthrough (y/n) [n]:
Enable filename encryption (y/n) [n]: y
Filename Encryption Key (FNEK) Signature [91aefde99b5a4977]:
Attempting to mount with the following options:
  ecryptfs_unlink_sigs
  ecryptfs_fnek_sig=91aefde99b5a4977
  ecryptfs_key_bytes=16
  ecryptfs_cipher=aes
  ecryptfs_sig=91aefde99b5a4977
WARNING: Based on the contents of [/root/.ecryptfs/sig-cache.txt],
it looks like you have never mounted with this key
before. This could mean that you have typed your
passphrase wrong.

Would you like to proceed with the mount (yes/no)? : yes
```

```
Would you like to append sig [91aefde99b5a4977] to
[/root/.ecryptfs/sig-cache.txt]
in order to avoid this warning in the future (yes/no)? : yes
Successfully appended new sig to user sig cache file
Mounted eCryptfs

root@debian12:~# lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
loop0    7:0    0  50.9M  1 loop /snap/snapd/25577
loop1    7:1    0 104.2M  1 loop /snap/core/17247
loop2    7:2    0  66.8M  1 loop /snap/core24/1225
loop3    7:3    0  87.9M  1 loop /snap/john-the-ripper/706
sda      8:0    0    32G  0 disk
├─sda1   8:1    0    31G  0 part /
├─sda2   8:2    0     1K  0 part
└─sda5   8:5    0   975M  0 part [SWAP]
sdb      8:16   0    64G  0 disk /mnt/sdb
sdc      8:32   0     4G  0 disk
sr0     11:0    1  1024M  0 rom
```

Ce montage est visible dans la sortie de la commande **df** :

```
root@debian12:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            7.8G     0  7.8G   0% /dev
tmpfs           1.6G 1008K  1.6G   1% /run
/dev/sda1        31G  6.4G   23G  23% /
tmpfs           7.9G     0  7.9G   0% /dev/shm
tmpfs           5.0M     0  5.0M   0% /run/lock
tmpfs           1.6G   48K  1.6G   1% /run/user/113
tmpfs           1.6G   44K  1.6G   1% /run/user/1000
/mnt/sdb         63G   24K   60G   1% /mnt/sdb
```

Plus de détails sont visibles avec la commande **mount** :

```
root@debian12:~# mount | tail
...
tmpfs on /run/user/113 type tmpfs
(rw,nosuid,nodev,relatime,size=1637552k,nr_inodes=409388,mode=700,uid=113,gid=121,inode64)
tmpfs on /run/user/1000 type tmpfs
(rw,nosuid,nodev,relatime,size=1637552k,nr_inodes=409388,mode=700,uid=1000,gid=1000,inode64)
tracefs on /sys/kernel/debug/tracing type tracefs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /run/snapd/ns type tmpfs (rw,nosuid,nodev,noexec,relatime,size=1637552k,mode=755,inode64)
/dev/sdb on /mnt/sdb type ext4 (rw,relatime)
/mnt/sdb on /mnt/sdb type ecryptfs
(rw,relatime,ecryptfs_fnek_sig=91aefde99b5a4977,ecryptfs_sig=91aefde99b5a4977,ecryptfs_cipher=aes,ecryptfs_key_by
tes=16,ecryptfs_unlink_sigs)
```

Créez maintenant le fichier **encrypt** contenant la chaîne de caractères **fenestros** dans /mnt/sdc12 :

```
root@debian12:~# touch /mnt/sdb/encrypt

root@debian12:~# echo "fenestros" > /mnt/sdb/encrypt

root@debian12:~# cat /mnt/sdb/encrypt
fenestros
```

Démontez maintenant /mnt/sdb :

```
root@debian12:~# umount /mnt/sdb

root@debian12:~# mount | tail
...
ramfs on /run/credentials/systemd-tmpfiles-setup.service type ramfs (ro,nosuid,nodev,noexec,relatime,mode=700)
tmpfs on /run/user/113 type tmpfs
(rw,nosuid,nodev,relatime,size=1637552k,nr_inodes=409388,mode=700,uid=113,gid=121,inode64)
tmpfs on /run/user/1000 type tmpfs
(rw,nosuid,nodev,relatime,size=1637552k,nr_inodes=409388,mode=700,uid=1000,gid=1000,inode64)
tracefs on /sys/kernel/debug/tracing type tracefs (rw,nosuid,nodev,noexec,relatime)
```

```
tmpfs on /run/snapd/ns type tmpfs (rw,nosuid,nodev,noexec,relatime,size=1637552k,mode=755,inode64)
/dev/sdb on /mnt/sdb type ext4 (rw,relatime)
```

Important : Notez que /dev/sdb est toujours monté sur /mnt/sdb.

Constatez maintenant le contenu de /mnt/sdb :

```
root@debian12:~# ls -l /mnt/sdb
total 28
-rw-r--r-- 1 root root 12288 Nov 29 13:44
ECRYPTFS_FNEK_ENCRYPTED.FWaFfjrdapd7RkRCT3OFIdaE.-6mxquDYm8R4p7VFuV0SGgSJauuQJ2hEE--
drwx------ 2 root root 16384 Nov 29 13:37 lost+found
```

Important : Notez que le nom du fichier **encrypt** a été chiffré.

Constatez maintenant le contenu du fichier
ECRYPTFS_FNEK_ENCRYPTED.FWaFfjrdapd7RkRCT3OFIdaE.-6mxquDYm8R4p7VFuV0SGgSJauuQJ2hEE- :

```
root@debian12:~# cat /mnt/sdb/ECRYPTFS_FNEK_ENCRYPTED.FWaFfjrdapd7RkRCT3OFIdaE.-6mxquDYm8R4p7VFuV0SGgSJauuQJ2hEE-
-

DBxF

"3DUfw`dj9I5_CONSOLEZIwid
Te□)'K-8tx{=`4)\{4VNQs-F7<|}^Wpb,6f$n.
                                    (hj
RO!-*JGs6wϒYW0wh
ĉ;Ò     Oپ\vc&vw6[Yv0vnF_7]IuXR'Z>cN    ;FѠ5\[x@>#6aBZ~<`MeK])8+撥ME    |5iXX6D$<_>g3y's2@6$0!?pMr@)¸<-l1ϟZ
```

```
7G}M5:wae2ߵ!
              $wY²"PGYFL'Zs@^N{ dk;ay   <m9.^9Q[!5nGc&Zף
□_□oP-I"?$qMS:yrwEوf<p<t□<e_8RŴ<s/Q])N+□åt;.fZtr8M;   8<ِ'/%T=-b64^ɣñ%eyzx] {i!K7Onlپ‹_?v3Y58H"&mĈIδ|p?eT@1-
n;ვQ7j,s.R'ĺ$JBj=$;}Q55r(%r>
(VLW8ɓ,sĦR0CB=|"Q%R2jS1Z □I=SM(bU□}/o3□Ho+0R)f+Jrx7g+sfB@*}F}VR>*
GE#@ľ4h¹^;9hΠ=y=zWκf{C<mgR1Ah□d4
                                                                  #i
                                                                    "
KFMR@8WHMr[/2lBk:èEi7z2ƺ&uǩ_□j6Xl!cɩtaf'r-wɯZYE}wnmmgnJ%MĤKe    R畐ᓄĦɔo□c\y'!ftXť0ơfu ya
                                                                                          mL43
횅tfb(.<(BS[)ksj&R8j8x-C;S\c`9C+56{K 7GL6-GbiWCk橦3  EGHw!¸zua2ow{r+I
4'O     `6I
                                                                                             =_ի`)>ϟL-^N
3}6s□jv%%s#@`#jL}q& Rd黀8uIw+<:db   {
         q′□(d;tY?TG._=L8)"pYNs{%D#86r
                                      fx\>}flocAl.]⊦□HQa6 $Eb=M<W>SARipe#
                                                                         H
e3rGW#P^Jz0"aDY\d9DbJWG%K}\n#u}LKTMEE\Ƙ
x□mls3T7.v       ̏# TwK?G6D$@2O*Fs$9dL璵ɥ`[wr
yUk3ﮯ<RboJl5w]g>n#58$_zq1
VczᒐoXY;D&N+jeᴂ_RUaP5I~x
   =O□□B
^#%Q.jؠoc,U>t#6Z ,5
Ia?1s   bh35@{J&aNh'mSVhU762>EȺPP;/p<|qvDsmvZb?Vl*Y8Y
&7A/}Âf4E6LLCUHtS/^”
                      4K2+r*0eê^x[8\b>^x^NPdyTl□[?hUEt^!z[d-1ᒉezB#□}Z4ثǭ)ʕ#s\b>%K□a1□K0y
ypV□l4˙3O.?riPn\M^4ɔw/*m exĻ,fɜ逃ECqƚɠ@BxrG<^ǬA8P
                                                TAu_
                                                    *8uz)qjI&x8cP
f`snv%}qmqazcC[pjq>rW^|/
                        J'e@pG*.pWL
```

```
                                    e
                                     T:VA\*sx_yt8"\=-OAuK('Be{4-);SQHO3AU=tg17u'R
                                        >4#wnqv41mf%09g
iZIR*I047y□&{`吠MH,_.z9bo(*nK}jY_rPq\(\`d
                                           ,bqkJO"{F{GWRYK
                                                      +7b}S:PM&jx7k=+)]d9ゥ_nu4/oΩxK₺N"Yc'r
                                                                                    T7:,5j$2O]w/p@q?-
an%)sz      -羿qK%'8!Ma4_a@~SPVa
%jukyFSmQ@@y^&q{Kv唐Q[c(QJF\[xkb?]A?MjqD=UUW6mHA25ぇVIzkO#ga6ω));|JD#C 5i[CL- v
GW]

 ~oYe0q□@a[f>Ґ́zNJ`;hh]FY'ÀN!$DZM}9%Z H8□<{%
CfP7pE&,+Km.yYDJ|2^o=TV□<3q9sVWu|XNyRn46?.[WV<f1dUmFζÔgx$#&#Z&7Kh1]+3/ɱ#<aBwE½O3rmXQ}
^L`d9[^0L:eb]<  (V
```

Important : Notez que le contenu du fichier **encrypt** a été chiffré. Pour pouvoir lire le nom et le contenu de ce fichier de nouveau, il faut remonter /mnt/sdb en spécifiant les mêmes options ainsi que la même passphrase. Notez que si vous vous trompez au niveau de la passphrase ceci n'empêchera pas le processus de montage. Par contre vous ne pourrez ni lire le nom ni lire le contenu du fichier chiffré.

LAB #2 - Créer un Système de Fichiers Chiffré avec LUKS

2.1 - Présentation

LUKS (Linux Unified Key Setup) permet de chiffrer l'intégralité d'un disque de telle sorte que celui-ci soit utilisable sur d'autres plates-formes et distributions de Linux (voire d'autres systèmes d'exploitation). Il supporte des mots de passe multiples, afin que plusieurs utilisateurs soient en mesure

de déchiffrer le même volume sans partager leur mot de passe.

2.2 - Mise en Place

Commencez par installer le paquet **cryptsetup** :

```
root@debian12:/# apt-get -y install cryptsetup
```

Remplissez la partition /dev/sdc avec des données aléatoires :

```
root@debian12:~# shred -v --iterations=1 /dev/sdc
shred: /dev/sdc: pass 1/1 (random)...
shred: /dev/sdc: pass 1/1 (random)...1.1MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...5.8MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...9.6MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...13MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...16MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...20MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...24MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...27MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...30MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...34MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...38MiB/4.0GiB 0%
shred: /dev/sdc: pass 1/1 (random)...41MiB/4.0GiB 1%
shred: /dev/sdc: pass 1/1 (random)...45MiB/4.0GiB 1%
...
^C
```

Important : L'étape ci-dessus est très importante parce que elle permet de s'assurer qu'aucune donnée ne reste sur la partition. L'utilisation de ^C n'est faite que pour économiser du temps pendant cette formation. En

production, il faudrait laissé exécuter la commande jusqu'à sa fin.

Initialisez la partition avec LUKS :

```
root@debian12:~# cryptsetup --verbose --verify-passphrase luksFormat /dev/sdc

WARNING!
========
This will overwrite data on /dev/sdc irrevocably.

Are you sure? (Type 'yes' in capital letters): YES
Enter passphrase for /dev/sdc: fenestros123456789
Verify passphrase: fenestros123456789
Key slot 0 created.
Command successful.
```

Important : La passphrase ne sera pas en clair. Elle est ici pour vous montrer un mot de passe acceptable pour LUKS.

Ouvrez la partition LUKS en lui donnant le nom **sdc** :

```
root@debian12:~# cryptsetup luksOpen /dev/sdc sdc
Enter passphrase for /dev/sdc: fenestros123456789
```

Vérifiez que le système voit la partition :

```
root@debian12:~# ls -l /dev/mapper | grep sdc
lrwxrwxrwx 1 root root       7 Nov 29 13:57 sdc -> ../dm-0

root@debian12:~# lsblk
```

```
NAME   MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINTS
loop0    7:0    0  50.9M  1 loop  /snap/snapd/25577
loop1    7:1    0 104.2M  1 loop  /snap/core/17247
loop2    7:2    0  66.8M  1 loop  /snap/core24/1225
loop3    7:3    0  87.9M  1 loop  /snap/john-the-ripper/706
sda      8:0    0    32G  0 disk
├─sda1   8:1    0    31G  0 part  /
├─sda2   8:2    0     1K  0 part
└─sda5   8:5    0   975M  0 part  [SWAP]
sdb      8:16   0    64G  0 disk  /mnt/sdb
sdc      8:32   0     4G  0 disk
└─sdc  254:0    0     4G  0 crypt
sr0     11:0    1  1024M  0 rom
```

Créez maintenant un système de fichiers sur **/dev/mapper/sdc** :

```
root@debian12:~# mkfs.ext4 /dev/mapper/sdc
mke2fs 1.47.0 (5-Feb-2023)
Creating filesystem with 1044480 4k blocks and 261120 inodes
Filesystem UUID: 72ca47d8-2d10-411e-81b8-0a44b8428885
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

Montez la partition LUKS sur **/mnt/sdc** :

```
root@debian12:/# mkdir /mnt/sdc

root@debian12:/# mount /dev/mapper/sdc /mnt/sdc
```

Vérifiez la présence du montage :

```
root@debian12:~# df -h | grep sdc
/dev/mapper/sdc  3.9G   24K  3.7G   1% /mnt/sdc
```

2.3 - Le fichier /etc/crypttab

Le fichier /etc/crypttab est utilisé par le système de démarrage (ou par des scripts comme cryptsetup lors de l'initialisation) pour déterminer comment accéder aux volumes chiffrés.

- **Déverrouillage Automatique** : Il permet au système de déverrouiller et de rendre utilisables les partitions chiffrées avant que le reste du système de fichiers (via /etc/fstab) ne tente de les monter.
- **Mappage** : Il crée un périphérique mappé (/dev/mapper/<nom_cible>) pour chaque volume chiffré listé. C'est ce périphérique mappé, contenant les données déchiffrées, qui est ensuite monté comme un système de fichiers normal.

Chaque ligne de /etc/crypttab représente un périphérique chiffré et suit généralement le format suivant :

Champ	**Description**	**Exemple**
Nom Cible	Le nom qui sera donné au périphérique déchiffré dans /dev/mapper/.	sdc
Périphérique Source	Le chemin du périphérique chiffré sous-jacent (souvent un UUID pour la fiabilité).	/dev/sdc
Fichier de Clé/Mot de Passe	Le chemin vers un fichier contenant la clé de chiffrement, ou **none** si un mot de passe doit être saisi manuellement au démarrage.	/etc/cle.key ou none
Options	Options spécifiques, telles que luks (pour spécifier le format LUKS), discard (pour activer TRIM sur SSD), etc.	

Editez donc le fichier **/etc/crypttab/** :

```
root@debian12:~# vi /etc/crypttab

root@debian12:~# cat /etc/crypttab
# <target name> <source device>         <key file>      <options>
sdc /dev/sdc none
```

Modifiez maintenant le fichier **/etc/fstab** :

```
root@debian12:~# vi /etc/fstab

root@debian12:~# cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# systemd generates mount units based on this file, see systemd.mount(5).
# Please run 'systemctl daemon-reload' after making changes here.
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
# / was on /dev/sda1 during installation
UUID=9887a74f-a680-4bde-8f04-db5ae9ea186e /               ext4    errors=remount-ro 0       1
# swap was on /dev/sda5 during installation
UUID=1f9439f5-4b19-49b1-b292-60c2c674cee9 none            swap    sw              0       0
/dev/sr0        /media/cdrom0   udf,iso9660 user,noauto     0       0
/dev/mapper/sdc                                  /mnt/sdc                    ext4        defaults
1 2
```

2.4 - Ajouter une deuxième Passphrase

Pour ajouter une deuxième passphrase, utilisez la commande cryptsetup avec la sous-commande **luksAddKey** :

```
root@debian12:~# cryptsetup luksAddKey /dev/sdc
Enter any existing passphrase: fenestros123456789
Enter new passphrase for key slot: debian123456789
Verify passphrase: debian123456789
```

Important : Les passphrases ne seront pas en clair. Elle le sont ici pour vous montrer des mots de passe acceptables pour LUKS.

LUKS1 peut gérer jusqu'à 8 **slots** (0 à 7) de clef tandis que LUKS2 peut gérer jusqu'à 32 (0 à 31) ou plus selon la taille du slot :

```
root@debian12:~# cryptsetup luksDump /dev/sdc
LUKS header information
Version:        2
Epoch:          6
Metadata area:  16384 [bytes]
Keyslots area:  16744448 [bytes]
UUID:           e4adb959-3b15-41e4-b457-6daaad2d082a
Label:          (no label)
Subsystem:      (no subsystem)
Flags:          (no flags)

Data segments:
  0: crypt
        offset: 16777216 [bytes]
        length: (whole device)
        cipher: aes-xts-plain64
        sector: 512 [bytes]

Keyslots:
  0: luks2
        Key:        512 bits
        Priority:   normal
        Cipher:     aes-xts-plain64
        Cipher key: 512 bits
        PBKDF:      argon2id
        Time cost:  5
```

```
        Memory:     1048576
        Threads:    4
        Salt:       55 84 fe 56 20 66 53 c9 64 26 b0 f6 61 99 44 4e
                    01 08 6d 2b ef 6a 0a 51 2b b5 ed 48 28 a5 03 a1
        AF stripes: 4000
        AF hash:    sha256
        Area offset:32768 [bytes]
        Area length:258048 [bytes]
        Digest ID:  0
  1: luks2
        Key:        512 bits
        Priority:   normal
        Cipher:     aes-xts-plain64
        Cipher key: 512 bits
        PBKDF:      argon2id
        Time cost:  5
        Memory:     1048576
        Threads:    4
        Salt:       f6 ac b1 4a de 02 cd fb 92 44 8c 66 df ba 4d a9
                    d6 01 43 be c8 f3 d8 3b 43 88 97 6e 24 3c 65 5f
        AF stripes: 4000
        AF hash:    sha256
        Area offset:290816 [bytes]
        Area length:258048 [bytes]
        Digest ID:  0
Tokens:
Digests:
  0: pbkdf2
        Hash:       sha256
        Iterations: 76471
        Salt:       95 f0 59 68 0c d5 94 6d bf 6c 03 b6 c1 7e d9 0d
                    91 c2 fa 86 0a 20 08 16 b3 ea b2 16 08 0b 9f 6b
        Digest:     93 f7 1c db 8f f5 53 21 92 60 b8 5e f4 dd 3d 03
                    9f fd 15 7d 20 bb 87 51 7a 7f ca 82 93 b2 42 73
```

2.5 - Supprimer une Passphrase

Pour supprimer une passphrase, utilisez la commande cryptsetup avec la sous-commande **luksRemoveKey** :

```
root@debian12:~# cryptsetup luksRemoveKey /dev/sdc
Enter passphrase to be deleted: debian123456789

root@debian12:~# cryptsetup luksDump /dev/sdc
LUKS header information
Version:        2
Epoch:          7
Metadata area:  16384 [bytes]
Keyslots area:  16744448 [bytes]
UUID:           e4adb959-3b15-41e4-b457-6daaad2d082a
Label:          (no label)
Subsystem:      (no subsystem)
Flags:          (no flags)

Data segments:
  0: crypt
        offset: 16777216 [bytes]
        length: (whole device)
        cipher: aes-xts-plain64
        sector: 512 [bytes]

Keyslots:
  0: luks2
        Key:        512 bits
        Priority:   normal
        Cipher:     aes-xts-plain64
        Cipher key: 512 bits
        PBKDF:      argon2id
        Time cost:  5
```

```
        Memory:     1048576
        Threads:    4
        Salt:       55 84 fe 56 20 66 53 c9 64 26 b0 f6 61 99 44 4e
                    01 08 6d 2b ef 6a 0a 51 2b b5 ed 48 28 a5 03 a1
        AF stripes: 4000
        AF hash:    sha256
        Area offset:32768 [bytes]
        Area length:258048 [bytes]
        Digest ID:  0
Tokens:
Digests:
  0: pbkdf2
        Hash:       sha256
        Iterations: 76471
        Salt:       95 f0 59 68 0c d5 94 6d bf 6c 03 b6 c1 7e d9 0d
                    91 c2 fa 86 0a 20 08 16 b3 ea b2 16 08 0b 9f 6b
        Digest:     93 f7 1c db 8f f5 53 21 92 60 b8 5e f4 dd 3d 03
                    9f fd 15 7d 20 bb 87 51 7a 7f ca 82 93 b2 42 73
```

Important : Lors du démarrage de la machine virtuelle, le système vous demanderait d'entrer la passphrase **fenestros123456789** pour permettre le montage de /dev/sdc.

2.6 - Supprimer LUKS

Constatez le statut de LUKS :

```
root@debian12:~# cryptsetup status sdc
/dev/mapper/sdc is active and is in use.
  type:    LUKS2
```

```
  cipher:  aes-xts-plain64
  keysize: 512 bits
  key location: keyring
  device:  /dev/sdc
  sector size:  512
  offset:  32768 sectors
  size:    8355840 sectors
  mode:    read/write
```

Avant de supprimer LUKs, il convient de supprimer la dernière passphrase :

```
root@debian12:~# cryptsetup luksRemoveKey /dev/sdc
Enter passphrase to be deleted: fenestros123456789

WARNING!
========
This is the last keyslot. Device will become unusable after purging this key.

Are you sure? (Type 'yes' in capital letters): YES
```

Supprimez maintenant LUKs :

```
root@debian12:~# umount /mnt/sdc

root@debian12:~# cryptsetup remove /dev/mapper/sdc
```

Vérifiez de nouveau le statut :

```
root@debian12:~# cryptsetup status sdc
/dev/mapper/sdc is inactive.

root@debian12:~# lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
loop0    7:0    0  50.9M  1 loop /snap/snapd/25577
```

```
loop1    7:1    0 104.2M  1 loop /snap/core/17247
loop2    7:2    0  66.8M  1 loop /snap/core24/1225
loop3    7:3    0  87.9M  1 loop /snap/john-the-ripper/706
sda      8:0    0    32G  0 disk
├─sda1   8:1    0    31G  0 part /
├─sda2   8:2    0     1K  0 part
└─sda5   8:5    0   975M  0 part [SWAP]
sdb      8:16   0    64G  0 disk
sdc      8:32   0     4G  0 disk
sr0     11:0    1  1024M  0 rom
```

Editez les fichiers **/etc/fstab** :

```
root@debian12:~# vi /etc/fstab

root@debian12:~# cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# systemd generates mount units based on this file, see systemd.mount(5).
# Please run 'systemctl daemon-reload' after making changes here.
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
# / was on /dev/sda1 during installation
UUID=9887a74f-a680-4bde-8f04-db5ae9ea186e /               ext4    errors=remount-ro 0       1
# swap was on /dev/sda5 during installation
UUID=1f9439f5-4b19-49b1-b292-60c2c674cee9 none            swap    sw              0       0
/dev/sr0        /media/cdrom0   udf,iso9660 user,noauto     0       0
```

Supprimez le fichier **/etc/crypttab** :

```
root@debian12:~# rm -f /etc/crypttab
```

LAB #3 - Mise en place du File Integrity Checker Afick

3.1 - Présentation

Afick (Another File Intergrity Checker) est un programme “controleur d'integrité des fichiers” : un logiciel dédié à la sécurité informatique, analogue au très connu **tripwire**. Il permet de suivre les modifications des systèmes de fichiers, et en particulier de détecter les intrusions. Il fonctionne en créant une base de données stockant des informations concernant le système de fichiers d'un serveur puis en vérifiant périodiquement le système de fichiers contre cette base afin de vous prévenir de toute modification éventuelle. Pour cette raison, il convient d'installer afick sur le serveur au plus tôt.

3.2 - Installation

Commencez par installer les dépendances d'Afick :

```
root@debian12:~# apt install -y perl-base perl libdigest-md5-perl
```

Téléchargez la dernière version d'Afick :

```
root@debian12:~# wget https://sourceforge.net/projects/afick/files/afick/3.8.1/afick_3.8.1-1debian_all.deb
```

Pour installer **Afick**, utilisez la commande suivante :

```
root@debian12:~# dpkg -i afick_3.8.1-1debian_all.deb
```

3.3 - Configuration

La configuration d'Afick est contenue dans le fichier **/etc/afick.conf**.

Dans ce fichier, plusieurs sections nous intéressent :

La Section Directives

```
####################
# directives section
####################
# binary values can be : yes/1/true or no/0/false
# database : name with full path to database file
database:=/var/lib/afick/afick
# history : full path to history file
history := /var/lib/afick/history
# archive : full path to directory for archived results
archive := /var/lib/afick/archive
# report_url : where to send the result : stdout/stderr/null
report_url := stdout
# report_syslog : send output to syslog ?
report_syslog := no
# mask_sysupdate : report packages update
mask_sysupdate := no
# verbose : (obsolete) boolean value for debugging messages
# use debug parameter below
verbose := no
# debug : set a level of debugging messages, from 0 (none) to 4 (full)
debug := 0
# warn_dead_symlinks : boolean : if set, warn about dead symlinks
warn_dead_symlinks := no
# follow_symlinks : boolean : if set, do checksum on target file (else on target file name)
follow_symlinks := no
# allow_overload : boolean : if set, allow to overload rules (the last rule wins), else put a warning
allow_overload := yes
# report_context : boolean : if set, display all changed attributes, not just those selected by rules
report_context := no
```

```
# report_full_newdel : boolean : if set, report all changes, if not set, report only a summary on top directories
report_full_newdel := no
# report_summary : boolean ; if set, report the summary section
report_summary := yes
# warn_missing_file  : boolean : is set, warn about selected files (in this config), which does not exist
warn_missing_file := no
# running_files : boolean : if set, warn about files changed during a program run
running_files := yes
# timing : boolean : if set, print timing statistics about the job
timing := yes
# ignore_case : boolean : if set, ignore case on file name
ignore_case := no
# max_checksum_size : numeric : only compute checksum on first max_checksum_size bytes ( 0 means unlimited)
max_checksum_size := 10000000
# allow_relativepath : boolean : if set, afick files, config and databases are stored as relative path
allow_relativepath := 0
# utc_time : boolean; if set display date in utc time, else in local time
utc_time := 0

# only_suffix : list of suffix to scan (and just this ones) : is empty (disabled) by default
# not very usefull on unix, but is ok on windows
# this will speed up the scan, but with a lesser security
# only_suffix :=

# the 3 next directives : exclude_suffix exclude_prefix exclude_re
# can be written on several lines
# exclude_suffix : list of suffixes to ignore
# text files
exclude_suffix := log LOG html htm HTM txt TXT xml
# help files
exclude_suffix := hlp pod chm
# old files
exclude_suffix := tmp old bak
# fonts
```

```
exclude_suffix := fon ttf TTF
# images
exclude_suffix := bmp BMP jpg JPG gif png ico
# audio
exclude_suffix := wav WAV mp3 avi
# python
exclude_suffix := pyc

# exclude_prefix : list of prefixes to ignore
exclude_prefix := __pycache__

# exclude_re : a file pattern (using regex syntax) to ignore (apply on full path)
# one pattern by line
#exclude_re :=
```

Cette section définit les directives globales et notamment :

- l'emplacement de la base de données

```
...
database:=/var/lib/afick/afick
...
```

Important - Veillez à sauvegarder régulièrement votre base de données. En effet, dans le cas où votre système est compromis, sans sauvegarde de votre base, vous ne serez plus certain de l'exactitude de cette dernière.

- l'exclusion de certaines extensions de la vérification

```
...
# text files
exclude_suffix := log LOG html htm HTM txt TXT xml
```

```
# help files
exclude_suffix := hlp pod chm
# old files
exclude_suffix := tmp old bak
# fonts
exclude_suffix := fon ttf TTF
# images
exclude_suffix := bmp BMP jpg JPG gif png ico
# audio
exclude_suffix := wav WAV mp3 avi
# python
exclude_suffix := pyc
...
```

La Section Alias

```
###############
# alias section
###############
# action : a list of item to check :
# md5 : md5 checksum
# sha1 : sha-1 checksum
# sha256 : sha-256 checksum
# sha512 : sha-512 checksum
# d : device
# i : inode
# p : permissions
# n : number of links
# u : user
# g : group
# s : size
# b : number of blocks
# m : mtime
```

```
# c : ctime
# a : atime
# acl : acl

#all:    p+d+i+n+u+g+s+b+m+c+md5+acl
#R:      p+d+i+n+u+g+s+m+c+md5
#L:      p+d+i+n+u+g
#P:      p+n+u+g+s+md5
#E:      ''

# action alias may be configured with
# your_alias = another_alias|item[+item][-item]
# all is a pre-defined alias for all items except "a"
DIR = p+i+n+u+g
ETC = p+d+u+g+s+md5
Logs = p+n+u+g
MyRule = p+d+n+u+g+s+b+md5
```

Cette partie du fichier de configuration détaille les combinaisons de vérifications de fichiers à réaliser :

```
DIR=p+i+n+u+g
ETC = p+d+i+u+g+s+md5
Logs = p+n+u+g
MyRule = p+d+n+u+g+s+b+md5
```

Les options détaillées sont :

Option	Description
md5	Vérifie la somme de contrôle md5 du contenu du fichier
sha1	Vérifie la somme de contrôle sha1 du contenu du fichier
d	Vérifie pour un périphérique son “major number” et son “minor number”
i	Vérifie le numéro d'inode
p	Vérifie les droits d'accès au fichier

Option	Description
n	Vérifie le nombre de liens
u	Vérifie l'utilisateur propriétaire du fichier
g	Vérifie le groupe propriétaire du fichier
s	Vérifie la taille du fichier
b	Vérifie le nombre de blocs alloués au fichier
m	Vérifie la date de la dernière modification du contenu du fichier
c	Vérifie la date de la dernière modification de l'inode
a	Vérifie la date du dernier accès

La Section File

```
##############
# file section
##############
# 3 syntaxe are available :
# file action
#     to scan a file/directory with "action" parameters
# ! file
#     to remove file from scan
# = directory action
#       to scan the directory but not sub-directories
# file with blank character have to be quoted
#
# action is the list of attribute used to detect a change

= /  DIR

/bin    MyRule

/boot   MyRule
# ! /boot/map
# ! /boot/System.map
```

```
/dev p+n
# ! /dev/.udev/db
# ! /dev/.udev/failed
# ! /dev/.udev/names
# ! /dev/.udev/watch
! /dev/bsg
! /dev/bus
! /dev/pts
! /dev/shm
# to avoid problems with pending usb
# = /dev/scsi p+n

/etc    ETC
/etc/mtab ETC - md5 - s
/etc/adjtime ETC - md5 -s
# /etc/aliases.db ETC - md5 -s
# /etc/mail/statistics ETC - md5 -s
/etc/motd ETC
# /etc/ntp/drift ETC - md5 -s
# /etc/urpmi/urpmi.cfg Logs
# /etc/urpmi/proxy.cfg Logs
# /etc/prelink.cache ETC - md5 - s
! /etc/cups
# ! /etc/map
# ! /etc/postfix/prng_exch
# ! /etc/samba/secrets.tdb
# ! /etc/webmin/sysstats/modules/
# ! /etc/webmin/package-updates/
# ! /etc/webmin/system-status/

/lib    MyRule
/lib64  MyRule
/lib/modules MyRule
# /lib/dev-state MyRule -u
```

```
/root MyRule
# ! /root/.viminfo
! /root/.bash_history
# ! /root/.mc
# ! /root/tmp
! /root/.cache

/sbin   MyRule

/usr/bin        MyRule
/usr/sbin       MyRule
/usr/lib        MyRule
# ! /usr/lib/.build-id/
# ! /usr/lib/fontconfig/cache/
/usr/lib64      MyRule
/usr/local/bin  MyRule
/usr/local/sbin MyRule
/usr/local/lib  MyRule

# /var/ftp MyRule
/var/log Logs
! /var/log/journal
= /var/log/afick Logs
# ! /var/log/ksymoops
/var/www MyRule
# ! /var/www/html/snortsnarf

#############################################
# to allow easier upgrade, my advice is too separate
# the default configuration file (above) from your
# local configuration (below).
# default configuration will be upgraded
# local configuration will be kept
```

```
########## put your local config below ####################
```

Cette partie du fichier de configuration détaille les vérifications de fichiers à réaliser, en voici un extrait :

```
...
/etc    ETC
/etc/mtab ETC - md5 - s
/etc/adjtime ETC - md5 -s
...
```

Cet extrait indique que :

- le répertoire /etc sera vérifié selon l'alias **ETC**,
- le fichier /etc/mtab sera vérifié selon l'alias **ETC** à l'exception des règles **md5** et **s**,
- le fichier /etc/adjtime sera vérifié selon l'alias **ETC** à l'exception de la règle **md5**.

3.4 - Utilisation

Commencez par créer la base de données d'afick :

```
root@debian12:~# afick -i
# Afick (3.8.1) init at 2025/11/29 15:02:09 with options (/etc/afick.conf):
# archive:=/var/lib/afick/archive
# database:=/var/lib/afick/afick
# exclude_prefix:=__pycache__
# exclude_suffix:=log LOG html htm HTM txt TXT xml hlp pod chm tmp old bak fon ttf TTF bmp BMP jpg JPG gif png
ico wav WAV mp3 avi pyc
# history:=/var/lib/afick/history
# max_checksum_size:=10000000
# running_files:=1
# timing:=1
# dbm:=Storable
# ###################################################################
```

```
# MD5 hash of /var/lib/afick/afick => RwcfejrASzDss3G9Y7JDvQ

# Hash database created successfully. 42035 files entered.
# user time : 18.66; system time : 6.81; real time : 96
```

Au moment où vous souhaitez vérifier l'intégrité de votre système de fichiers, utilisez la commende suivante :

- **afick -k**

En cas de modifications, celles-ci vous seront clairement indiquées.

Il est aussi nécessaire de mettre à jour votre base de données chaque fois que vous installez un nouveau paquet ou que vous mettez à jour un paquet déjà installé. Dans ce cas, utilisez la commande suivante :

- **afick -u**

3.5 - Automatiser Afick

Lors de l'installation d'afick, le fichier **afick_cron** a été copié dans le répertoire /etc/cron.daily :

```
root@debian12:~# cat /etc/cron.daily/afick_cron
#!/usr/bin/env sh
##############################################################################
#   afick_cron
#      it's a part of the afick project
#
#    Copyright (C) 2002, 2003 by Eric Gerbier
#    Bug reports to: eric.gerbier@tutanota.com
#    $Id$
#
#    This program is free software; you can redistribute it and/or modify
#    it under the terms of the GNU General Public License as published by
#    the Free Software Foundation; either version 2 of the License, or
```

```
#    (at your option) any later version.
#
#    This program is distributed in the hope that it will be useful,
#    but WITHOUT ANY WARRANTY; without even the implied warranty of
#    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
#    GNU General Public License for more details.
#
##############################################################################
# script for cron job
# this script use the "macro" lines of afick configuration file
# the goals are :
# - set the nice priority
# - truncate too long reports to avoid big mails
# - avoid mails if no changes detected
# - sent report to the specicified email adress
# - write reports to /var/log/afick
# - archive retention management

AFICK="/usr/bin/afick.pl"
PATH="/bin:/usr/bin"
LOGDIR="/var/log/afick"
LOGFILE="$LOGDIR/afick.log"
ERRORLOG="$LOGDIR/error.log"
CONFFILE="/etc/afick.conf"

# the default action is "update" (-u), you can also use "compare" (-k)
ACTION="-u"

##############################################################################
treat_log() {
        if [ -n "$VERBOSE_AFICK" ]
        then
                echo "# This is an automated report generated by Another File Integrity Checker on $FQDN $DATE."
        fi
```

```
        # "normal" afick output : changes result
        if [ -s $LOGFILE ]; then
                loglines=`wc -l $LOGFILE | awk '{ print $1 }'`
                if [ ${loglines:=0} -gt $LINES ]; then
                        echo "# TRUNCATED (!) output of the daily afick run:"
                        echo "# Output is $loglines lines, truncated to $LINES."
                        head -$LINES $LOGFILE
                        echo "# The full output can be found in $LOGFILE."
                else
                        echo "# Output of the daily afick run:"
                        cat $LOGFILE
                fi
        elif [ -n "$VERBOSE_AFICK" ]
        then
                echo "# afick detected no changes."
        fi

        # afick errors
        if [ -s $ERRORLOG ]; then
                errorlines=`wc -l $ERRORLOG | awk '{ print $1 }'`
                if [ ${errorlines:=0} -gt $LINES ]; then
                        echo "# TRUNCATED (!) output of errors produced:"
                        echo "# Error output is $errorlines lines, truncated to $LINES."
                        head -$LINES $ERRORLOG
                        echo "# The full output can be found in $ERRORLOG."
                else
                        echo "# Errors produced:"
                        cat $ERRORLOG
                fi
        elif [ -n "$VERBOSE_AFICK" ]
        then
                echo "# afick produced no errors."
        fi
```

```
        # check end of report (summary)
        if [ -s $LOGFILE ]; then
                summary=` grep "MD5 hash of" $LOGFILE `
                if [ -z "$summary" ]
                then
                        echo "WARNING: truncated report (no summary)"
                fi
        fi

}
################################################################################
# extract macro value from config file
macro () {
        key=$1
        grep -m 1 "^@@define $key" $CONFFILE |  sed -e "s/^@@define $key *//"
}
################################################################################
send_mail() {
        echo "$OUTPUT" | mail -s "[AFICK] Daily report for $FQDN" $MAILTO
}
################################################################################
send_nagios() {
        NAGIOS_STATUS=3 # UNKNOWN initial status
        if [ -s $LOGFILE ]
        then
                NAGIOS_MSG=`tail -4 $LOGFILE | head -1 | sed -e "s/^[^0-9]*\(.*changed\) (.*$/\1/ "`
                NUM_CHANGES=`echo $NAGIOS_MSG | cut -d " " -f 4`
                if [ $NUM_CHANGES -gt 0 ]
                then
                        if [ $NUM_CHANGES -ge $NAGIOS_CRITICAL_CHANGES ]
                        then
                                NAGIOS_STATUS=2 # CRITICAL
                        else
                                NAGIOS_STATUS=1 # WARNING
```

```
                        fi
                else
                        NAGIOS_STATUS=0 # OK
                fi
        fi
        HOST=`hostname`
        echo "${HOST}\t${NAGIOS_CHECK_NAME}\t${NAGIOS_STATUS}\t${NAGIOS_MSG}\n" | $NAGIOS_NSCA -H $NAGIOS_SERVER
-c $NAGIOS_CONFIG >/dev/null
}
################################################################################
# MAIN
################################################################################

[ -x $AFICK ] || exit 0

# hostname -f only exists on GNU systems,
# on others (HPUX, AIX, Solaris, Tru64), it return an error on stderr
# and a usage message on stdout
FQDN=`( hostname -f  || hostname ) 2>/dev/null |tail -1`
DATE=`date +"at %X on %x"`
MAILTO=`macro MAILTO`
LINES=`macro LINES`
VERBOSE=`macro VERBOSE`
REPORT=`macro REPORT`
NICE=`macro NICE`
BATCH=`macro BATCH`
MOUNT=`macro MOUNT`
NAGIOS=`macro NAGIOS`
NAGIOS_SERVER=`macro NAGIOS_SERVER`
NAGIOS_CONFIG=`macro NAGIOS_CONFIG`
NAGIOS_CHECK_NAME=`macro NAGIOS_CHECK_NAME`
NAGIOS_CRITICAL_CHANGES=`macro NAGIOS_CRITICAL_CHANGES`
NAGIOS_NSCA=`macro NAGIOS_NSCA`
ARCHIVE_RETENTION=`macro ARCHIVE_RETENTION`
```

```
# default values
[ -z "$FQDN" ] && FQDN=`hostname`
[ -z "$MAILTO" ] && MAILTO="root"
[ -z "$LINES" ] && LINES="1000"
[ -z "$VERBOSE" ] && VERBOSE=0
[ -z "$REPORT" ] && REPORT=1
[ -z "$NICE" ] && NICE=15
[ -z "$BATCH" ] && BATCH=1
[ -z "$NAGIOS" ] && NAGIOS=0
[ -z "$NAGIOS_SERVER" ] && NAGIOS="localhost"
[ -z "$NAGIOS_CONFIG" ] && NAGIOS_CONFIG="/etc/send_nsca.cfg"
[ -z "$NAGIOS_CHECK_NAME" ] && NAGIOS_CHECK_NAME="Another File Integrity Checker"
[ -z "$NAGIOS_CRITICAL_CHANGES" ] && NAGIOS_CRITICAL_CHANGES=2
[ -z "$NAGIOS_NCSA" ] && NAGIOS_NCSA="/usr/sbin/send_nsca"
[ -z "$ARCHIVE_RETENTION" ] && ARCHIVE_RETENTION=0

#echo "MAILTO=$MAILTO LINES=$LINES VERBOSE=$VERBOSE NICE=$NICE BATCH=$BATCH"

if [ "$BATCH" = "0" ]
then
        exit 0
fi

if [ "$VERBOSE" = "1" ]
then
        # verbose mail
        export VERBOSE_AFICK=1
fi

# the mount point must be already defined in /etc/fstab
if [ -n "$MOUNT" ]
then
        mount $MOUNT
fi
```

```
# launch command
nice -n $NICE $AFICK -c $CONFFILE $ACTION > $LOGFILE 2> $ERRORLOG

# archive retention
if [ "$ARCHIVE_RETENTION" != "0" ]
then
        echo "###############" >> $LOGFILE
        echo "# afick_archive" >> $LOGFILE
        /usr/bin/afick_archive.pl -c $CONFFILE -H -k $ARCHIVE_RETENTION >> $LOGFILE 2>> $ERRORLOG
fi

if [ -n "$MOUNT" ]
then
        umount $MOUNT
fi

# nagios ?
if [ "$NAGIOS" = "1" ]
then
        send_nagios
fi

if [ "$REPORT" = "0" ]
then
        # no report
        exit
fi

# filter output to send by mail
OUTPUT=`treat_log`
if [ "$VERBOSE" = "1" ]
then
        send_mail
else
```

```
        # skip comments and empty lines
        OUTPUT_FILTRE=`echo "$OUTPUT" | grep -v "^#" | grep -v "^$"`
        if [ -n "$OUTPUT_FILTRE" ]
        then
                send_mail
        fi
fi
```

Ce fichier permet d'intégrer Afick dans les tâches gérées par **cron**. Entre autre, il envoie un résumé par email à **root**.

L'adresse email à utiliser peut être modifiée dans la section **macros section** du fichier **/etc/afick.conf** :

```
#################
# macros section
#################
# used by cron job (afick_cron)
# define the mail adress to send cron job result
@@define MAILTO root@localhost
# truncate the result sended by mail to the number of lines (avoid too long mails)
@@define LINES 1000
# REPORT = 1 to enable mail reports, =0 to disable report
@@define REPORT 1
# VERBOSE = 1 to have one mail by run, =0 to have a mail only if changes are detected
@@define VERBOSE 0
# define the nice value : from 0 to 19 (priority of the job)
@@define NICE 18
# = 1 to allow cron job, = 0 to suppress cron job
@@define BATCH 1
# (optionnal, for unix) specify a file system to mount before the scan
# it must be defined in /etc/fstab
#@@define MOUNT /mnt/dist
# if set to 0, keep all archives, else define the number of days to keep
# with the syntaxe nS , n for a number, S for the scale
# (d for day, w for week, m for month, y for year)
```

```
# ex : for 5 months : 5m
@@define ARCHIVE_RETENTION 0

# send nagios messages by NSCA (= 1 to allow, = 0 to block)
@@define NAGIOS 0
# address of the nagios server to send messages to
@@define NAGIOS_SERVER my.nagios.server.org
# NSCA configuration file
# @@define NAGIOS_CONFIG /etc/send_nsca.cfg
# name used for nagios passive check on the nagios server side
@@define NAGIOS_CHECK_NAME Another File Integrity Checker
# number c of the changes that are considered critical => nagios state CRITICAL
# (0 changes => nagios state OK; 0> and <c changes => nagios state WARNING)
@@define NAGIOS_CRITICAL_CHANGES 2
# path to nsca binary
# @@define NAGIOS_NSCA /usr/sbin/send_nsca
```

Root Kits

Le Problématique

Un **rootkit** est un paquet logiciel qui permet à un utilisateur non autorisé d'obtenir les droits de **root**.

Les rootkits sont essentiellement de deux types, voire un mélange des deux :

- des modules du noyau,
- des paquets logiciels d'un utilisateur qui prennent la place de binaires système.

Les rootkits de type modules du noyau insèrent des modules qui remplacent des appels système et cachent des informations concernant certains processus spécifiques.

Les rootkits de type paquets logiciels remplacement en règle générale des binaires système tels que **ps**, **login** etc. Les binaires de remplacement

cachent des processus et des répertoires de l'attaquant.

Contre-Mesures

La mise en place de logiciels de vérification.

LAB #4 - rkhunter

rkhunter est un logiciel utilisé pour détecter les rootkits présents sur votre machine.

4.1 - Installation

L'installation de rkhunter se fait simplement en utilisant APT :

```
root@debian12:~# apt -y install rkhunter
```

4.2 - Utilisation

Lancez **rkhunter** en appelant son exécutable. A l'issu de son exécution, vous observerez un résumé :

```
root@debian12:~# rkhunter -c
...
System checks summary
=====================

File properties checks...
    Files checked: 144
    Suspect files: 1
```

```
Rootkit checks...
    Rootkits checked : 497
    Possible rootkits: 2

Applications checks...
    All checks skipped

The system checks took: 2 minutes and 57 seconds

All results have been written to the log file: /var/log/rkhunter.log

One or more warnings have been found while checking the system.
Please check the log file (/var/log/rkhunter.log)
```

```
root@debian12:~# more /var/log/rkhunter.log
[16:01:30] Running Rootkit Hunter version 1.4.6 on debian12
[16:01:30]
[16:01:30] Info: Start date is Sat Nov 29 04:01:30 PM CET 2025
[16:01:30]
[16:01:30] Checking configuration file and command-line options...
[16:01:30] Info: Detected operating system is 'Linux'
[16:01:30] Info: Found O/S name: Debian GNU/Linux 12 (bookworm)
[16:01:30] Info: Command line is /usr/bin/rkhunter -c
[16:01:30] Info: Environment shell is /bin/bash; rkhunter is using dash
[16:01:30] Info: Using configuration file '/etc/rkhunter.conf'
[16:01:30] Info: Installation directory is '/usr'
[16:01:30] Info: Using language 'en'
[16:01:30] Info: Using '/var/lib/rkhunter/db' as the database directory
[16:01:30] Info: Using '/usr/share/rkhunter/scripts' as the support script directory
[16:01:30] Info: Using '/usr/local/sbin /usr/local/bin /usr/sbin /usr/bin /sbin /bin /snap/bin /usr/libexec' as
the command directories
[16:01:30] Info: Using '/var/lib/rkhunter/tmp' as the temporary directory
[16:01:30] Info: No mail-on-warning address configured
[16:01:30] Info: X will be automatically detected
```

```
[16:01:30] Info: Found the 'basename' command: /usr/bin/basename
[16:01:30] Info: Found the 'diff' command: /usr/bin/diff
[16:01:30] Info: Found the 'dirname' command: /usr/bin/dirname
[16:01:30] Info: Found the 'file' command: /usr/bin/file
[16:01:30] Info: Found the 'find' command: /usr/bin/find
[16:01:30] Info: Found the 'ifconfig' command: /usr/sbin/ifconfig
[16:01:30] Info: Found the 'ip' command: /usr/sbin/ip
[16:01:30] Info: Found the 'ipcs' command: /usr/bin/ipcs
[16:01:30] Info: Found the 'ldd' command: /usr/bin/ldd
[16:01:30] Info: Found the 'lsattr' command: /usr/bin/lsattr
[16:01:30] Info: Found the 'lsmod' command: /usr/sbin/lsmod
[16:01:30] Info: Found the 'lsof' command: /usr/bin/lsof
[16:01:30] Info: Found the 'mktemp' command: /usr/bin/mktemp
[16:01:30] Info: Found the 'netstat' command: /usr/bin/netstat
[16:01:30] Info: Found the 'numfmt' command: /usr/bin/numfmt
[16:01:30] Info: Found the 'perl' command: /usr/bin/perl
[16:01:30] Info: Found the 'pgrep' command: /usr/bin/pgrep
[16:01:30] Info: Found the 'ps' command: /usr/bin/ps
[16:01:30] Info: Found the 'pwd' command: /usr/bin/pwd
[16:01:30] Info: Found the 'readlink' command: /usr/bin/readlink
[16:01:30] Info: Found the 'stat' command: /usr/bin/stat
[16:01:30] Info: Found the 'strings' command: /usr/bin/strings
[16:01:30] Info: System is not using prelinking
[16:01:30] Info: Using the '/usr/bin/sha256sum' command for the file hash checks
[16:01:30] Info: Stored hash values used hash function '/usr/bin/sha256sum'
[16:01:30] Info: Stored hash values did not use a package manager
[16:01:30] Info: The hash function field index is set to 1
[16:01:30] Info: No package manager specified: using hash function '/usr/bin/sha256sum'
[16:01:30] Info: Previous file attributes were stored
[16:01:30] Info: Enabled tests are: all
[16:01:30] Info: Disabled tests are: suspscan hidden_ports hidden_procs deleted_files packet_cap_apps apps
[16:01:30] Info: Found kernel symbols file '/proc/kallsyms'
[16:01:31] Info: Using syslog for some logging - facility/priority level is 'authpriv.warning'.
[16:01:31] Info: Found the 'logger' command: /usr/bin/logger
```

```
[16:01:31] Info: Using 'date' to process epoch second times
[16:01:31]
[16:01:31] Checking if the O/S has changed since last time...
--More--(2%)
```

Options de la commande

Les options de cette commande sont :

```
root@debian12:~# rkhunter --help

Usage: rkhunter {--check | --unlock | --update | --versioncheck |
                 --propupd [{filename | directory | package name},...] |
                 --list [{tests | {lang | languages} | rootkits | perl | propfiles}] |
                 --config-check | --version | --help} [options]

Current options are:
         --append-log                 Append to the logfile, do not overwrite
         --bindir <directory>...      Use the specified command directories
     -c, --check                      Check the local system
     -C, --config-check               Check the configuration file(s), then exit
  --cs2, --color-set2                 Use the second color set for output
         --configfile <file>          Use the specified configuration file
         --cronjob                    Run as a cron job
                                      (implies -c, --sk and --nocolors options)
         --dbdir <directory>          Use the specified database directory
         --debug                      Debug mode
                                      (Do not use unless asked to do so)
         --disable <test>[,<test>...] Disable specific tests
                                      (Default is to disable no tests)
         --display-logfile            Display the logfile at the end
         --enable  <test>[,<test>...] Enable specific tests
                                      (Default is to enable all tests)
```

```
         --hash {MD5 | SHA1 | SHA224 | SHA256 | SHA384 | SHA512 |
                NONE | <command>}     Use the specified file hash function
                                       (Default is SHA256)
     -h, --help                        Display this help menu, then exit
 --lang, --language <language>         Specify the language to use
                                       (Default is English)
         --list [tests | languages |   List the available test names, languages,
                rootkits | perl |      rootkit names, perl module status
                propfiles]             or file properties database, then exit
     -l, --logfile [file]              Write to a logfile
                                       (Default is /var/log/rkhunter.log)
         --noappend-log                Do not append to the logfile, overwrite it
         --nocf                        Do not use the configuration file entries
                                       for disabled tests (only valid with --disable)
         --nocolors                    Use black and white output
         --nolog                       Do not write to a logfile
--nomow, --no-mail-on-warning          Do not send a message if warnings occur
   --ns, --nosummary                   Do not show the summary of check results
 --novl, --no-verbose-logging          No verbose logging
         --pkgmgr {RPM | DPKG | BSD |  Use the specified package manager to obtain
                  BSDng | SOLARIS |    or verify file property values.
                  NONE}                (Default is NONE)
         --propupd [file | directory | Update the entire file properties database,
                   package]...         or just for the specified entries
     -q, --quiet                       Quiet mode (no output at all)
  --rwo, --report-warnings-only        Show only warning messages
   --sk, --skip-keypress               Don't wait for a keypress after each test
         --summary                     Show the summary of system check results
                                       (This is the default)
         --syslog [facility.priority]  Log the check start and finish times to syslog
                                       (Default level is authpriv.notice)
         --tmpdir <directory>          Use the specified temporary directory
         --unlock                      Unlock (remove) the lock file
         --update                      Check for updates to database files
```

```
   --vl, --verbose-logging             Use verbose logging (on by default)
     -V, --version                     Display the version number, then exit
         --versioncheck                Check for latest version of program
     -x, --autox                       Automatically detect if X is in use
     -X, --no-autox                    Do not automatically detect if X is in use
```

4.3 - Configuration

rkhunter peut être configuré soit par des options sur la ligne de commande soit par l'édition de son fichier de configuration **/etc/rkhunter.conf** :

```
root@debian12:~# more /etc/rkhunter.conf
#
# This is the main configuration file for Rootkit Hunter.
#
# You can modify this file directly, or you can create a local configuration
# file. The local file must be named 'rkhunter.conf.local', and must reside
# in the same directory as this file. Alternatively you can create a directory,
# named 'rkhunter.d', which also must be in the same directory as this
# configuration file. Within the 'rkhunter.d' directory you can place further
# configuration files. There is no restriction on the file names used, other
# than they must end in '.conf'.
#
# Please modify the configuration file(s) to your own requirements. It is
# recommended that the command 'rkhunter -C' is run after any changes have
# been made.
#
# Please review the documentation before posting bug reports or questions.
# To report bugs, provide patches or comments, please go to:
# http://rkhunter.sourceforge.net
#
# To ask questions about rkhunter, please use the 'rkhunter-users' mailing list.
# Note that this is a moderated list, so please subscribe before posting.
#
```

```
# In the configuration files, lines beginning with a hash (#), and blank lines,
# are ignored. Also, end-of-line comments are not supported.
#
# Any of the configuration options may appear more than once. However, several
# options only take one value, and so the last one seen will be used. Some
# options are allowed to appear more than once, and the text describing the
# option will say if this is so. These configuration options will, in effect,
# have their values concatenated together. To delete a previously specified
# option list, specify the option with no value (that is, a null string).
#
# Some of the options are space-separated lists, others, typically those
# specifying pathnames, are newline-separated lists. These must be entered
# as one item per line. Quotes must not be used to surround the pathname.
#
# For example, to specify two pathnames, '/tmp/abc' and '/tmp/xyz', for an
# option:         XXX=/tmp/abc                (correct)
#                 XXX=/tmp/xyz
#
#                 XXX="/tmp/abc"              (incorrect)
#                 XXX="/tmp/xyz"
#
#                 XXX=/tmp/abc  /tmp/xyz      (incorrect)
#    or           XXX="/tmp/abc  /tmp/xyz"    (incorrect)
#    or           XXX="/tmp/abc"  "/tmp/xyz"  (incorrect)
#
# The last three examples are being configured as space-separated lists,
# which is incorrect, generally, for options specifying pathnames. They
# should be configured with one entry per line as in the first example.
#
# If wildcard characters (globbing) are allowed for an option, then the
# text describing the option will say so. Any globbing character explicitly
# required in a pathname should be escaped.
#
--More--(5%)
```

LAB #5 - chkrootkit

chkrootkit est un autre logiciel utilisé pour détecter les rootkits présents sur votre machine.

5.1 - Installation

L'installation de chkrootkit se fait simplement en utilisant APT :

```
root@debian12:~# apt install -y chkrootkit
```

5.2 - Utilisation

Lancez **chkrootkit** en appelant son exécutable.

```
root@debian12:~# chkrootkit
ROOTDIR is `/'
Checking `amd'...                                           not found
Checking `basename'...                                      not infected
Checking `biff'...                                          not found
Checking `chfn'...                                          not infected
Checking `chsh'...                                          not infected
Checking `cron'...                                          not infected
Checking `crontab'...                                       not infected
Checking `date'...                                          not infected
Checking `du'...                                            not infected
Checking `dirname'...                                       not infected
Checking `echo'...                                          not infected
Checking `egrep'...                                         not infected
Checking `env'...                                           not infected
Checking `find'...                                          not infected
```

```
Checking `fingerd'...                                 not found
Checking `gpm'...                                     not found
Checking `grep'...                                    not infected
Checking `hdparm'...                                  not found
Checking `su'...                                      not infected
Checking `ifconfig'...                                not infected
Checking `inetd'...                                   not infected
Checking `inetdconf'...                               not found
Checking `identd'...                                  not found
Checking `init'...                                    not infected
Checking `killall'...                                 not infected
Checking `ldsopreload'...                             not infected
Checking `login'...                                   not infected
Checking `ls'...                                      not infected
Checking `lsof'...                                    not infected
Checking `mail'...                                    not infected
Checking `mingetty'...                                not found
Checking `netstat'...                                 not infected
Checking `named'...                                   not found
Checking `passwd'...                                  not infected
Checking `pidof'...                                   not infected
Checking `pop2'...                                    not found
Checking `pop3'...                                    not found
Checking `ps'...                                      not infected
Checking `pstree'...                                  not infected
Checking `rpcinfo'...                                 not found
Checking `rlogind'...                                 not found
Checking `rshd'...                                    not found
Checking `slogin'...                                  not infected
Checking `sendmail'...                                not infected
Checking `sshd'...                                    not infected
Checking `syslogd'...                                 not found
Checking `tar'...                                     not infected
Checking `tcpd'...                                    not found
```

```
Checking `tcpdump'...                                    not infected
Checking `top'...                                        not infected
Checking `telnetd'...                                    not found
Checking `timed'...                                      not found
Checking `traceroute'...                                 not infected
Checking `vdir'...                                       not infected
Checking `w'...                                          not infected
Checking `write'...                                      not infected
Checking `aliens'...                                     started
Searching for suspicious files in /dev...                not found
Searching for known suspicious directories...            not found
Searching for known suspicious files...                  not found
Searching for sniffer's logs...                          not found
Searching for HiDrootkit rootkit...                      not found
Searching for t0rn rootkit...                            not found
Searching for t0rn v8 (or variation)...                  not found
Searching for Lion rootkit...                            not found
Searching for RSHA rootkit...                            not found
Searching for RH-Sharpe rootkit...                       not found
Searching for Ambient (ark) rootkit...                   not found
Searching for suspicious files and dirs...               WARNING

WARNING: The following suspicious files and directories were found:
/usr/lib/jvm/.java-1.17.0-openjdk-amd64.jinfo
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/noentry/.htaccess
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/digest/.htaccess
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/digest/.htpasswd
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/digest_anon/.htaccess
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/digest_anon/.htpasswd
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/digest_time/.htaccess
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/digest_time/.htpasswd
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/digest_wrongrelm/.htaccess
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/digest_wrongrelm/.htpasswd
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/basic/authz_owner/.htaccess
```

```
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/basic/authz_owner/.htpasswd
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/basic/file/.htaccess
/usr/lib/python3/dist-packages/fail2ban/tests/files/config/apache-auth/basic/file/.htpasswd
/usr/lib/ruby/vendor_ruby/rubygems/tsort/.document
/usr/lib/ruby/vendor_ruby/rubygems/optparse/.document
/usr/lib/ruby/vendor_ruby/rubygems/ssl_certs/.document
/usr/lib/ruby/gems/3.1.0/gems/typeprof-0.21.2/vscode/.vscodeignore
/usr/lib/ruby/gems/3.1.0/gems/typeprof-0.21.2/vscode/.vscode
/usr/lib/ruby/gems/3.1.0/gems/typeprof-0.21.2/vscode/.gitignore
/usr/lib/libreoffice/share/.registry

Searching for LPD Worm...                                  not found
Searching for Ramen Worm rootkit...                        not found
Searching for Maniac rootkit...                            not found
Searching for RK17 rootkit...                              not found
Searching for Ducoci rootkit...                            not found
Searching for Adore Worm...                                not found
Searching for ShitC Worm...                                not found
Searching for Omega Worm...                                not found
Searching for Sadmind/IIS Worm...                          not found
Searching for MonKit...                                    not found
Searching for Showtee rootkit...                           not found
Searching for OpticKit...                                  not found
Searching for T.R.K...                                     not found
Searching for Mithra rootkit...                            not found
Searching for OBSD rootkit v1...                           not tested
Searching for LOC rootkit...                               not found
Searching for Romanian rootkit...                          not found
Searching for HKRK rootkit...                              not found
Searching for Suckit rootkit...                            not found
Searching for Volc rootkit...                              not found
Searching for Gold2 rootkit...                             not found
Searching for TC2 rootkit...                               not found
Searching for Anonoying rootkit...                         not found
```

```
Searching for ZK rootkit...                                not found
Searching for ShKit rootkit...                             not found
Searching for AjaKit rootkit...                            not found
Searching for zaRwT rootkit...                             not found
Searching for Madalin rootkit...                           not found
Searching for Fu rootkit...                                not found
Searching for Kenga3 rootkit...                            not found
Searching for ESRK rootkit...                              not found
Searching for rootedoor...                                 not found
Searching for ENYELKM rootkit...                           not found
Searching for common ssh-scanners...                       not found
Searching for Linux/Ebury 1.4 - Operation Windigo...       not tested
Searching for Linux/Ebury 1.6...                           not found
Searching for 64-bit Linux Rootkit...                      not found
Searching for 64-bit Linux Rootkit modules...              not found
Searching for Mumblehard...                                not found
Searching for Backdoor.Linux.Mokes.a...                    not found
Searching for Malicious TinyDNS...                         not found
Searching for Linux.Xor.DDoS...                            not found
Searching for Linux.Proxy.1.0...                           not found
Searching for CrossRAT...                                  not found
Searching for Hidden Cobra...                              not found
Searching for Rocke Miner rootkit...                       not found
Searching for PWNLNX4 lkm rootkit...                       not found
Searching for PWNLNX6 lkm rootkit...                       not found
Searching for Umbreon lrk...                               not found
Searching for Kinsing.a backdoor rootkit...                not found
Searching for RotaJakiro backdoor rootkit...               not found
Searching for Syslogk LKM rootkit...                       not found
Searching for Kovid LKM rootkit...                         not tested
Searching for suspect PHP files...                         not found
Searching for zero-size shell history files...             not found
Searching for hardlinked shell history files...            not found
Checking `aliens'...                                       finished
```

```
Checking `asp'...                                           not infected
Checking `bindshell'...                                     not found
Checking `lkm'...                                           started
Searching for Adore LKM...                                  not tested
Searching for sebek LKM (Adore based)...                    not tested
Searching for knark LKM rootkit...                          not found
Searching for for hidden processes with chkproc...          not found
Searching for for hidden directories using chkdirs...       not found
Checking `lkm'...                                           finished
Checking `rexedcs'...                                       not found
Checking `sniffer'...                                       WARNING

WARNING: Output from ifpromisc:
lo: not promisc and no packet sniffer sockets
ens18: PACKET SNIFFER(/usr/sbin/NetworkManager[4344])

Checking `w55808'...                                        not found
Checking `wted'...                                          WARNING

WARNING: output from chkwtmp:
1 deletion(s) between Sat Oct 14 07:48:45 2023 and Mon Oct 16 08:43:48 2023

Checking `scalper'...                                       not found
Checking `slapper'...                                       not found
Checking `z2'...                                            not found
Checking `chkutmp'...                                       not found
Checking `OSX_RSPLUG'...                                    not tested
```

Options de la commande

Les options de cette commande sont :

```
root@debian12:~# chkrootkit --help
```

```
Usage: /usr/sbin/chkrootkit [options] [test ...]
Options:
        -h               show this help and exit
        -V               show version information and exit
        -l               show available tests and exit
        -d               debug
        -q               quiet mode
        -x               expert mode
        -e 'FILE1 FILE2'  exclude files/dirs from results. Must be followed by a space-separated list of
files/dirs.
                         Read /usr/share/doc/chkrootkit/README.FALSE-POSITIVES first.
        -s REGEXP        filter results of sniffer test through 'grep -Ev REGEXP' to exclude expected
                         PACKET_SNIFFERs. Read /usr/share/doc/chkrootkit/README.FALSE-POSITIVES first.
        -r DIR           use DIR as the root directory
        -p DIR1:DIR2:DIRN path for the external commands used by chkrootkit
        -n               skip NFS mounted dirs
```

5.3 - Configuration

chkrootkit peut être configuré soit par des options sur la ligne de commande soit par l'édition de son fichier de configuration **/etc/chkrootkit/chkrootkit.conf** :

```
root@debian12:~# more /etc/chkrootkit/chkrootkit.conf
### This file is used to configure the daily cron job for chkrootkit(1)
## It is sourced by chkrootkit-daily so needs to be a valid
## shell script.
##
## The majority of the options allow the output of chkrootkit to be
## filtered (changed) and/or ignored to hide false positives.

## Whether the daily cron job should run chkrootkit at all
# true/false, default: true
RUN_DAILY="true"
```

```
## Arguments to pass to chkrootkit (default: "").
# See chkrootkit(1) for details, but particularly useful are
#  "-q" (especially useful if you set DIFF_MODE=false above )
#  "-e" and "-s" (which are yet another way to hide output)
# The default is to pass no arguments so you see all output: this is
# particularly useful with DIFF_MODE=true as it gives context when the
# output changes, but DIFF_MODE hides the text that does not change
RUN_DAILY_OPTS=""

## Whether to show changes since last run (true/false, default: true)
# true means you will see how the entire differs (using diff(1)) to
# the 'expected' output in /var/log/chkrootkit/log.expected
# if that file does not exist you will see the whole output.
#
# If set to false you see the whole output every day - if you do set
# DIFF_MODE to "false" you probably also want RUN_DAILY_OPTS="-q"
#
DIFF_MODE="true"

### The results of chkrootkit are passed through $FILTER and $IGNORE_FILE

## FILTER is a way of changing output to make it stable or hide it
## completely (especially useful when DIFF_MODE=true
#
# FILTER can be any shell command which will be piped unfiltered output
# on stdin and anything written to stdout will become the new,
# filtered, output.
#
# The default uses sed to do two things
# 1) stops message (from ifpromisc, run by the 'sniffer' test) about
#     'usual' network managers changing if their pid, interface name,
#     or order changes
# 2) stops list of processes not using utmp (from chkutmp) changing if
#     their pid changes
```

```
# To disable both of these defaults you can set this to "" or "cat"
FILTER="sed -re 's![[:alnum:]]+: PACKET SNIFFER\(((/lib/systemd/systemd-
networkd|(/usr)?/sbin/(dhclient|dhcpc?d[0-9]*|wpa_supplicant|NetworkManager))\[[0-9]+\](, )?)+\)!<interface>:
PACKET SNIFFER\([systemd-net
workd|dhclient|dhcpd|dhcpcd|wpa_supplicant|NetworkManager]{PID}\)!' -e 's/(! [[:alnum:]+-]+)\s+[0-9]+/\1 {PID}/'"

## If $IGNORE_FILE exists then lines of output matching patterns that
## appear in $IGNORE_FILE are removed from the output. Each line in
## $IGNORE_FILE interpreted as an extended regexp, see grep(1). If
## $IGNORE_FILE is empty (the default) or does not exist then no
## filtering is done.
#
--More--(63%)
```

Dernièrement, le fichier **/etc/chkrootkit/chkrootkit.ignore** permet à l'administrateur système de supprimer les faux positifs de la sortie de l'outil d'analyse de rootkits chkrootkit.
