Version : **2021.01**

Dernière mise-à-jour : 2021/03/03 07:13

LRF406 - Système de Fichiers

Contenu du module

- LAB #3 - Mise en place de rkhunter
  - Installation
  - Les options de la commande
  - Utilisation
  - Configuration

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@centos7 ~]# cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Sat Apr 30 11:27:02 2016
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=e65fe7da-cda8-4f5a-a827-1b5cabe94bed /                       xfs     defaults        0 0
UUID=2d947276-66e8-41f4-8475-b64b67d7a249 /boot                   xfs     defaults        0 0
UUID=3181601a-7295-4ef0-a92c-f21f76b18e64 swap                    swap    defaults        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	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
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

Option	Systèmes de Fichier	Description	Valeur par Défaut
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 repertoires **/sbin**, **/bin**, **/usr/sbin** et **/usr/bin** ne doivent pas posseder des droits **standards** superieurs a 755.

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

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.

Préparation

Créez la partition /dev/sda5 :

```
[root@centos7 ~]# fdisk /dev/sda
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.


Command (m for help): n
Partition type:
   p   primary (3 primary, 0 extended, 1 free)
```

```
   e   extended
Select (default e): e
Selected partition 4
First sector (25083904-41943039, default 25083904):
Using default value 25083904
Last sector, +sectors or +size{K,M,G} (25083904-41943039, default 41943039):
Using default value 41943039
Partition 4 of type Extended and of size 8 GiB is set

Command (m for help): n
All primary partitions are in use
Adding logical partition 5
First sector (25085952-41943039, default 25085952):
Using default value 25085952
Last sector, +sectors or +size{K,M,G} (25085952-41943039, default 41943039): +500M
Partition 5 of type Linux and of size 500 MiB is set

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
[root@centos7 ~]# partprobe
```

Important : L'étape ci-dessus est très importante parce que elle permet de s'assurer qu'aucune donnée ne reste sur la partition.

Initialisez la partition avec LUKS :

```
[root@centos7 ~]# cryptsetup --verbose --verify-passphrase luksFormat /dev/sda5

WARNING!
========
This will overwrite data on /dev/sda5 irrevocably.

Are you sure? (Type uppercase yes): YES
Enter passphrase: fenestros123456789
Verify passphrase: fenestros123456789
Command successful.
```

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

Ouvrez la partition LUKS en lui donnant le nom **sda5** :

```
[root@centos7 ~]# cryptsetup luksOpen /dev/sda5 sda5
Enter passphrase for /dev/sda5: fenestros123456789
```

Vérifiez que le système voit la partition :

```
[root@centos7 ~]# ls -l /dev/mapper | grep sda5
lrwxrwxrwx. 1 root root       7 Oct  5 20:18 sda5 -> ../dm-0
```

Créez maintenant un système de fichiers sur **/dev/mapper/sda5** :

```
[root@centos7 ~]# mkfs.xfs /dev/mapper/sda5
meta-data=/dev/mapper/sda5      isize=256    agcount=4, agsize=12672 blks
         =                       sectsz=512   attr=2, projid32bit=1
```

```
         =                       crc=0        finobt=0
data     =                       bsize=4096   blocks=50688, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=0
log      =internal log           bsize=4096   blocks=853, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
```

Montez la partition LUKS :

```
[root@centos7 ~]# mkdir /mnt/sda5
[root@centos7 ~]# mount /dev/mapper/sda5 /mnt/sda5
```

Vérifiez la présence du montage :

```
[root@centos7 ~]# df -h | grep sda5
/dev/mapper/sda5  195M   11M  185M   6% /mnt/sda5
```

Créez le fichier **/etc/crypttab/** :

```
[root@centos7 ~]# vi /etc/crypttab
[root@centos7 ~]# cat /etc/crypttab
sda5 /dev/sda5 none
```

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

```
[root@centos7 ~]# cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Sun Mar  8 12:38:10 2015
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
```

```
#
UUID=b35de665-5ec8-4226-a533-58a1b567ac91 /                       xfs     defaults        1 1
UUID=e8d3bd48-1386-411c-9675-41c3f8f1a309 /boot                   xfs     defaults        1 2
UUID=11a4d11d-81e4-46a7-82e0-7796cd597dc9 swap                    swap    defaults        0 0
/dev/mapper/sda5                          /mnt/sda5              xfs     defaults          1 2
```

Restaurer les SC par défaut de SELinux :

```
[root@centos7 ~]# /sbin/restorecon -v -R /mnt/sda5
/sbin/restorecon reset /mnt/sda5 context system_u:object_r:unlabeled_t:s0->system_u:object_r:mnt_t:s0
```

Redémarrez votre machine virtuelle :

```
[root@centos7 ~]# shutdown -r now
```

> **Important** : Lors du démarrage de la machine virtuelle, le système devrait vous demander d'entrer la passphrase **fenestros123456789** pour permettre le montage de /dev/sda5.

Ajouter une deuxième Passphrase

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

```
[root@centos7 ~]# cryptsetup luksAddKey /dev/sda5
Enter any existing passphrase: fenestros123456789
Enter new passphrase for key slot: redhat123456789
Verify passphrase: redhat123456789
[root@centos7 ~]#
```

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

Supprimer une Passphrase

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

```
[root@centos7 ~]# cryptsetup luksRemoveKey /dev/sda5
Enter passphrase to be deleted: redhat123456789
```

LAB #2 - Mise en place du File Integrity Checker Afick

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.

Installation

Commencez par installer les dépendances d'Afick :

```
[root@centos7 ~]# yum install perl perl-Digest-MD5
```

Téléchargez la dernière version d'Afick :

```
[root@centos7 ~]# wget https://sourceforge.net/projects/afick/files/afick/3.6.0/afick-3.6.0-1.noarch.rpm
```

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

```
[root@centos7 ~]# yum localinstall afick-3.6.0-1.noarch.rpm --nogpgcheck
```

Configuration

La configuration d'Afick est contenu 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
# 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

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

# 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 - Veuillez 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

```
exclude_suffix := log LOG html htm HTM txt TXT xml
```

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

#all:    p+d+i+n+u+g+s+b+m+c+md5
#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
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
/etc/aliases.db ETC - md5
# /etc/mail/statistics ETC - md5
/etc/motd ETC
# /etc/ntp/drift ETC - md5
# /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

/sbin   MyRule
```

```
/usr/bin    MyRule
/usr/sbin   MyRule
/usr/lib    MyRule
/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
...
```

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**.

Utilisation

Commencez par créer la base de données d'afick :

```
[root@centos7 ~]# afick -i
# Afick (3.6.0) init at 2018/06/17 15:28:43 with options (/etc/afick.conf):
# archive:=/var/lib/afick/archive
# database:=/var/lib/afick/afick
# 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
# history:=/var/lib/afick/history
# max_checksum_size:=10000000
# running_files:=1
# timing:=1
# dbm:=Storable

# ###################################################################
# MD5 hash of /var/lib/afick/afick => uoNMcB5r9Jb7MUKJvDWL7g

# Hash database created successfully. 40262 files entered.
# user time : 84.76; system time : 39.16; real time : 552
You have new mail in /var/spool/mail/root
```

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**

Automatiser Afick

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

```
[root@centos7 ~]# cat /etc/cron.daily/afick_cron
#!/bin/sh
##############################################################################
#   afick_cron
#      it's a part of the afick project
#
#    Copyright (C) 2002, 2003 by Eric Gerbier
#    Bug reports to: gerbier@users.sourceforge.net
#    $Id$
#
#    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
    /usr/bin/afick_archive.pl -c $CONFFILE -H -k $ARCHIVE_RETENTION
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 systèmes et cachent des informations concernant certains processus spécifiques.

Les rootkits de type paquets logiciels remplacement en règle générale des binaires système tels **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 #3 - Mise en place de rkhunter

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

Installation

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

```
[root@centos7 ~]# yum install rkhunter
```

Les options de la commande

Les options de cette commande sont :

```
[root@centos7 ~]# 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
```

Utilisation

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

```
[root@centos7 ~]# rkhunter -c
...
System checks summary
=====================

File properties checks...
    Required commands check failed
    Files checked: 135
    Suspect files: 4

Rootkit checks...
    Rootkits checked : 503
    Possible rootkits: 0

Applications checks...
    All checks skipped

The system checks took: 9 minutes and 23 seconds

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

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

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**.

<html>

</html>