Version : **2024.01**

Dernière mise-à-jour : 2024/12/08 15:57

Topic 110: Security

Contenu du Module

        - Configuration
            - Serveur
        - Utilisation
        - Tunnels SSH
    - SCP
        - Introduction
        - Utilisation
        - Mise en place des clefs

Le Fichier /etc/nologin

Dans le cas où le fichier **/etc/nologin** existe, toute connexion au système est interdite, sauf pour l'utilisateur **root**.

La Commande find

Cette commande sert à rechercher un ou des fichiers dans le répertoire courant ou le répertoire spécifié en argument :

```
[trainee@centos7 ~]$ touch aac
[trainee@centos7 ~]$ find acc
find: ‘acc’: No such file or directory
[trainee@centos7 ~]$ find aac
aac
```

Important : Notez que si le fichier n'existe pas le système vous en informe clairement.
Notez aussi que ce fichier existe le système vous en informe en vous indiquant son nom.

Une utilisation courante de la commande find dans le contexte de la sécurité du système est la recherche d'exécutables ayant le SGID ou SUID bit :

```
# find / -type f \( -perm -4000 -o -perm -2000 \) -exec ls {} \; [Entrée]
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **find** pour visualiser les options de la commande.

La Commande lsof

La commande **lsof** affiche des informations sur les fichiers ouverts par des processus :

```
[trainee@centos7 tmp]$ su -
Password: fenestros
Last login: Thu Sep 29 06:24:28 CEST 2016 on pts/0
[root@centos7 ~]# lsof | more
COMMAND     PID   TID           USER   FD      TYPE             DEVICE  SIZE/OFF       NODE NAME
systemd       1                 root  cwd       DIR                8,2      4096        128 /
systemd       1                 root  rtd       DIR                8,2      4096        128 /
systemd       1                 root  txt       REG                8,2   1494056   33607555
/usr/lib/systemd/systemd
systemd       1                 root  mem       REG                8,2     20032   17241167
/usr/lib64/libuuid.so.1.3.0
systemd       1                 root  mem       REG                8,2    252696   20114965
/usr/lib64/libblkid.so.1.1.0
systemd       1                 root  mem       REG                8,2     90632   18751025
/usr/lib64/libz.so.1.2.7
systemd       1                 root  mem       REG                8,2     19888   17326380
/usr/lib64/libattr.so.1.1.0
systemd       1                 root  mem       REG                8,2     19520   17239651
```

```
/usr/lib64/libdl-2.17.so
systemd       1                 root  mem       REG               8,2    153192   18023397
/usr/lib64/liblzma.so.5.0.99
systemd       1                 root  mem       REG               8,2    398272   18105157
/usr/lib64/libpcre.so.1.2.0
systemd       1                 root  mem       REG               8,2   2107816   17239525
/usr/lib64/libc-2.17.so
systemd       1                 root  mem       REG               8,2    142304   17240882
/usr/lib64/libpthread-2.17.so
systemd       1                 root  mem       REG               8,2     88720   17511831
/usr/lib64/libgcc_s-4.8.5-20150702.so.1
systemd       1                 root  mem       REG               8,2     44096   17241011
/usr/lib64/librt-2.17.so
systemd       1                 root  mem       REG               8,2    260784   20114966
/usr/lib64/libmount.so.1.1.0
systemd       1                 root  mem       REG               8,2     91768   18023376
/usr/lib64/libkmod.so.2.2.10
systemd       1                 root  mem       REG               8,2    118792   17326381
/usr/lib64/libaudit.so.1.0.0
systemd       1                 root  mem       REG               8,2     61648   18105139
/usr/lib64/libpam.so.0.83.1
systemd       1                 root  mem       REG               8,2     20024   17326423
/usr/lib64/libcap.so.2.22
systemd       1                 root  mem       REG               8,2    147120   18447456
/usr/lib64/libselinux.so.1
systemd       1                 root  mem       REG               8,2    164440   21744303 /usr/lib64/ld-2.17.so
systemd       1                 root  mem       REG               8,2   1333123   50334524
/etc/selinux/targeted/contexts/files/file_contexts.bin
--More--
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **lsof** pour visualiser les options de la commande.

La Commande ulimit

Les ressources disponibles aux utilisateurs peuvent être limitées par l'utilisation de la commande **ulimit**.

La commande **ulimit** gère deux types de limite, la limite *hard* en utilisant l'option **-H** et la limite *soft* en utilisant l'option **-S**. Seul root peut positionner une limite *hard* et ceci à condition que la limite ne dépasse pas les ressources réelles.

La limite *soft* est la limite imposée à l'utilisateur par défaut tandis que la limite *hard* est la limite que l'utilisateur peut atteindre en utilisant la commande ulimit lui-même.

L'utilisateur root peut paramétrer les limites accordées en éditant la fichier **/etc/security/limits.conf** :

```
[root@centos7 ~]# cat /etc/security/limits.conf
# /etc/security/limits.conf
#
#This file sets the resource limits for the users logged in via PAM.
#It does not affect resource limits of the system services.
#
#Also note that configuration files in /etc/security/limits.d directory,
#which are read in alphabetical order, override the settings in this
#file in case the domain is the same or more specific.
#That means for example that setting a limit for wildcard domain here
#can be overriden with a wildcard setting in a config file in the
#subdirectory, but a user specific setting here can be overriden only
#with a user specific setting in the subdirectory.
#
#Each line describes a limit for a user in the form:
#
#<domain>        <type>  <item>  <value>
```

```
#
#Where:
#<domain> can be:
#        - a user name
#        - a group name, with @group syntax
#        - the wildcard *, for default entry
#        - the wildcard %, can be also used with %group syntax,
#                for maxlogin limit
#
#<type> can have the two values:
#        - "soft" for enforcing the soft limits
#        - "hard" for enforcing hard limits
#
#<item> can be one of the following:
#        - core - limits the core file size (KB)
#        - data - max data size (KB)
#        - fsize - maximum filesize (KB)
#        - memlock - max locked-in-memory address space (KB)
#        - nofile - max number of open file descriptors
#        - rss - max resident set size (KB)
#        - stack - max stack size (KB)
#        - cpu - max CPU time (MIN)
#        - nproc - max number of processes
#        - as - address space limit (KB)
#        - maxlogins - max number of logins for this user
#        - maxsyslogins - max number of logins on the system
#        - priority - the priority to run user process with
#        - locks - max number of file locks the user can hold
#        - sigpending - max number of pending signals
#        - msgqueue - max memory used by POSIX message queues (bytes)
#        - nice - max nice priority allowed to raise to values: [-20, 19]
#        - rtprio - max realtime priority
#
#<domain>      <type>  <item>         <value>
```

```
#

#*               soft    core            0
#*               hard    rss             10000
#@student        hard    nproc           20
#@faculty        soft    nproc           20
#@faculty        hard    nproc           50
#ftp             hard    nproc           0
#@student        -       maxlogins       4

# End of file
```

Important : La valeur de la limite peut être un **nombre** ou le mot **unlimited**.

Par exemple, si root inscrit les deux ligne suivantes dans le fichier /etc/security/limits.conf :

```
...
trainee                 soft        nofile          1024
trainee                 hard        nofile          4096
...
```

la limite du nombre de fichiers ouverts simultanément par trainee est de 1 024. Par contre, trainee a la possibilité d'augmenter cette limite jusqu'à 4 096 en utilisant la commande suivante :

```
$ ulimit -n 4096
```

Pour consulter la liste des limites actuelles, il convient d'utiliser la commande ulimit avec l'option **-a** :

```
[root@centos7 ~]# ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
```

```
scheduling priority             (-e) 0
file size                   (blocks, -f) unlimited
pending signals                 (-i) 6929
max locked memory           (kbytes, -l) 64
max memory size             (kbytes, -m) unlimited
open files                      (-n) 1024
pipe size                (512 bytes, -p) 8
POSIX message queues         (bytes, -q) 819200
real-time priority              (-r) 0
stack size                  (kbytes, -s) 8192
cpu time                   (seconds, -t) unlimited
max user processes              (-u) 6929
virtual memory              (kbytes, -v) unlimited
file locks                      (-x) unlimited
```

Options de la commande

Les options de **ulimit** sont :

```
[root@centos7 ~]# help ulimit
ulimit: ulimit [-SHacdefilmnpqrstuvx] [limit]
    Modify shell resource limits.
    Provides control over the resources available to the shell and processes
    it creates, on systems that allow such control.
    Options:
      -S    use the `soft' resource limit
      -H    use the `hard' resource limit
      -a    all current limits are reported
      -b    the socket buffer size
      -c    the maximum size of core files created
      -d    the maximum size of a process's data segment
      -e    the maximum scheduling priority (`nice')
      -f    the maximum size of files written by the shell and its children
```

```
      -i    the maximum number of pending signals
      -l    the maximum size a process may lock into memory
      -m    the maximum resident set size
      -n    the maximum number of open file descriptors
      -p    the pipe buffer size
      -q    the maximum number of bytes in POSIX message queues
      -r    the maximum real-time scheduling priority
      -s    the maximum stack size
      -t    the maximum amount of cpu time in seconds
      -u    the maximum number of user processes
      -v    the size of virtual memory
      -x    the maximum number of file locks
    If LIMIT is given, it is the new value of the specified resource; the
    special LIMIT values `soft', `hard', and `unlimited' stand for the
    current soft limit, the current hard limit, and no limit, respectively.
    Otherwise, the current value of the specified resource is printed.  If
    no option is given, then -f is assumed.
    Values are in 1024-byte increments, except for -t, which is in seconds,
    -p, which is in increments of 512 bytes, and -u, which is an unscaled
    number of processes.
    Exit Status:
    Returns success unless an invalid option is supplied or an error occurs.
```

La Commande sudo

Important : Afin de mettre en pratique les exemples qui suivent, vous devez être connecté à votre système en tant que root. Tapez donc la commande **exit** pour sortir de l'environnement de **fenestros2**.

La commande **sudo** permet à un utilisateur autorisé d'exécuter une commande en tant que **root** ou en tant qu'un autre utilisateur. Lors de l'exécution

de la commande, l'UID et le GID éffectifs et réels sont ceux de l'identité de l'utilisateur cible. L'utilisation de la commande **sudo** est une façon simple de déléguer des tâches administratives à d'autres utilisateurs sans communiquer le mot de passe de **root** et sans placer un SUID bit sur l'exécutable. La commande **sudo** est configurée grâce au fichier **/etc/sudoers**.

Saisissez la commande suivante :

```
[root@centos5 ~]# cat /etc/sudoers
## Sudoers allows particular users to run various commands as
## the root user, without needing the root password.
##
## Examples are provided at the bottom of the file for collections
## of related commands, which can then be delegated out to particular
## users or groups.
##
## This file must be edited with the 'visudo' command.

## Host Aliases
## Groups of machines. You may prefer to use hostnames (perhap using
## wildcards for entire domains) or IP addresses instead.
# Host_Alias     FILESERVERS = fs1, fs2
# Host_Alias     MAILSERVERS = smtp, smtp2

## User Aliases
## These aren't often necessary, as you can use regular groups
## (ie, from files, LDAP, NIS, etc) in this file - just use %groupname
## rather than USERALIAS
# User_Alias ADMINS = jsmith, mikem


## Command Aliases
## These are groups of related commands...

## Networking
#Cmnd_Alias NETWORKING = /sbin/route, /sbin/ifconfig, /bin/ping, /sbin/dhclient, /usr/bin/net, /sbin/iptables,
```

```
/usr/bin/rfcomm, /usr/bin/wvdial, /sbin/iwconfig, /sbin/mii-tool

## Installation and management of software
#Cmnd_Alias SOFTWARE = /bin/rpm, /usr/bin/up2date, /usr/bin/yum

## Services
#Cmnd_Alias SERVICES = /sbin/service, /sbin/chkconfig

## Updating the locate database
#Cmnd_Alias LOCATE = /usr/bin/updatedb

## Storage
#Cmnd_Alias STORAGE = /sbin/fdisk, /sbin/sfdisk, /sbin/parted, /sbin/partprobe, /bin/mount, /bin/umount

## Delegating permissions
#Cmnd_Alias DELEGATING = /usr/sbin/visudo, /bin/chown, /bin/chmod, /bin/chgrp

## Processes
#Cmnd_Alias PROCESSES = /bin/nice, /bin/kill, /usr/bin/kill, /usr/bin/killall

## Drivers
#Cmnd_Alias DRIVERS = /sbin/modprobe

# Defaults specification

#
# Disable "ssh hostname sudo <cmd>", because it will show the password in clear.
#         You have to run "ssh -t hostname sudo <cmd>".
#
Defaults    requiretty

#
# Refuse to run if unable to disable echo on the tty. This setting should also be
# changed in order to be able to use sudo without a tty. See requiretty above.
```

```
#
Defaults   !visiblepw

Defaults    env_reset
Defaults    env_keep = "COLORS DISPLAY HOSTNAME HISTSIZE INPUTRC KDEDIR \
                        LS_COLORS MAIL PS1 PS2 QTDIR USERNAME \
                        LANG LC_ADDRESS LC_CTYPE LC_COLLATE LC_IDENTIFICATION \
                        LC_MEASUREMENT LC_MESSAGES LC_MONETARY LC_NAME LC_NUMERIC \
                        LC_PAPER LC_TELEPHONE LC_TIME LC_ALL LANGUAGE LINGUAS \
                        _XKB_CHARSET XAUTHORITY"

## Next comes the main part: which users can run what software on
## which machines (the sudoers file can be shared between multiple
## systems).
## Syntax:
##
##      user    MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)       ALL

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, DRIVERS

## Allows people in group wheel to run all commands
# %wheel        ALL=(ALL)       ALL

## Same thing without a password
# %wheel        ALL=(ALL)       NOPASSWD: ALL

## Allows members of the users group to mount and unmount the
```

```
## cdrom as root
# %users  ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom

## Allows members of the users group to shutdown this system
# %users  localhost=/sbin/shutdown -h now
```

```
[root@centos6 ~]# cat /etc/sudoers
## Sudoers allows particular users to run various commands as
## the root user, without needing the root password.
##
## Examples are provided at the bottom of the file for collections
## of related commands, which can then be delegated out to particular
## users or groups.
##
## This file must be edited with the 'visudo' command.

## Host Aliases
## Groups of machines. You may prefer to use hostnames (perhaps using
## wildcards for entire domains) or IP addresses instead.
# Host_Alias     FILESERVERS = fs1, fs2
# Host_Alias     MAILSERVERS = smtp, smtp2

## User Aliases
## These aren't often necessary, as you can use regular groups
## (ie, from files, LDAP, NIS, etc) in this file - just use %groupname
## rather than USERALIAS
# User_Alias ADMINS = jsmith, mikem


## Command Aliases
## These are groups of related commands...

## Networking
# Cmnd_Alias NETWORKING = /sbin/route, /sbin/ifconfig, /bin/ping, /sbin/dhclient, /usr/bin/net, /sbin/iptables,
```

```
/usr/bin/rfcomm, /usr/bin/wvdial, /sbin/iwconfig, /sbin/mii-tool

## Installation and management of software
# Cmnd_Alias SOFTWARE = /bin/rpm, /usr/bin/up2date, /usr/bin/yum

## Services
# Cmnd_Alias SERVICES = /sbin/service, /sbin/chkconfig

## Updating the locate database
# Cmnd_Alias LOCATE = /usr/bin/updatedb

## Storage
# Cmnd_Alias STORAGE = /sbin/fdisk, /sbin/sfdisk, /sbin/parted, /sbin/partprobe, /bin/mount, /bin/umount

## Delegating permissions
# Cmnd_Alias DELEGATING = /usr/sbin/visudo, /bin/chown, /bin/chmod, /bin/chgrp

## Processes
# Cmnd_Alias PROCESSES = /bin/nice, /bin/kill, /usr/bin/kill, /usr/bin/killall

## Drivers
# Cmnd_Alias DRIVERS = /sbin/modprobe

# Defaults specification

#
# Disable "ssh hostname sudo <cmd>", because it will show the password in clear.
#         You have to run "ssh -t hostname sudo <cmd>".
#
Defaults    requiretty

#
# Preserving HOME has security implications since many programs
# use it when searching for configuration files.
```

```
#
Defaults    always_set_home

Defaults    env_reset
Defaults    env_keep =  "COLORS DISPLAY HOSTNAME HISTSIZE INPUTRC KDEDIR LS_COLORS"
Defaults    env_keep += "MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE"
Defaults    env_keep += "LC_COLLATE LC_IDENTIFICATION LC_MEASUREMENT LC_MESSAGES"
Defaults    env_keep += "LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER LC_TELEPHONE"
Defaults    env_keep += "LC_TIME LC_ALL LANGUAGE LINGUAS _XKB_CHARSET XAUTHORITY"

#
# Adding HOME to env_keep may enable a user to run unrestricted
# commands via sudo.
#
# Defaults   env_keep += "HOME"

Defaults    secure_path = /sbin:/bin:/usr/sbin:/usr/bin

## Next comes the main part: which users can run what software on
## which machines (the sudoers file can be shared between multiple
## systems).
## Syntax:
##
##  user    MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)     ALL

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, DRIVERS
```

```
## Allows people in group wheel to run all commands
# %wheel    ALL=(ALL)    ALL

## Same thing without a password
# %wheel    ALL=(ALL)    NOPASSWD: ALL

## Allows members of the users group to mount and unmount the
## cdrom as root
# %users  ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom

## Allows members of the users group to shutdown this system
# %users  localhost=/sbin/shutdown -h now
```

```
[root@centos7 ~]# cat /etc/sudoers
## Sudoers allows particular users to run various commands as
## the root user, without needing the root password.
##
## Examples are provided at the bottom of the file for collections
## of related commands, which can then be delegated out to particular
## users or groups.
##
## This file must be edited with the 'visudo' command.

## Host Aliases
## Groups of machines. You may prefer to use hostnames (perhaps using
## wildcards for entire domains) or IP addresses instead.
# Host_Alias     FILESERVERS = fs1, fs2
# Host_Alias     MAILSERVERS = smtp, smtp2

## User Aliases
## These aren't often necessary, as you can use regular groups
## (ie, from files, LDAP, NIS, etc) in this file - just use %groupname
## rather than USERALIAS
# User_Alias ADMINS = jsmith, mikem
```

```
## Command Aliases
## These are groups of related commands...

## Networking
# Cmnd_Alias NETWORKING = /sbin/route, /sbin/ifconfig, /bin/ping, /sbin/dhclient, /usr/bin/net, /sbin/iptables,
/usr/bin/rfcomm, /usr/bin/wvdial, /sbin/iwconfig, /sbin/mii-tool

## Installation and management of software
# Cmnd_Alias SOFTWARE = /bin/rpm, /usr/bin/up2date, /usr/bin/yum

## Services
# Cmnd_Alias SERVICES = /sbin/service, /sbin/chkconfig

## Updating the locate database
# Cmnd_Alias LOCATE = /usr/bin/updatedb

## Storage
# Cmnd_Alias STORAGE = /sbin/fdisk, /sbin/sfdisk, /sbin/parted, /sbin/partprobe, /bin/mount, /bin/umount

## Delegating permissions
# Cmnd_Alias DELEGATING = /usr/sbin/visudo, /bin/chown, /bin/chmod, /bin/chgrp

## Processes
# Cmnd_Alias PROCESSES = /bin/nice, /bin/kill, /usr/bin/kill, /usr/bin/killall

## Drivers
# Cmnd_Alias DRIVERS = /sbin/modprobe

# Defaults specification

#
# Disable "ssh hostname sudo <cmd>", because it will show the password in clear.
#         You have to run "ssh -t hostname sudo <cmd>".
#
```

```
Defaults    requiretty

#
# Refuse to run if unable to disable echo on the tty. This setting should also be
# changed in order to be able to use sudo without a tty. See requiretty above.
#
Defaults   !visiblepw

#
# Preserving HOME has security implications since many programs
# use it when searching for configuration files. Note that HOME
# is already set when the the env_reset option is enabled, so
# this option is only effective for configurations where either
# env_reset is disabled or HOME is present in the env_keep list.
#
Defaults    always_set_home

Defaults    env_reset
Defaults    env_keep =  "COLORS DISPLAY HOSTNAME HISTSIZE INPUTRC KDEDIR LS_COLORS"
Defaults    env_keep += "MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE"
Defaults    env_keep += "LC_COLLATE LC_IDENTIFICATION LC_MEASUREMENT LC_MESSAGES"
Defaults    env_keep += "LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER LC_TELEPHONE"
Defaults    env_keep += "LC_TIME LC_ALL LANGUAGE LINGUAS _XKB_CHARSET XAUTHORITY"

#
# Adding HOME to env_keep may enable a user to run unrestricted
# commands via sudo.
#
# Defaults   env_keep += "HOME"

Defaults    secure_path = /sbin:/bin:/usr/sbin:/usr/bin

## Next comes the main part: which users can run what software on
## which machines (the sudoers file can be shared between multiple
```

```
## systems).
## Syntax:
##
##  user    MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)     ALL

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, DRIVERS

## Allows people in group wheel to run all commands
%wheel  ALL=(ALL)    ALL

## Same thing without a password
# %wheel    ALL=(ALL)    NOPASSWD: ALL

## Allows members of the users group to mount and unmount the
## cdrom as root
# %users  ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom

## Allows members of the users group to shutdown this system
# %users  localhost=/sbin/shutdown -h now

## Read drop-in files from /etc/sudoers.d (the # here does not mean a comment)
#includedir /etc/sudoers.d
```

Important : Notez la présence de la ligne en commentaire **# %wheel ALL=(ALL) ALL**. Cette ligne possède le format **Qui Où = (En tant que qui) Quoi**. La ligne implique donc que les membres du groupe **wheel** peuvent exécuter à partir de n'importe quel hôte et en

tant que n'importe quel utilisateur, toutes les commandes du système. Dans ce fichier donc, un groupe est référencé par un **%**. Un nom sans ce caractère est forcément un utilisateur. Pour éditer le fichier **/etc/sudoers**, il est **nécessaire** d'utiliser la commande **visudo**.

Services réseaux

Quand un client émet une demande de connexion vers une application réseau sur un serveur, il utilise un socket attaché à un port local **supérieur à 1023**, alloué d'une manière dynamique. La requête contient le port de destination sur le serveur. Certaines applications serveurs se gèrent toutes seules, ce qui est la cas par exemple d'**httpd**. Par contre d'autres sont gérées par le service **xinetd**.

xinetd

Sous RHEL/CentOS 6 xinetd n'est pas installé par défaut. Installez-le grâce à yum :

```
[root@centos6 ~]# yum install xinetd
Loaded plugins: fastestmirror, refresh-packagekit
Loading mirror speeds from cached hostfile
 * base: fr2.rpmfind.net
 * extras: fr2.rpmfind.net
 * updates: fr2.rpmfind.net
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package xinetd.i686 2:2.3.14-33.el6 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved
```

```
================================================================================
 Package          Arch           Version                    Repository      Size
================================================================================
Installing:
 xinetd           i686           2:2.3.14-33.el6            base           121 k

Transaction Summary
================================================================================
Install       1 Package(s)
Upgrade       0 Package(s)

Total download size: 121 k
Installed size: 258 k
Is this ok [y/N]: y
Downloading Packages:
xinetd-2.3.14-33.el6.i686.rpm                              | 121 kB     00:00
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing     : 2:xinetd-2.3.14-33.el6.i686                            1/1

Installed:
  xinetd.i686 2:2.3.14-33.el6

Complete!
```

Le programme xinetd est configuré via le fichier **/etc/xinetd.conf** :

```
[root@centos6 ~]# cat /etc/xinetd.conf
#
# This is the master xinetd configuration file. Settings in the
# default section will be inherited by all service configurations
# unless explicitly overridden in the service configuration. See
```

```
# xinetd.conf in the man pages for a more detailed explanation of
# these attributes.

defaults
{
# The next two items are intended to be a quick access place to
# temporarily enable or disable services.
#
#   enabled        =
#   disabled    =

# Define general logging characteristics.
    log_type    = SYSLOG daemon info
    log_on_failure    = HOST
    log_on_success    = PID HOST DURATION EXIT

# Define access restriction defaults
#
#   no_access    =
#   only_from    =
#   max_load    = 0
    cps        = 50 10
    instances    = 50
    per_source    = 10

# Address and networking defaults
#
#   bind        =
#   mdns        = yes
    v6only        = no

# setup environmental attributes
#
#   passenv        =
```

```
    groups        = yes
    umask        = 002

# Generally, banners are not used. This sets up their global defaults
#
#   banner        =
#   banner_fail    =
#   banner_success    =
}

includedir /etc/xinetd.d
```

Les valeurs des directives dans le fichier **/etc/xinetd.conf** sont héritées par toutes les configurations des services sauf dans le cas où une variable est explicitement fixée dans un des fichiers de définitions des services se trouvant dans **/etc/xinetd.d**.

Les variables les plus importantes dans **/etc/xinetd.conf** :

Directive	Déscription
instances	Le nombre de demandes d'accès simultanés
log_type	Indique à xinetd d'adresser les traces à SYSLOG avec l'étiquette de sous-système applicatif **daemon** et la priorité de **info**
log_on_succes	Indique que SYSLOG doit journaliser le PID, HOST, DURATION et EXIT en cas de succès
log_on_failure	Indique que SYSLOG doit journaliser le HOST en cas d'échec
cps	Indique 50 connexions par seconde avec un temps d'indisponibilité de 10 secondes si le seuil est atteint

Les options concernant les journaux sont :

HOST	Journalisation de l'adresse IP de l'hôte distant
PID	Journalisation du PID du processus qui reçoit la demande d'accès
DURATION	Journalisation des durées d'utilisation
EXIT	Journalisation de l'état ou di signal de fin de service

Il est aussi possible de trouver les options suivantes pour les journaux :

ATTEMPT	Journalisation des connexions en échec

USERID	Journalisation des données concernant l'utilisateur selon la RFC 1413

Examinons maintenant le répertoire **/etc/xinetd.d** :

```
[root@centos6 ~]# ls -l /etc/xinetd.d
total 52
-rw-------. 1 root root 1157  7 déc.  22:07 chargen-dgram
-rw-------. 1 root root 1159  7 déc.  22:07 chargen-stream
-rw-r--r--. 1 root root  523 25 juin   2011 cvs
-rw-------. 1 root root 1157  7 déc.  22:07 daytime-dgram
-rw-------. 1 root root 1159  7 déc.  22:07 daytime-stream
-rw-------. 1 root root 1157  7 déc.  22:07 discard-dgram
-rw-------. 1 root root 1159  7 déc.  22:07 discard-stream
-rw-------. 1 root root 1148  7 déc.  22:07 echo-dgram
-rw-------. 1 root root 1150  7 déc.  22:07 echo-stream
-rw-r--r--. 1 root root  332 20 mai    2009 rsync
-rw-------. 1 root root 1212  7 déc.  22:07 tcpmux-server
-rw-------. 1 root root 1149  7 déc.  22:07 time-dgram
-rw-------. 1 root root 1150  7 déc.  22:07 time-stream
```

A l'examen de ce répertoire vous noterez que celui-ci contient des fichiers nominatifs par application-serveur, par exemple pour le serveur cvs :

```
[root@centos6 ~]# cat /etc/xinetd.d/cvs
# default: off
# description: The CVS service can record the history of your source \
#              files. CVS stores all the versions of a file in a single \
#              file in a clever way that only stores the differences \
#              between versions.
service cvspserver
{
    disable           = yes
    port            = 2401
    socket_type       = stream
    protocol        = tcp
```

```
    wait            = no
    user            = root
    passenv            = PATH
    server           = /usr/bin/cvs
    env            = HOME=/var/cvs
    server_args        = -f --allow-root=/var/cvs pserver
#   bind            = 127.0.0.1
}
```

Les directives principales de ce fichier sont :

Paramètre	Déscription
disable	**no** : Le service est actif. **yes** : Le service est désactivé
port	Le numéro de port ou, à défaut, le numéro indiqué pour le service dans le fichier /etc/services
socket_type	Nature du socket, soit **stream** pour TCP soit **dgram** pour UDP
protocol	Protocole utilisé soit **TCP** soit **UDP**
wait	**no** : indique si xinetd active un serveur par client. **yes** : indique que xinetd active un seul serveur pour tous les client
user	Indique le compte sous lequel le serveur est exécuté
server	Indique le chemin d'accès de l'application serveur
env	Définit un environnement système
server_args	Donne les arguments transmis à l'application serveur

Cependant il est aussi possible d'utiliser les directives suivantes :

Paramètre	Déscription	Exemple
nice	Fixe le niveau de nice entre -19 et +20	10
max_load	Fixe la charge CPU maximum admise. Au delà aucune connexion supplémenatire en sera acceptée	2.5
bind	Limite le service à l'interface dont l'adresse IP est indiquée	192.168.1.1
only_from	Limite le service aux seuls clients indiqués par la plage donnée	192.168.1.0/24 fenestros.loc
no_access	Interdit le service aux clients indiqués par la plage donnée	192.168.2.0/24, i2tch.loc
access_time	Limite l'accès au service à une plage horaire	09:00-19:00
redirect	Redirige les requêtes sur un port donné à une autre adresse IP	192.168.1.10 23

Afin d'activer une application serveur, il suffit de modifier le paramètre **disable** dans le fichier concerné et de relancer le service xinetd.

TCP Wrapper

TCP Wrapper contrôle l'accès à des services réseaux grâce à des **ACL**.

Quand une requête arrive pour un serveur, xinetd active le wrapper **tcpd** au lieu d'activer le serveur directement.

tcpd met à jour un journal et vérifie si le client a le droit d'utiliser le service concerné. Les ACL se trouvent dans deux fichiers:

- **/etc/hosts.allow**
- **/etc/hosts.deny**

Il faut noter que si ces fichiers n'existent pas ou sont vides, il n'y a pas de contrôle d'accès.

Le format d'une ligne dans un de ces deux fichiers est:

```
démon : liste_de_clients
```

Par exemple dans le cas d'un serveur **démon**, on verrait une ligne dans le fichier **/etc/hosts.allow** similaire à:

```
démon : LOCAL, .fenestros.loc
```

ce qui implique que les machines dont le nom ne comporte pas de point ainsi que les machines du domaine **fenestros.loc** sont autorisées à utiliser le service.

Le mot clef **ALL** peut être utilisé pour indiquer tout. Par exemple, **ALL:ALL** dans le fichier **/etc/host.deny** bloque effectivement toute tentative de connexion à un service xinetd sauf pour les ACL inclus dans le fichier /etc/host.allow.

SSH

Introduction

La commande **ssh** est le successeur et la remplaçante de la commande **rlogin**. Il permet d'établir des connexions sécurisées avec une machine distante. SSH comporte cinq acteurs :

- Le **serveur SSH**
  - le démon sshd, qui s'occupe des authentifications et autorisations des clients,
- Le **client SSH**
  - ssh ou scp, qui assure la connexion et le dialogue avec le serveur,
- La **session** qui représente la connexion courante et qui commence juste après l'authentification réussie,
- Les **clefs**
  - **Couple de clef utilisateur asymétriques** et persistantes qui assurent l'identité d'un utilisateur et qui sont stockés sur disque dur,
  - **Clef hôte asymétrique et persistante** garantissant l'identité du serveur er qui est conservé sur disque dur
  - **Clef serveur asymétrique et temporaire** utilisée par le protocole SSH1 qui sert au chiffrement de la clé de session,
  - **Clef de session symétrique qui est générée aléatoirement** et qui permet le chiiffrement de la communication entre le client et le serveur. Elle est détruite en fin de session. SSH-1 utilise une seule clef tandis que SSH-2 utilise une clef par direction de la communication,
- La **base de données des hôtes connus** qui stocke les clés des connexions précédentes.

SSH fonctionne de la manière suivante pour la la mise en place d'un canal sécurisé:

- Le client contacte le serveur sur son port 22,
- Les client et le serveur échangent leur version de SSH. En cas de non-compatibilité de versions, l'un des deux met fin au processus,
- Le serveur SSH s'identifie auprès du client en lui fournissant :
  - Sa clé hôte,
  - Sa clé serveur,
  - Une séquence aléatoire de huit octets à inclure dans les futures réponses du client,
  - Une liste de méthodes de chiffrage, compression et authentification,
- Le client et le serveur produisent un identifiant identique, un haché MD5 long de 128 bits contenant la clé hôte, la clé serveur et la séquence aléatoire,
- Le client génère sa clé de session symétrique et la chiffre deux fois de suite, une fois avec la clé hôte du serveur et la deuxième fois avec la clé serveur. Le client envoie cette clé au serveur accompagnée de la séquence aléatoire et un choix d'algorithmes supportés,
- Le serveur déchiffre la clé de session,
- Le client et le serveur mettent en place le canal sécurisé.

SSH-1

SSH-1 utilise une paire de clefs de type RSA1. Il assure l'intégrité des données par une **Contrôle de Redondance Cyclique** (CRC) et est un bloc dit **monolithique**.

Afin de s'identifier, le client essaie chacune des six méthodes suivantes :

- **Kerberos**,
- **Rhosts**,
- **RhostsRSA**,
- Par **clef asymétrique**,
- **TIS**,
- Par **mot de passe**.

SSH-2

SSH-2 utilise **DSA** ou **RSA**. Il assure l'intégrité des données par l'algorithme **HMAC**. SSH-2 est organisé en trois **couches** :

- **SSH-TRANS** – Transport Layer Protocol,
- **SSH-AUTH** – Authentification Protocol,
- **SSH-CONN** – Connection Protocol.

SSH-2 diffère de SSH-1 essentiellement dans la phase authentification.

Trois méthodes d'authentification :

- Par **clef asymétrique**,
  - Identique à SSH-1 sauf avec l'algorithme DSA,
- **RhostsRSA** appelé **hostbased authentification**,
- Par **mot de passe**.

Options de la commande

Les options de cette commande sont :

```
[root@centos6 ~]# ssh --help
usage: ssh [-1246AaCfgKkMNnqsTtVvXxYy] [-b bind_address] [-c cipher_spec]
           [-D [bind_address:]port] [-e escape_char] [-F configfile]
           [-i identity_file] [-L [bind_address:]port:host:hostport]
           [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]
           [-R [bind_address:]port:host:hostport] [-S ctl_path]
           [-w local_tun[:remote_tun]] [user@]hostname [command]
```

RhostsRSA et Rlogin

Rlogin est un mécanism d'authentification basé sur le nom de la machine est le nom d'un utilisateur associé à cette machine. La machine est ensuite classifiée comme “machine hôte de confince”. Aucun mot de passe n'est necéssaire pour se connecter au serveur distant.

Selon Wikipédia **https://fr.wikipedia.org/wiki/Rlogin**

“Normalement, les utilisateurs doivent fournir un code d'accès et un mot de passe valable sur la machine locale avant de pouvoir se connecter. Le concept de machine hôte de confiance repose sur le fait que les utilisateurs qui appellent à partir d'une machine hôte de confiance ne sont pas obligés de fournir un mot de passe. Quoique pratique, ce concept est dangereux vis-à-vis de la sécurité du système. De plus la confiance est transitive en ce sens que si la machine A fait confiance à la machine B et que B fait de même pour la machine C, alors un utilisateur sur C peut effectuer un rlogin sans mot de passe vers B, puis un autre rlogin, toujours sans mot de passe, vers A et ce, même si A ne fait pas explicitement confiance à la machine hôte C.”

RhostsRSA pour SSH-1, appelé hostbased authentification pour SSH-2, est un mécanism plus sur car l'authentification utilise la clef hôte du client. Pour une déscription complète, consultez la section “3.4.2.3. Trusted-host authentication (Rhosts and RhostsRSA)” à cette adresse - **https://docstore.mik.ua/orelly/networking_2ndEd/ssh/ch03_04.htm#ch03-19795.html**.

L'authentification par mot de passe

L'utilisateur fournit un mot de passe au client ssh. Le client ssh le transmet de façon sécurisée au serveur ssh puis le serveur vérifie le mot de passe et l'accepte ou non.

Avantage:

- Aucune configuration de clef asymétrique n'est nécessaire.

Inconvénients:

- L'utilisateur doit fournir à chaque connexion un identifiant et un mot de passe,
- Moins sécurisé qu'un système par clef asymétrique.

L'authentification par clef asymétrique

- Le **client** envoie au serveur une requête d'authentification par clé asymétrique qui contient le module de la clé à utiliser,
- Le **serveur** recherche une correspondance pour ce module dans le fichier des clés autorisés **~/.ssh/authorized_keys**,
  - Dans le cas où une correspondance n'est pas trouvée, le serveur met fin à la communication,
  - Dans le cas contraire le serveur génère une chaîne aléatoire de 256 bits appelée un **challenge** et la chiffre avec la **clé publique du client**,
- Le **client** reçoit le challenge et le décrypte avec la partie privée de sa clé. Il combine le challenge avec l'identifiant de session et chiffre le résultat. Ensuite il envoie le résultat chiffré au serveur.
- Le **serveur** génère le même haché et le compare avec celui reçu du client. Si les deux hachés sont identiques, l'authentification est réussie.

Installation

Pour installer/mettre à jour le serveur **sshd**, utilisez **yum** :

```
[root@centos7 ~]# yum install openssh-server
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
```

```
 * base: centos.mirror.fr.planethoster.net
 * extras: ftp.ciril.fr
 * updates: centos.mirrors.ovh.net
Package openssh-server-6.6.1p1-25.el7_2.x86_64 already installed and latest version
Nothing to do
```

Important - Pour les stations de travail, installez le client : **openssh-clients**.

Options de la commande

Les options de la commande sont :

```
SYNOPSIS
     sshd [-46DdeiqTt] [-b bits] [-C connection_spec] [-f config_file] [-g login_grace_time] [-h host_key_file]
[-k key_gen_time] [-o option] [-p port]  [-u len]
```

Configuration

Important - La configuration doit s'effectuer dans la fenêtre de la VM sous VirtualBox. Les connexions en ssh doivent de faire à partir d'un terminal ou à partir de l'application putty.

Serveur

La configuration du serveur s'effectue dans le fichier **/etc/ssh/sshd_config** :

```
[root@centos7 ~]# cat /etc/ssh/sshd_config
#   $OpenBSD: sshd_config,v 1.93 2014/01/10 05:59:19 djm Exp $

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/bin:/usr/bin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

# If you want to change the port on a SELinux system, you have to tell
# SELinux about this change.
# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER
#
#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

# The default requires explicit activation of protocol 1
#Protocol 2

# HostKey for protocol version 1
#HostKey /etc/ssh/ssh_host_key
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_dsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key

# Lifetime and size of ephemeral version 1 server key
```

```
#KeyRegenerationInterval 1h
#ServerKeyBits 1024

# Ciphers and keying
#RekeyLimit default none

# Logging
# obsoletes QuietMode and FascistLogging
#SyslogFacility AUTH
SyslogFacility AUTHPRIV
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#RSAAuthentication yes
#PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile  .ssh/authorized_keys

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#RhostsRSAAuthentication no
```

```
# similar for protocol version 2
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# RhostsRSAAuthentication and HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no
PasswordAuthentication yes

# Change to no to disable s/key passwords
#ChallengeResponseAuthentication yes
ChallengeResponseAuthentication no

# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#KerberosGetAFSToken no
#KerberosUseKuserok yes

# GSSAPI options
GSSAPIAuthentication yes
GSSAPICleanupCredentials no
#GSSAPIStrictAcceptorCheck yes
#GSSAPIKeyExchange no
#GSSAPIEnablek5users no

# Set this to 'yes' to enable PAM authentication, account processing,
# and session processing. If this is enabled, PAM authentication will
# be allowed through the ChallengeResponseAuthentication and
```

```
# PasswordAuthentication.  Depending on your PAM configuration,
# PAM authentication via ChallengeResponseAuthentication may bypass
# the setting of "PermitRootLogin without-password".
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# and ChallengeResponseAuthentication to 'no'.
# WARNING: 'UsePAM no' is not supported in Red Hat Enterprise Linux and may cause several
# problems.
UsePAM yes

#AllowAgentForwarding yes
#AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PermitTTY yes
#PrintMotd yes
#PrintLastLog yes
#TCPKeepAlive yes
#UseLogin no
UsePrivilegeSeparation sandbox      # Default for new installations.
#PermitUserEnvironment no
#Compression delayed
#ClientAliveInterval 0
#ClientAliveCountMax 3
#ShowPatchLevel no
#UseDNS yes
#PidFile /var/run/sshd.pid
#MaxStartups 10:30:100
#PermitTunnel no
#ChrootDirectory none
#VersionAddendum none
```

```
# no default banner path
#Banner none

# Accept locale-related environment variables
AcceptEnv LANG LC_CTYPE LC_NUMERIC LC_TIME LC_COLLATE LC_MONETARY LC_MESSAGES
AcceptEnv LC_PAPER LC_NAME LC_ADDRESS LC_TELEPHONE LC_MEASUREMENT
AcceptEnv LC_IDENTIFICATION LC_ALL LANGUAGE
AcceptEnv XMODIFIERS

# override default of no subsystems
Subsystem   sftp    /usr/libexec/openssh/sftp-server

# Example of overriding settings on a per-user basis
#Match User anoncvs
#   X11Forwarding no
#   AllowTcpForwarding no
#   PermitTTY no
#   ForceCommand cvs server
```

Pour ôter les lignes de commentaires dans ce fichier, utilisez la commande suivante :

```
[root@centos7 ~]# cd /tmp ; grep -E -v '^(#|$)'  /etc/ssh/sshd_config > sshd_config
[root@centos7 tmp]# cat sshd_config
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key
SyslogFacility AUTHPRIV
AuthorizedKeysFile  .ssh/authorized_keys
PasswordAuthentication yes
ChallengeResponseAuthentication no
GSSAPIAuthentication yes
GSSAPICleanupCredentials no
UsePAM yes
X11Forwarding yes
```

```
UsePrivilegeSeparation sandbox      # Default for new installations.
AcceptEnv LANG LC_CTYPE LC_NUMERIC LC_TIME LC_COLLATE LC_MONETARY LC_MESSAGES
AcceptEnv LC_PAPER LC_NAME LC_ADDRESS LC_TELEPHONE LC_MEASUREMENT
AcceptEnv LC_IDENTIFICATION LC_ALL LANGUAGE
AcceptEnv XMODIFIERS
Subsystem   sftp    /usr/libexec/openssh/sftp-server
```

Pour sécuriser le serveur ssh, ajoutez ou modifiez les directives suivantes :

```
AllowGroups adm
Banner /etc/issue.net
HostbasedAuthentication no
IgnoreRhosts yes
LoginGraceTime 60
LogLevel INFO
PermitEmptyPasswords no
PermitRootLogin no
PrintLastLog yes
Protocol 2
StrictModes yes
X11Forwarding no
```

Votre fichier ressemblera à celui-ci :

```
AllowGroups adm
Banner /etc/issue.net
HostbasedAuthentication no
IgnoreRhosts yes
LoginGraceTime 60
LogLevel INFO
PermitEmptyPasswords no
PermitRootLogin no
PrintLastLog yes
Protocol 2
```

```
StrictModes yes
X11Forwarding no
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key
SyslogFacility AUTHPRIV
AuthorizedKeysFile  .ssh/authorized_keys
PasswordAuthentication yes
ChallengeResponseAuthentication no
GSSAPIAuthentication yes
GSSAPICleanupCredentials no
UsePAM yes
UsePrivilegeSeparation sandbox      # Default for new installations.
AcceptEnv LANG LC_CTYPE LC_NUMERIC LC_TIME LC_COLLATE LC_MONETARY LC_MESSAGES
AcceptEnv LC_PAPER LC_NAME LC_ADDRESS LC_TELEPHONE LC_MEASUREMENT
AcceptEnv LC_IDENTIFICATION LC_ALL LANGUAGE
AcceptEnv XMODIFIERS
Subsystem   sftp    /usr/libexec/openssh/sftp-server
```

A Faire - Renommez le fichier **/etc/ssh/sshd_config** en **/etc/ssh/sshd_config.old** puis copiez le fichier **/tmp/sshd_config** vers **/etc/ssh/**. Redémarrez ensuite le service sshd. N'oubliez pas de mettre l'utilisateur **trainee** dans le groupe **adm** !

Pour générer les clefs sur le serveur saisissez la commande suivante en tant que **root**:

Lors de la génération des clefs, la passphrase doit être **vide**.

```
[root@centos7 ~]# ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/root/.ssh/id_dsa): /etc/ssh/ssh_host_dsa_key
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
```

```
Your identification has been saved in /etc/ssh/ssh_host_dsa_key.
Your public key has been saved in /etc/ssh/ssh_host_dsa_key.pub.
The key fingerprint is:
d5:54:d3:30:1c:f5:da:f8:21:15:1f:c8:6c:3b:b1:ff root@centos7.fenestros.loc
The key's randomart image is:
+--[ DSA 1024]----+
|            +oBB.|
|           o *.o*|
|          . o +.o|
|         .   +.+ |
|        S    .=..|
|              .o.|
|                o|
|                E|
|                 |
+-----------------+
```

Le chemin à indiquer pour le fichier est **/etc/ssh/ssh_host_dsa_key**. De la même façon, il est possible de générer les clefs au format **RSA**, **ECDSA** et **ED25519**.

Les clefs publiques générées possèdent l'extension **.pub**. Les clefs privées n'ont pas d'extension :

```
[root@centos7 ~]# ls /etc/ssh
moduli      sshd_config       ssh_host_dsa_key.pub  ssh_host_ecdsa_key.pub  ssh_host_ed25519_key.pub
ssh_host_rsa_key.pub
ssh_config  ssh_host_dsa_key  ssh_host_ecdsa_key    ssh_host_ed25519_key    ssh_host_rsa_key
```

Re-démarrez ensuite le service sshd :

```
[root@centos7 ~]# systemctl restart sshd.service
```

Saisissez maintenant les commandes suivantes en tant que **trainee** :

Lors de la génération des clefs, la passphrase doit être **vide**.

```
[trainee@centos7 ~]$ ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/home/trainee/.ssh/id_dsa):
Created directory '/home/trainee/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/trainee/.ssh/id_dsa.
Your public key has been saved in /home/trainee/.ssh/id_dsa.pub.
The key fingerprint is:
97:92:85:d1:ae:97:f7:64:d2:54:45:89:eb:57:b1:66 trainee@centos7.fenestros.loc
The key's randomart image is:
+--[ DSA 1024]----+
|        ..    ..=|
|         o.  . o.|
|        ...   ..o|
|         o.. ..E.|
|        S.o..oo .|
|        .oo o.+. |
|         . . =.  |
|              .  |
|                 |
+-----------------+
[trainee@centos7 ~]$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/trainee/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
```

```
Your identification has been saved in /home/trainee/.ssh/id_rsa.
Your public key has been saved in /home/trainee/.ssh/id_rsa.pub.
The key fingerprint is:
80:4c:5a:bf:d0:2f:d1:a1:34:7c:09:a1:9c:0d:ed:2d trainee@centos7.fenestros.loc
The key's randomart image is:
+--[ RSA 2048]----+
|    +o=o..       |
|   * Xo+o.       |
|  . B.Bo.        |
|     .E=.        |
|      o.S        |
|       .         |
|                 |
|                 |
|                 |
+-----------------+
[trainee@centos7 ~]$ ssh-keygen -t ecdsa
Generating public/private ecdsa key pair.
Enter file in which to save the key (/home/trainee/.ssh/id_ecdsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/trainee/.ssh/id_ecdsa.
Your public key has been saved in /home/trainee/.ssh/id_ecdsa.pub.
The key fingerprint is:
41:5d:64:cf:d6:4a:ce:8e:a9:a8:4a:62:04:57:09:fc trainee@centos7.fenestros.loc
The key's randomart image is:
+--[ECDSA  256]---+
| .....  .. o+    |
|  ...  .  .. o . |
|. ..    .     = .|
| o  E    .   = . |
|  .     S     +  |
| .           +   |
|  o .       o .  |
```

```
| . o     . .     |
|    ..... .      |
+-----------------+
[trainee@centos7 ~]$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/trainee/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/trainee/.ssh/id_ed25519.
Your public key has been saved in /home/trainee/.ssh/id_ed25519.pub.
The key fingerprint is:
66:3a:83:d1:6d:79:46:48:88:7c:d9:65:59:bb:e6:d0 trainee@centos7.fenestros.loc
The key's randomart image is:
+--[ED25519  256--+
|   . . +..oo.    |
|    o +..o.  .   |
|     .  . . .    |
|     . . o . .   |
|    . . S + E    |
|     o = o +     |
|    . +     .    |
|       o         |
|                 |
+-----------------+
```

Les clés générées seront placées dans le répertoire **~/.ssh/**.

Utilisation

La commande ssh prend la forme suivante:

```
ssh -l nom_de_compte numero_ip (nom_de_machine)
```

En saisissant cette commande sur votre propre machine, vous obtiendrez un résultat similaire à celle-ci :

```
[trainee@centos7 ~]$ su -
Mot de passe :
Dernière connexion : lundi  9 mai 2016 à 22:47:48 CEST sur pts/0

[root@centos7 ~]# ssh -l trainee localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is 19:cd:05:58:af:2c:10:82:52:ba:e3:31:df:bd:72:54.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
trainee@localhost's password: trainee
Last login: Mon May  9 23:25:15 2016 from localhost.localdomain
```

Tunnels SSH

Le protocole SSH peut être utilisé pour sécuriser les protocoles tels telnet, pop3 etc.. En effet, on peut créer un *tunnel* SSH dans lequel passe les communications du protocole non-sécurisé.

La commande pour créer un tunnel ssh prend la forme suivante :

```
ssh -N -f compte@hôte -Lport-local:localhost:port_distant
```

Dans votre cas, vous allez créer un tunnel dans votre propre vm entre le port 15023 et le port 23 :

```
[root@centos7 ~]# ssh -N -f trainee@localhost -L15023:localhost:23
trainee@localhost's password:
```

Installez maintenant le client et le serveur telnet :

```
[root@centos7 ~]# yum install telnet telnet-server
```

Telnet n'est ni démarré ni activé. Il convient donc de le démarrer et de l'activer :

```
[root@centos7 ~]# systemctl status telnet.socket
● telnet.socket - Telnet Server Activation Socket
   Loaded: loaded (/usr/lib/systemd/system/telnet.socket; disabled; vendor preset: disabled)
   Active: inactive (dead)
     Docs: man:telnetd(8)
   Listen: [::]:23 (Stream)
 Accepted: 0; Connected: 0

[root@centos7 ~]# systemctl start telnet.socket

[root@centos7 ~]# systemctl status telnet.socket
● telnet.socket - Telnet Server Activation Socket
   Loaded: loaded (/usr/lib/systemd/system/telnet.socket; disabled; vendor preset: disabled)
   Active: active (listening) since Mon 2016-05-09 23:40:13 CEST; 3s ago
     Docs: man:telnetd(8)
   Listen: [::]:23 (Stream)
 Accepted: 0; Connected: 0

May 09 23:40:13 centos7.fenestros.loc systemd[1]: Listening on Telnet Server Activation Socket.
May 09 23:40:13 centos7.fenestros.loc systemd[1]: Starting Telnet Server Activation Socket.

[root@centos7 ~]# systemctl enable telnet.socket
Created symlink from /etc/systemd/system/sockets.target.wants/telnet.socket to
/usr/lib/systemd/system/telnet.socket.
[root@centos7 ~]# systemctl status telnet.socket
● telnet.socket - Telnet Server Activation Socket
   Loaded: loaded (/usr/lib/systemd/system/telnet.socket; enabled; vendor preset: disabled)
   Active: active (listening) since Mon 2016-05-09 23:40:13 CEST; 36s ago
     Docs: man:telnetd(8)
   Listen: [::]:23 (Stream)
```

```
 Accepted: 0; Connected: 0

May 09 23:40:13 centos7.fenestros.loc systemd[1]: Listening on Telnet Server Activation Socket.
May 09 23:40:13 centos7.fenestros.loc systemd[1]: Starting Telnet Server Activation Socket.
```

Connectez-vous ensuite via telnet sur le port 15023, vous constaterez que votre connexion n'aboutit pas :

```
[root@centos7 ~]# telnet localhost 15023
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Kernel 3.10.0-327.13.1.el7.x86_64 on an x86_64
centos7 login: trainee
Password:
Last login: Mon May  9 23:26:32 from localhost.localdomain
[trainee@centos7 ~]$
```

Notez bien que votre communication telnet passe par le tunnel SSH.

SCP

Introduction

La commande **scp** est le successeur et la remplaçante de la commande **rcp** de la famille des commandes **remote**. Il permet de faire des transferts sécurisés à partir d'une machine distante :

```
$ scp compte@numero_ip(nom_de_machine):/chemin_distant/fichier_distant /chemin_local/fichier_local
```

ou vers une machine distante :

```
$ scp /chemin_local/fichier_local compte@numero_ip(nom_de_machine):/chemin_distant/fichier_distant
```

Utilisation

Nous allons maintenant utiliser **scp** pour chercher un fichier sur le «serveur» :

Créez le fichier **/home/trainee/scp_test** :

```
[trainee@centos7 ~]$ pwd
/home/trainee
[trainee@centos7 ~]$ touch scp_test
```

Récupérez le fichier **scp_test** en utilisant scp :

```
[trainee@centos7 ~]$ touch /home/trainee/scp_test
[trainee@centos7 ~]$ scp trainee@127.0.0.1:/home/trainee/scp_test /tmp/scp_test
The authenticity of host '127.0.0.1 (127.0.0.1)' can't be established.
ECDSA key fingerprint is 19:cd:05:58:af:2c:10:82:52:ba:e3:31:df:bd:72:54.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '127.0.0.1' (ECDSA) to the list of known hosts.
trainee@127.0.0.1's password: trainee
scp_test                                                                                    100%    0
0.0KB/s   00:00
[trainee@centos7 ~]$ ls /tmp/scp_test
/tmp/scp_test
```

Mise en place des clefs

Il convient maintenant de se connecter sur le «serveur» en utilisant ssh et vérifiez la présence du répertoire ~/.ssh :

En saisissant cette commande, vous obtiendrez une fenêtre similaire à celle-ci :

```
[trainee@centos7 ~]$ ssh -l trainee 127.0.0.1
trainee@127.0.0.1's password:
Last login: Mon May  9 23:42:46 2016 from localhost.localdomain
[trainee@centos7 ~]$ ls -la | grep .ssh
drwx------.  2 trainee trainee 4096 May  9 23:25 .ssh
[trainee@centos7 ~]$ exit
logout
Connection to 127.0.0.1 closed.
```

Si le dossier distant .ssh n'existe pas dans le répertoire personnel de l'utilisateur connecté, il faut le créer avec des permissions de 700. Dans votre cas, puisque votre machine joue le rôle de serveur **et** du client, le dossier /home/trainee/.ssh existe **déjà**.

Ensuite, il convient de transférer le fichier local **.ssh/id_ecdsa.pub** du «client» vers le «serveur» en le renommant en **authorized_keys** :

```
[trainee@centos7 ~]$ scp .ssh/id_ecdsa.pub trainee@127.0.0.1:/home/trainee/.ssh/authorized_keys
trainee@127.0.0.1's password: trainee
id_ecdsa.pub                                                                  100%  227
0.2KB/s   00:00
```

Connectez-vous via telnet et insérer les clefs publiques restantes dans le fichier .ssh/authorized_keys :

```
root@centos7 ~]# ssh -l trainee localhost
trainee@localhost's password: trainee
Last login: Tue May 10 01:39:33 2016 from localhost.localdomain
[trainee@centos7 ~]$ cat .ssh/id_rsa.pub >> .ssh/authorized_keys
[trainee@centos7 ~]$ cat .ssh/id_dsa.pub >> .ssh/authorized_keys
[trainee@centos7 ~]$ cat .ssh/id_ed25519.pub >> .ssh/authorized_keys
[trainee@centos7 ~]$ cat .ssh/authorized_keys
```

```
ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBG5BtOMFLrUbxD//RLELvkkA06CQvXuJqKSjSB2dlUXgaPyEJXuwHO0pxcdbr
g4qqb0f9sE75oMVowXxYgqhWDE= trainee@centos7.fenestros.loc
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQC9KOuEH5+kyihhm99Na8UTA4Gi5AfiOVeJyS3UzH7ta73ewmv7JZqaXzar1NlHcpEMkCUs2yKxHyO/yAfjb
CSdow5vfwJiuJTes+HbpvsJqKp1+OR7tf+OMgjDcajoGi7DYuybIs9QrbWgh57QclbldHQXRO+xbeUTykxcRun7AvR5uWZe4zMooBAmVVEms+l1rn
8CUi+D811jqQGSpU39PxkojTAwgbxlevT/Twy4sfeRR47UHc3AbrHb8SgyKqbx5/S9UxkbkhJjckx0s58fnAwf9nX5rKE7RdCQisRvdLeLHozq3EO
omvc7kzejefBtUDWxBEjnSeAgIP3+OEQl trainee@centos7.fenestros.loc
ssh-dss
AAAAB3NzaC1kc3MAAACBAK9/4siucBnf/NAHBMjZWIx1coA/wYVBjfudVyKArip1fVUuYqf0Ri9vTorG8KJ2zzLRbW5z7V5ZDSn4f6P5Kv7K5xVPn
e9dYQHxImkZIljpFseUW56BwCvcgTNZVLD0tYZzF+B0/Py4waJW+pnTDfZush6DYyAhVnEuxIPI4i+PAAAAFQCeCZyDRo1o41lf19qWGJTG7W+ChQ
AAAIAKtQe9QlkW4CA9kP+q4v3N07WR5TzWsvfZARjGXgrSqTo0BeQgMLwRJHeEOhdsgJ30cNbl6QXlB4G4J6dUoTiN/sY1dFbXzjzsT/MHLedsllV
fXXRQxgvN2nsbsKEUnmqEBWzgw5s6KOkGX33+0Six0E3xvOrYxkMNLP/5VT4aQwAAAIEAmOS94peBeo78yCKzCvSFnEL72dUCFFA6CGFGqgffhK1v
P5H5pG5vQxzBn9NnIXURCACF7ZxtZaxohSoB1M0/s0DfrfNIvXRMGvsJpZ9B2psTMDl9qBffIfIARnwkWKG1gC/lWaovUpDByE1wl09ZCDCnZp/16
ULJY0zvJ566Seg= trainee@centos7.fenestros.loc
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIENas3A3hmXFj1cb+lrn2NAt6g95Pla6qUFQHd1wg2y1 trainee@centos7.fenestros.loc
ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBG5BtOMFLrUbxD//RLELvkkA06CQvXuJqKSjSB2dlUXgaPyEJXuwHO0pxcdbr
g4qqb0f9sE75oMVowXxYgqhWDE= trainee@centos7.fenestros.loc
```

Lors de la connexion suivante au serveur, l'authentification utilise le couple de clefs asymétrique et aucun mot de passe n'est requis :

```
[trainee@centos7 ~]$ ssh -l trainee localhost
Last login: Tue May 10 01:50:39 2016 from localhost.localdomain
[trainee@centos7 ~]$ exit
déconnexion
Connection to localhost closed.
```

Le fichier **authorized_keys** doit avoir les permissions de 600.