

IE-0117 Programacion bajo plataformas abiertas

Laboratorio 2: Usuarios, permisos y credenciales en GNU/Linux

M. Sc. Ricardo Román Brenes - `ricardo.roman@ucr.ac.cr`

I-2018

Tabla de contenidos

1. Enunciado	1
1.1. Usuarios, grupos y permisos	1
1.2. Instalacion de paquetes y programas desde otros repositorios	2
1.3. Cron, crontab, rsync, ssh	2
1.4. Instalacion de programas desde codigo fuente	3
1.5. Capturas de pantalla	3

1. Enunciado

Entregue un archivo comprimido que incluya un directorio llamado informe con los archivos necesarios para generar el PDF del informe. El informe debe contener la documentacion necesaria que demuestre que se llego a una solucion satisfactorio de los problemas resueltos

1.1. Usuarios, grupos y permisos

1. Cambie de usuario a root. `sudo -s`
2. Cree un usuario nuevo llamado labo2, con la contraseña labo2.
3. Cambie de usuario a labo2. Cree un directorio llamado PruebasPermisos en el \$HOME de este usuario.
4. Dentro de este directorio nuevo, cree un archivo llamado README, que su contenido sea una única linea que diga: Realizando pruebas de permisos.

5. Con un solo comando, cambie los permisos del directorio PruebasPermisos y todos los archivos que contiene, de manera tal que solamente el usuario dueño del directorio, así como el grupo al que pertenece, tengan permisos de lectura, escritura y ejecución. Los demás no tendrán permisos de nada.
6. Cambie a su usuario original. Trate de escribir al directorio PruebasPermisos y al archivo README contenido. Documente el resultado.
7. Cambie de usuario a root nuevamente.
8. Cree un grupo nuevo llamado grupolabo2, agregue el usuario labo2 y su propio usuario a este grupo.
9. Cierre la sesión gráfica de Linux y vuelva a iniciarla.
10. Vuelva a iniciar sesión como su usuario, y cambie en terminal el usuario a labo2.
11. Cambie recursivamente el grupo de pertenencia del directorio PruebasPermisos al grupo grupolabo2.
12. Cierre la sesión del usuario labo2 y vuelva a su propio usuario. Trate de escribir al directorio PruebasPermisos y al archivo README. Documente el resultado.
13. Explique con sus propias palabras el procedimiento que se realizó anteriormente. ¿Por qué antes su usuario no podía escribir en el archivo? ¿Por qué ahora sí puede? En principio no era posible modificar con mi usuario el archivo README ni el directorio PruebasPermisos ya que solo el usuario labo2 tenía los permisos para hacerlo. Luego al crear un grupo grupolabo2 e incluir mi usuario y el usuario labo2 en este grupo, y modificar los permisos para que quienes pertenezcan al grupo puedan modificarlo, fue posible hacer modificaciones desde mi usuario.

1.2. Instalación de paquetes y programas desde otros repositorios

1. Siga el tutorial de instalación de Spotify en Linux que se encuentra en el siguiente enlace: <https://www.spotify.com/cr/download/linux/>. Documente en su informe el porqué se realizó cada uno de los pasos.

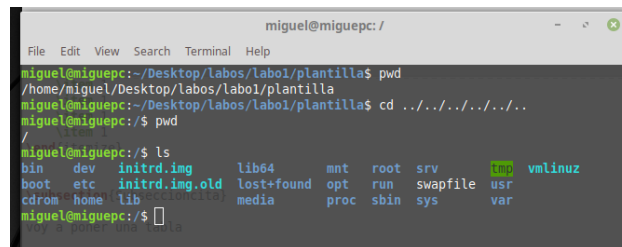
1.3. Cron, crontab, rsync, ssh

1. Realice revisión bibliográfica de qué es cron.
2. Escriba un breve resumen de cuál es el formato del archivo crontab y cómo se utiliza.
3. Investigue sobre el uso del comando rsync, y su uso a través de ssh. Puede utilizar este enlace como guía <https://www.digitalocean.com/community/tutorials/how-to-copy-files-with-rsync-over-ssh>
4. Cree llaves ssh para su usuario de manera que no le pida contraseña a la hora de conectarse la computadora de su compañero.
5. Ejecute un comando de rsync sobre ssh que permita respaldar su \$HOME en algún directorio del \$HOME del usuario de su compañero.
6. Programe su computadora para realizar el ejercicio del punto anterior todos los miércoles a las 3:00am.

1.4. Instalacion de programas desde codigo fuente

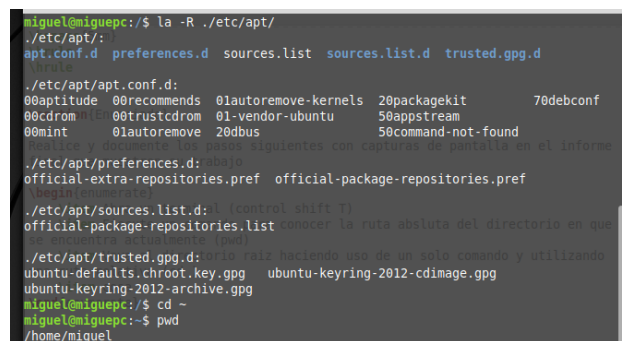
1. Visite el siguiente enlace: <https://ffmpeg.org/download.html>.
2. Descargue el fichero comprimido ffmpeg-3.4.2.tar.bz2.
3. Proceda a instalar el programa a traves del codigo fuente descargado. Nota: recuerde revisar el README o el INSTALL para conocer las dependencias e instalarlas antes de iniciar la configuracion del paquete y la compilacion. Si no lo hace el script de configure o la compilacion podrian fallar.

1.5. Capturas de pantalla



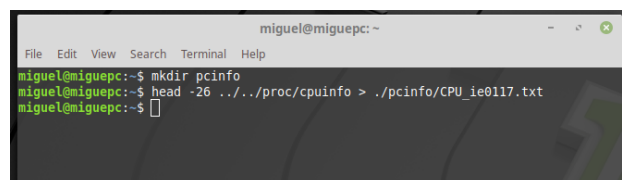
```
miguel@miguepc: /
File Edit View Search Terminal Help
miguel@miguepc:~/Desktop/labos/lab01/plantilla$ pwd
/home/miguel/Desktop/labos/lab01/plantilla
miguel@miguepc:~/Desktop/labos/lab01/plantilla$ cd ../../../../../../..
miguel@miguepc:/$ pwd
/
miguel@miguepc:/$ ls
bin      dev      initrd.img  lib64      mnt      root    srv      tmp      vmlinuz
boot    etc      initrd.img.old  lost+found  opt      run     swapfile  usr
cdrom   home    lib          media      proc     sbin    sys      var
```

Figura 1: comandos de enunciado del 1 al 4



```
miguel@miguepc:/$ la -R ./etc/apt/
./etc/apt/:
apt.conf.d preferences.d sources.list sources.list.d trusted.gpg.d
chroot
./etc/apt/apt.conf.d:
00aptitude 00recommends 01autoremove-kernels 20packagekit 70debconf
00cdrom 00trustcdrom 01-vendor-ubuntu 50appstream
00mint 01autoremove 20dbus 50command-not-found
Realice y documente los pasos siguientes con capturas de pantalla en el informe
./etc/apt/preferences.d: abajo
official-extra-repositories.pref official-package-repositories.pref
(begin(enumerate)
./etc/apt/sources.list.d: l (control shift T)
official-package-repositories.list conocer la ruta absoluta del directorio en que
se encuentra actualmente (pwd)
./etc/apt/trusted.gpg.d: orlo raiz haciendo uso de un solo comando y utilizando
ubuntu-defaults.chroot.key.gpg ubuntu-keyring-2012-cdimage.gpg
ubuntu-keyring-2012-archive.gpg
miguel@miguepc:/$ cd -
miguel@miguepc:/$ pwd
/home/miguel
```

Figura 2: comandos de enunciado del 5 al 6



```
miguel@miguepc: ~
File Edit View Search Terminal Help
miguel@miguepc:/$ mkdir pcinfo
miguel@miguepc:/$ head -26 ../../proc/cpuinfo > ./pcinfo/CPU_ie0117.txt
miguel@miguepc:/$
```

Figura 3: comandos de enunciado del 7 y 8

```
miguel@miguepc:~$ nano pcinfo/CPU_ie0117.txt
miguec@miguepc:~$ cat pcinfo/CPU_ie0117.txt
{
  "family": 6,
  "model": 58,
  "model name": "Intel(R) Core(TM) i3-3217U CPU @ 1.80GHz",
  "stepping": 9,
  "microcode": 0x19,
  "cpu MHz": 1795.926,
  "cache size": 3072 KB,
  "physical id": 0
}
```

Figura 4: comandos de enunciado del 9 y 10

```
GNU nano 2.9.3 pcinfo/CPU_ie0117.txt Modified
bugs: cpu_meltdown spectre_v1 spectre_v2
bogomips: 3591.85 : 1795.926
clflush size: 64 : 3072 KB
cache alignment: 64
address sizes: 36 bits physical, 48 bits virtual
power management:
Fabricante genuineintel
modelo 58
nucleos 2
cache 3072 KB
cpu Mhz 1795.926
```

Figura 5: archivo CPU_ie0117.txt modificado

```
ie0217@miguel:~$ rsync -avz -e "ssh -o StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/null" --progress ./id_rsa_text.txt jp@192.168.13.177:/home
Warning: Permanently added '192.168.13.177' (ECDSA) to the list of known hosts.
sending incremental file list
id_rsa_text.txt
rsync: mkstemp "/home/.id_rsa_text.txt.hDIRjp" failed: Permission denied (13)
sent 443 bytes received 117 bytes 65.88 bytes/sec
total size is 395 speedup is 0.71
rsync error: some files/attrs were not transferred (see previous errors) (code 2) at main.c(1196) [sender=3.1.2]
```

Figura 6: Uso de ssh para transferir un fichero a otra computadora

```
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow command
45 19 11 * * rsync -avz -e "ssh -o StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/null" --progress ./id_rsa_text.txt jp@192.168.13.177:/home/jp
```

Figura 7: Uso de crontab y ssh para respaldar periodicamente un fichero a otra computadora

```
ie0217@miguel:~$ wget https://ffmpeg.org/releases/ffmpeg-4.0.2.tar.bz2
--2018-09-11 19:48:25-- https://ffmpeg.org/releases/ffmpeg-4.0.2.tar.bz2
Resolving ffmpeg.org (ffmpeg.org)... 79.124.17.100
Connecting to ffmpeg.org (ffmpeg.org)|79.124.17.100|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10424065 (9.9M) [application/x-bzip2]
Saving to: 'ffmpeg-4.0.2.tar.bz2'
ffmpeg-4.0.2.tar.bz2 100%[=====] 9.94M 1.11MB/s in 17s
2018-09-11 19:48:43 (584 KB/s) - 'ffmpeg-4.0.2.tar.bz2' saved [10424065/10424065]
Programa a través del código fuente descargado.
```

Figura 8: Instalacion de programas desde código fuente