

**Universidad Tecnológica de Panamá**  
**Facultad de Ingeniería de Sistemas Computacionales**  
**Laboratorio # 6 - Sistemas Operativos I**

**Profesora: Aris Castillo**  
**Estudiante: Gabriel Díaz**  
**Cedula: 20-53-5198**  
**Grupo: 1 IF131**

**Objetivos.**

- Crear enlaces simbólicos de un archivo o directorio.
- Comandos: su, ln, chmod, chown, umask,
- Uso de pipe

**1. Comando su**

Switch user. Permite iniciar una sesión con otro usuario.

Ejemplo:  
`su root`

**2. Comando ln**

***Hard Link***

El comando ln permite que un archivo en el disco sea accedido con más de un nombre de archivo a través de un enlace duro. El comando crea un nuevo archivo y lo conecta con el original. Permite que dos usuarios compartan el mismo archivo o accederlo cuando se encuentra dentro de una cadena larga de subdirectorios.

Sintaxis:

`ln ruta_archivo nombre_acceso`

Ejemplo:

**`ln /home/trabajos/tareas/formula.txt formulatemp`**

Procedimiento:

- Cree el archivo formula.txt en dir1 el cual está en su directorio regular de trabajo (No. Grupo). Si no existen los directorios, créelos.
- El archivo formula.txt contiene las fórmulas de área de un triángulo, rectángulo y círculo.
- Desde el directorio de trabajo (~) cree el enlace
- Añada una línea al archivo formula.txt usando el enlace.
- Verifique el contenido de ambos archivos – formula.txt y formulatemp.txt
- Borre el enlace
- Escriba cada uno de los comandos para realizar estas tareas.

*Nota: formulatemp es un enlace al archivo formula.txt el cual se encuentra dentro de toda la cadena de directorios /home/trabajos/tareas/. Sólo existe una copia del archivo, pero puede ser accedido rápidamente por el enlace. Cuando se trate de borrar el archivo, Linux no lo permitirá mientras exista el enlace.*

***Symbolic Link***

Permite crear un enlace a un directorio. También es requerido cuando se desee enlazar un archivo en otra partición de disco o en la red. Para crear un enlace simbólico se debe agregar el parámetro -s al comando ln.

**ln -s /dev/fd0 A:**  
**ln -s /etc/httpd/conf web**

El primer ejemplo permite acceder al floppy como A y el segundo crea un enlace al directorio web en lugar de escribir toda la ruta **/etc/httpd/conf**.

Ejercicio:

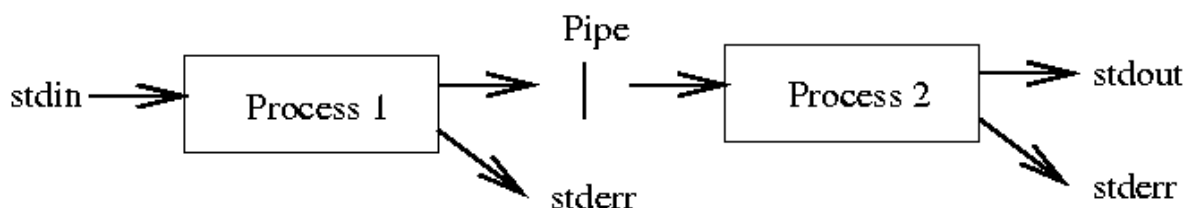
- Ubíquese en el directorio de trabajo de su usuario
- Haga un enlace simbólico a su directorio dir1 localizado dentro de su directorio de trabajo (no. Grupo). Nómbrelo midir
- Entre al enlace recién creado

```
gabrieldiaz@alex09-VirtualBox: ~/Desktop
gabrieldiaz@alex09-VirtualBox:~/Desktop$ ln -s /home/gabrieldiaz/Documents/dir1/
gabrieldiaz@alex09-VirtualBox:~/Desktop$
```

- Liste el contenido del directorio
- Haga un enlace a su unidad usb (ver /media para confirmar que la unidad está montada)
- Borre los enlaces
- Escriba cada uno de los comandos para realizar estas tareas.

### 3. Uso de filtro o pipe (|)

El pipe permite enlazar la salida de varios comandos juntos. Permite que procesos separados se comuniquen sin que explícitamente estén diseñados para trabajar juntos y que funcionen en forma compleja.



El símbolo “|” especifica que toda la salida (stdout) será entubada como entrada para el siguiente comando.

Ejemplo 1:

```
cd /lib
ls -l | more
ls -l | grep read
```

Ejemplo 2:

```
ps aux | grep sort
```

La salida nos permite ver específicamente si la aplicación sort está en ejecución. El comando ps brinda la lista de los procesos corriendo, y con grep se reduce la salida a unos cuantos procesos.

```
gabrieldiaz@alex09-VirtualBox: ~
gabrieldiaz@alex09-VirtualBox:~$ ps -aux | grep sort
gabriel+  2654  0.0  0.0 11568  724 pts/0    S+   08:41   0:00 grep --color=auto sort
gabrieldiaz@alex09-VirtualBox:~$
```

Ejemplo 3:

```
ls / | grep x
```

Bash y otros shells corren ambos comandos, conectando la salida del primero a la entrada del segundo; ls produce la lista de archivos en el directorio actual, mientras que grep toma dicha lista e imprime sólo aquellas líneas que contengan la letra x.

#### 4. Comando chmod

Permite cambiar los permisos a archivos.

Sintaxis:

```
chmod [option] archivo
```

```
u  permisos del usuario (dueño)
g  permisos del grupo
o  permisos de otros (público)
a  todos los usuarios, permisos de otros y del grupo
=  asigna un permiso absolutamente
+  agrega un permiso
-  retira un permiso
```

Ejemplos:

```
chmod u+x test1
chmod 700 test2
```

Con la primera línea se ha dado permiso de ejecución al dueño del archivo test1 y con la segunda se ha hecho lo mismo, usando representación octal, para el archivo test2 al dueño.

#### 5. Comando umask:

Establece los permisos por defecto. Debe darse en representación octal:

	<b>Owner</b>	<b>Group</b>	<b>Public (others)</b>
	r w x	r w x	r w x
R ----	400	40	4
-			
W --	200	20	2
---			
X --	100	10	1
---			

Ejemplo:

```
umask 644
```

```
400
200
40
4
---
644
```

644 es rw-r--r--

El dueño puede leer y escribir el archivo o directorio, los demás sólo pueden leerlo.

**Numeración octal para algunos permisos comunes:**

755 es rwxr-xr-x

El dueño puede leer, escribir y ejecutar el archivo, los demás sólo pueden leer o ejecutarlo. Para un directorio, esto es equivalente a 644.

711 es rwx--x--x

El dueño puede leer, escribir y ejecutar el archivo, los demás sólo pueden ejecutarlo.

444 es r--r--r--

Este permiso significa solo lectura para todos.

## 6. Comando chown

Cambia el dueño y el grupo a un archivo.

```
chown [OPTION]... OWNER[:[GROUP]] FILE...  
chown [OPTION]... :GROUP FILE...
```

Opciones:

-R, --recursivo: opera en archivos y directorios recursivamente.

Ejemplo:

```
chown estudiante1 test1
```

Ahora el dueño del archivo test1 es estudiante1. Todos los permisos del archivo pasan al nuevo dueño - usuario estudiante1.

## 7. Comando echo:

Despliega una línea de texto.

Ejemplo:

```
#echo hello  
#echo $HOME
```

## 8. which:

Muestra la ruta completa de los comandos del shell.

Ejemplo:

```
# which emacs  
/usr/bin/emacs
```

## 9. whereis:

Busca en una serie de lugares estándares de Linux. Muestra el binario, el fuente, y las páginas de ayuda del archivo o comando.

Ejemplo:

```
# whereis ifconfig
```

# Variables de Ambiente

**env:** lista las variables de ambiente.

Ejemplo:

```
#env  
.  
.  
.  
USERNAME=aris  
MAIL=/var/spool/aris
```

```

PATH=/usr/kerberos/bin:/usr/local/bin:/usr/bin:/bin:/home/aris/bin
DESKTOP_SESSION=gnome
PWD=/home/aris
LANG=en_US.utf8
.
.
.

```

**\$PATH:** Environment variable that stores the search path for executables.

Ejemplo:

```

echo $PATH
/usr/kerberos/bin:/usr/local/bin:/usr/bin:/bin:/home/aris/bin

```

**\$HOME:**

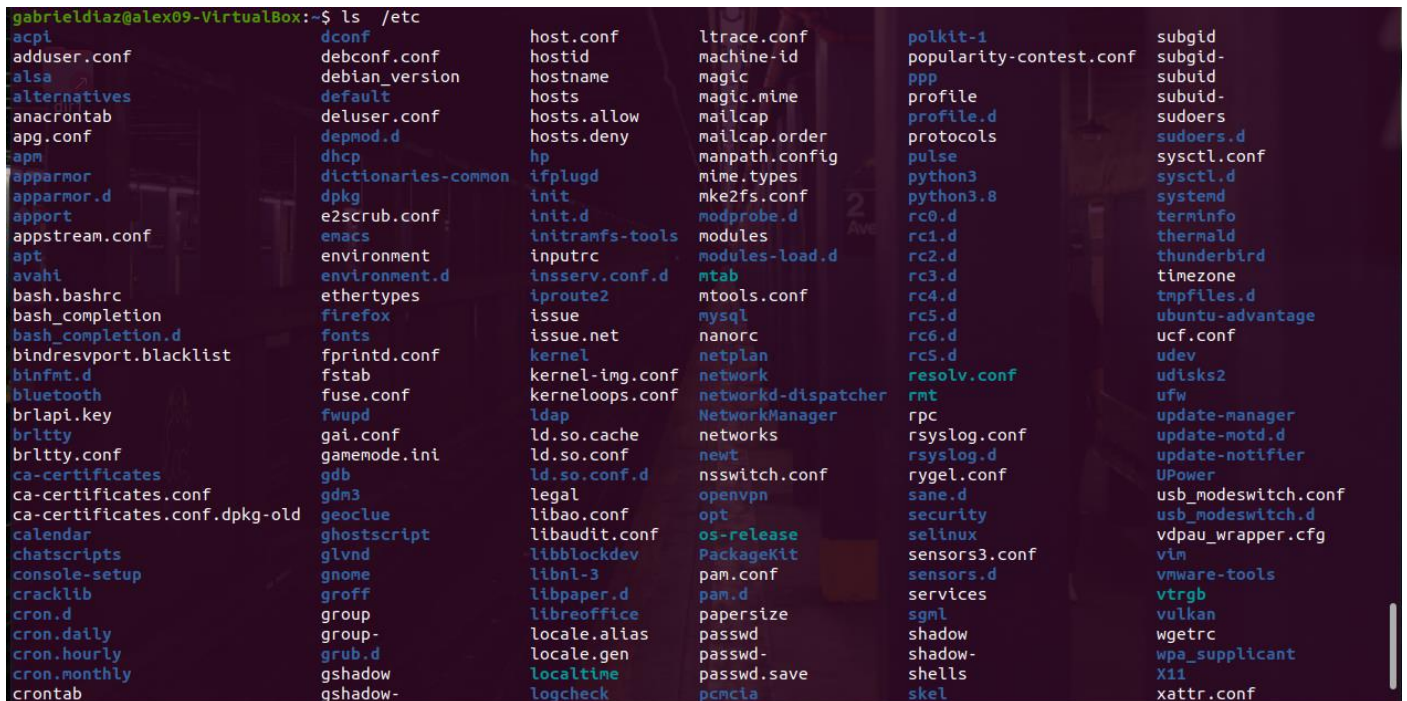
**export:** Establece una variable de ambiente en el bash shell.

Ejemplo:

```
export PATH=$PATH:/sbin
```

## Ejercicios:

1. Muestre en pantalla en formato completo el contenido del directorio etc, pantalla por pantalla.



```

gabrielidiaz@alex09-VirtualBox:~$ ls /etc
acpi          dconf          host.conf      ltrace.conf    polkit-1       subgid
adduser.conf  debconf.conf   hostid         machine-id     popularity-contest.conf subgid-
alsa          debian_version hostname       magic          ppp            subuid
alternatives  default        hosts          magic.mime     profile        subuid-
anacrontab    deluser.conf  hosts.allow   mailcap        profile.d      sudoers
apg.conf      depmod.d       hosts.deny    mailcap.order  protocols      sudoers.d
apn           dhcp          hp            manpath.config pulse          sysctl.conf
apparmor      dictionaries-common ifplugd        mime.types     python3        sysctl.d
apparmor.d    dpkg          init          mke2fs.conf   python3.8      systemd
appport       e2scrub.conf  init.d        modprobe.d     rc0.d          terminfo
appstream.conf emacs          initramfs-tools modules         rc1.d          thermald
apt           environment    inputrc       modules-load.d rc2.d          thunderbird
avahi         environment.d  inserv.conf.d mtab           rc3.d          timezone
bash.bashrc   ethertypes    iproute2      mysql          rc4.d          tmpfiles.d
bash_completion fireffox       issue         nanorc         rc5.d          ubuntu-advantage
bash_completion.d fonts          issue.net     netplan        rc6.d          ucf.conf
bindresvport.blacklist fprintd.conf  kernel        network        rc5.d          udev
binfmt.d      fstab          kernel-img.conf networkd-dispatcher rmt            udisks2
bluetooth     fuse.conf     kernelloops.conf networkmanager  rsyslog.conf  ufw
brlapi.key    fwupd         ldap          networks      rsyslog.d     update-manager
brltty        gai.conf      ld.so.cache   newt           update-motd.d update-notifier
brltty.conf   gamemode.ini  ld.so.conf.d  nsswitch.conf  rygel.conf    UPower
ca-certificates gdm3          legal         openvpn        sane.d         usb_modeswitch.conf
ca-certificates.conf gdm3         libao.conf    opt            security       usb_modeswitch.d
ca-certificates.conf.dpkg-old geoclue       libaudit.conf libblockdev    selinux        vdpau_wrapper.cfg
calendar      glvnd         gnome         libl-3         sensors3.conf  vim
chatscripts   groff         group         libreoffice    sensors.d      vmware-tools
console-setup groff         group         locale.alias   services       vtrgb
cracklib      grub.d        gshadow       locale.gen     sgml           vulkan
cron.d        grub.d        gshadow       localtime     shadow         wgetrc
cron.daily    grub.d        gshadow       localtime     shadow         wpa_supplicant
cron.hourly   grub.d        gshadow       localtime     shells         X11
cron.monthly  grub.d        gshadow       localtime     skel           xattr.conf
crontab       gshadow-     logcheck      pcmcia

```

2. Busque los archivos con caracteres host en el directorio etc y que le detalle la cantidad de líneas de los mismos.

```
gabrielldiaz@alex09-VirtualBox: ~  
gabrielldiaz@alex09-VirtualBox:~$ ls -ls /etc | grep host  
4 drwxr-xr-x 4 root root 4096 jul 31 11:29 ghostscript  
4 -rw-r--r-- 1 root root 92 dic 5 2019 host.conf  
4 -rw-r--r-- 1 root root 4 jul 31 11:33 hostid  
4 -rw-r--r-- 1 root root 18 ago 29 19:40 hostname  
4 -rw-r--r-- 1 root root 232 ago 29 19:40 hosts  
4 -rw-r--r-- 1 root root 411 jul 31 11:30 hosts.allow  
4 -rw-r--r-- 1 root root 711 jul 31 11:30 hosts.deny  
gabrielldiaz@alex09-VirtualBox:~$
```

3. Visualice el contenido del directorio /usr/bin pantalla por pantalla.

```
gabrielldiaz@alex09-VirtualBox: ~  
gabrielldiaz@alex09-VirtualBox:~$ ls -la /usr/bin | more  
total 175240  
drwxr-xr-x 2 root root 45056 oct 1 17:49 .  
drwxr-xr-x 14 root root 4096 jul 31 11:28 ..  
-rw-r--r-- 1 root root 59736 sep 5 2019 [  
-rw-r--r-- 1 root root 31248 may 19 11:59 aa-enabled  
-rw-r--r-- 1 root root 35344 may 19 11:59 aa-exec  
-rw-r--r-- 1 root root 22912 mar 4 2020 aconnect  
-rw-r--r-- 1 root root 19016 nov 28 2019 acpi_listen  
-rw-r--r-- 1 root root 7415 ago 7 08:15 add-apt-repository  
-rw-r--r-- 1 root root 30952 jul 21 02:49 addpart  
lrwxrwxrwx 1 root root 26 abr 7 06:42 addr2line -> x86_64-linux-gnu-addr2line  
-rw-r--r-- 1 root root 47552 mar 4 2020 alsabat  
-rw-r--r-- 1 root root 85296 mar 4 2020 alsaloop  
-rw-r--r-- 1 root root 72432 mar 4 2020 alsamixer  
-rw-r--r-- 1 root root 14720 mar 4 2020 alsatplg  
-rw-r--r-- 1 root root 31528 mar 4 2020 alsaucm  
-rw-r--r-- 1 root root 31112 mar 4 2020 amidi  
-rw-r--r-- 1 root root 63952 mar 4 2020 amixer  
-rw-r--r-- 1 root root 2668 mar 22 2020 amuFormat.sh  
-rw-r--r-- 1 root root 274 oct 1 2017 apg  
-rw-r--r-- 1 root root 26696 oct 1 2017 apgbfm  
-rw-r--r-- 1 root root 84400 mar 4 2020 aplay  
-rw-r--r-- 1 root root 27016 mar 4 2020 aplaymidi  
-rw-r--r-- 1 root root 2558 dic 4 2019 apport-bug  
-rw-r--r-- 1 root root 13367 ago 9 06:43 apport-cli  
lrwxrwxrwx 1 root root 10 ago 9 06:43 apport-collect -> apport-bug  
-rw-r--r-- 1 root root 2068 ago 9 06:43 apport-unpack  
-rw-r--r-- 1 root root 14648 feb 29 2020 appres  
-rw-r--r-- 1 root root 67816 mar 14 2020 appstreamcli  
lrwxrwxrwx 1 root root 6 ago 29 19:37 apropos -> whatis  
-rw-r--r-- 1 root root 18824 may 12 15:02 apt  
lrwxrwxrwx 1 root root 18 ago 7 08:15 apt-add-repository -> add-apt-repository  
-rw-r--r-- 1 root root 88536 may 12 15:02 apt-cache  
-rw-r--r-- 1 root root 31192 may 12 15:02 apt-cdrom  
-rw-r--r-- 1 root root 27016 may 12 15:02 apt-config  
--More--
```

4. Cree un archivo en vi con 5 líneas de texto (use echo en cada línea). Luego cambie los permisos de lectura, escritura y ejecución del mismo. Ejecútelo.

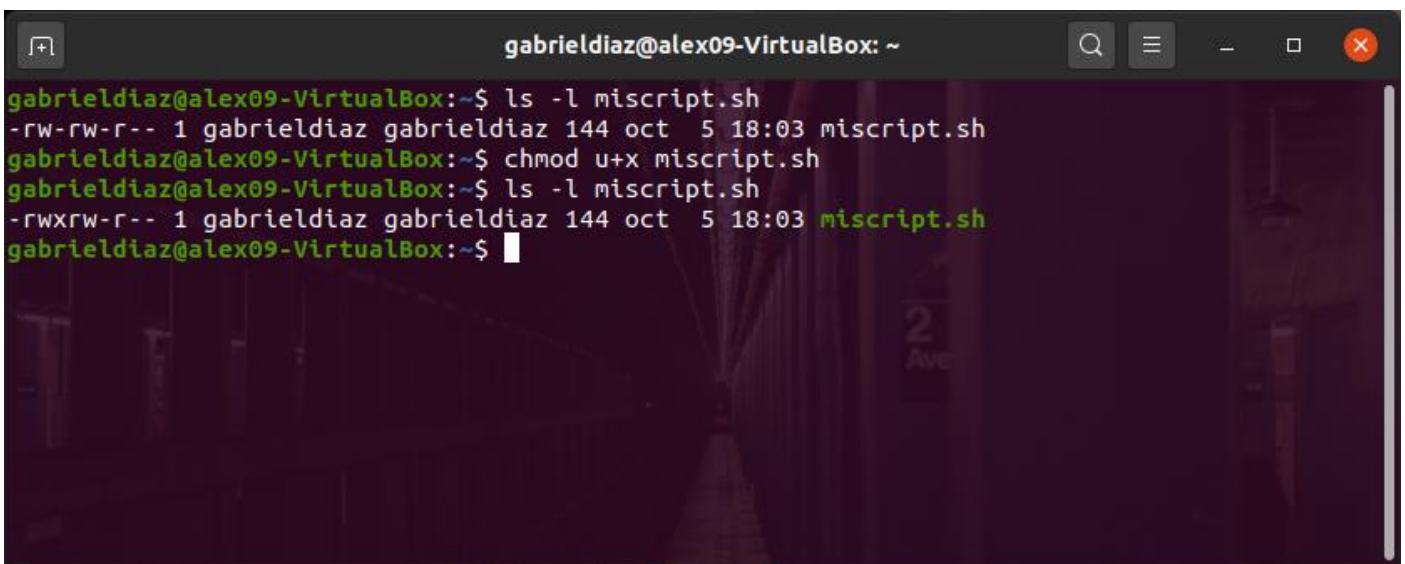


### Paso 1: Creando el script con vim



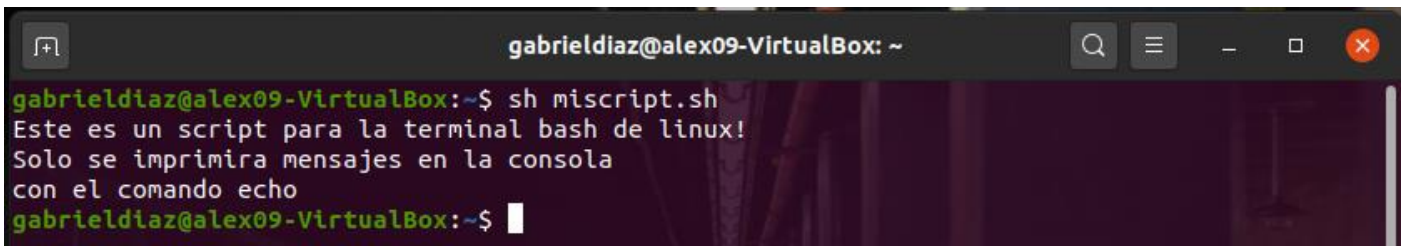
```
gabrieldiaz@alex09-VirtualBox: ~
#!/bin/bash
echo Este es un script para la terminal bash de linux!
echo Solo se imprimira mensajes en la consola
echo con el comando "echo"
:wq
```

### Paso 2: Cambiando los permisos de usuario



```
gabrieldiaz@alex09-VirtualBox: ~
gabrieldiaz@alex09-VirtualBox:~$ ls -l miscript.sh
-rw-rw-r-- 1 gabrieldiaz gabrieldiaz 144 oct  5 18:03 miscript.sh
gabrieldiaz@alex09-VirtualBox:~$ chmod u+x miscript.sh
gabrieldiaz@alex09-VirtualBox:~$ ls -l miscript.sh
-rwxrw-r-- 1 gabrieldiaz gabrieldiaz 144 oct  5 18:03 miscript.sh
gabrieldiaz@alex09-VirtualBox:~$
```

### Paso 3: Ejecutando el script con extension .sh



```
gabrieldiaz@alex09-VirtualBox: ~
gabrieldiaz@alex09-VirtualBox:~$ sh miscript.sh
Este es un script para la terminal bash de linux!
Solo se imprimira mensajes en la consola
con el comando echo
gabrieldiaz@alex09-VirtualBox:~$
```

5. Cámbiese a otro usuario. Pruebe si puede ejecutar el archivo. Realice las pruebas necesarias para verificar los permisos asignados al archivo.

Paso 1: Cambiando de usuario con el comando su –

```
gabrieldiaz@alex09-VirtualBox: ~  
gabrieldiaz@alex09-VirtualBox:~$ su - alex09  
Password: 
```

Paso 2: Ubicar el fichero miscript.sh e intentar ejecutarlo

```
alex09@alex09-VirtualBox: /home/gabrieldiaz  
alex09@alex09-VirtualBox:/home/gabrieldiaz$ ls -l miscript.sh  
-rwxrw-r-- 1 gabrieldiaz gabrieldiaz 144 oct  5 18:03 miscript.sh  
alex09@alex09-VirtualBox:/home/gabrieldiaz$ sh miscript.sh  
Este es un script para la terminal bash de linux!  
Solo se imprimira mensajes en la consola  
con el comando echo  
alex09@alex09-VirtualBox:/home/gabrieldiaz$ 
```

6. Cambie el dueño del archivo (debe hacerlo con el usuario root). Realice todas las pruebas de verificación nuevamente.

Paso 1: Cambiando al usuario root

```
root@alex09-VirtualBox: ~  
alex09@alex09-VirtualBox:/home/gabrieldiaz$ su - root  
Password:  
root@alex09-VirtualBox:~# 
```

Paso 2: Cambiando el dueno del fichero con el comando chown

```
root@alex09-VirtualBox: /home/gabrieldiaz  
root@alex09-VirtualBox:/home/gabrieldiaz# ls -l miscript.sh  
-rwxrw-r-- 1 gabrieldiaz gabrieldiaz 144 oct  5 18:03 miscript.sh  
root@alex09-VirtualBox:/home/gabrieldiaz# chown root miscript.sh  
root@alex09-VirtualBox:/home/gabrieldiaz# ls -l miscript.sh  
-rwxrw-r-- 1 root gabrieldiaz 144 oct  5 18:03 miscript.sh  
root@alex09-VirtualBox:/home/gabrieldiaz# 
```

Paso 3: Ejecutando el fichero con un usuario diferente de root

```
alex09@alex09-VirtualBox: /home/gabrieldiaz  
alex09@alex09-VirtualBox:/home/gabrieldiaz$ ls -l miscript.sh  
-rwxrw-r-- 1 root gabrieldiaz 144 oct  5 18:03 miscript.sh  
alex09@alex09-VirtualBox:/home/gabrieldiaz$ sh miscript.sh  
Este es un script para la terminal bash de linux!  
Solo se imprimira mensajes en la consola  
con el comando echo  
alex09@alex09-VirtualBox:/home/gabrieldiaz$ 
```