

1.- Conceptos Básicos y Variables

1.1

```
alumno@servidor:~$ echo $HOME
/home/alumno
alumno@servidor:~$ cd $HOME
alumno@servidor:~$ pwd
/home/alumno
alumno@servidor:~$ r
```

1.2

```
alumno@servidor:~$ whoami
alumno
alumno@servidor:~$ USUARIO_ACTUAL=$(whoami)
alumno@servidor:~$ echo $USUARIO_ACTUAL
whoami
alumno@servidor:~$ _
```

1.3

```
alumno@servidor:~$ touch dos palabras.txt
alumno@servidor:~$ ls
dos palabras.txt  script
alumno@servidor:~$ rm dos palabras.txt
alumno@servidor:~$ touch "dos palabras.txt"
alumno@servidor:~$ ls
'dos palabras.txt'  script
```

1.4

```
alumno@servidor:~$ type ls
ls is aliased to `ls --color=auto'
alumno@servidor:~$ type -a ls
ls is aliased to `ls --color=auto'
ls is /usr/bin/ls
ls is /bin/ls
alumno@servidor:~$ type cd
cd is a shell builtin
alumno@servidor:~$ _
```

1.5

```
alumno@servidor:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
alumno@servidor:~$ mkdir ~/mi_bin
alumno@servidor:~$ ls
'dos palabras.txt'  mi_bin  script
alumno@servidor:~$ export PATH=~/mi_bin:$PATH
alumno@servidor:~$ echo $PATH
/home/alumno/mi_bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
alumno@servidor:~$
```

2.- Obtener Ayuda y Localizar Archivos

2.1

```
CHMOD(1) User Commands CHMOD(1)
NAME
  chmod - change file mode bits
SYNOPSIS
  chmod [OPTION]... MODE[,MODE]... FILE...
  chmod [OPTION]... OCTAL-MODE FILE...
  chmod [OPTION]... --reference=RFILE FILE...
DESCRIPTION
  This manual page documents the GNU version of chmod.  chmod changes the file mode bits of each given file according to mode, which can be either a symbolic representation of changes to make, or an octal number representing the bit pattern for the new mode bits.
  The format of a symbolic mode is [ugoa...][[[-*]]perms...], where perms is either zero or more letters from the set rwxXst, or a single letter from the set ugo. Multiple symbolic modes can be given, separated by commas.
  A combination of the letters ugoa controls which users' access to the file will be changed: the user who owns it (u), other users in the file's group (g), other users not in the file's group (o), or all users (a). If none of these are given, the effect is as if (a) were given, but bits that are set in the umask are not affected.
  The operator + causes the selected file mode bits to be added to the existing file mode bits of each file; - causes them to be removed; and = causes them to be added and causes unmentioned bits to be removed except that a directory's unmentioned set user and group ID bits are not affected.
  The letters rwxXst select file mode bits for the affected users: read (r), write (w), execute (or search for directories) (x), execute/search only if the file is a directory or already has execute permission for some user (X), set user or group ID on execution (s), restricted deletion flag or sticky bit (t). Instead of one or more of these letters, you can specify exactly one of the letters ugo: the permissions granted to the user who owns the file (u), the permissions granted to other users who are members of the file's group (g), and the permissions granted to users that are in neither of the two preceding categories (o).
  A numeric mode is from one to four octal digits (0-7), derived by adding up the bits with values 4, 2, and 1. Omitted digits are assumed to be leading zeros. The first digit selects the set user ID (4) and set group ID (2) and restricted deletion or sticky (1) attributes. The second digit selects permissions for the user who owns the file: read (4), write (2), and execute (1); the third selects permissions for other users in the file's group, with the same values; and the fourth for other users not in the file's group, with the same values.
  chmod never changes the permissions of symbolic links: the chmod system call cannot change their permissions. This is not a problem since the permissions of symbolic links are never used. However, for each symbolic link listed on the command line, chmod changes the permissions of the pointed-to file. In contrast, chmod ignores symbolic links encountered during recursive directory traversals.
SETUID AND SETGID BITS
  chmod clears the set-group-ID bit of a regular file if the file's group ID does not match the user's effective group ID or one of the user's supplementary group IDs, unless the user has appropriate privileges. Additional restrictions may cause the set-user-ID and set-group-ID bits of MODE or RFILE to be ignored. This behavior depends on the policy and functionality of the underlying chmod system call. When in doubt, check the underlying system behavior.
  For directories, chmod preserves set-user-ID and set-group-ID bits unless you explicitly specify otherwise. You can set or clear the bits with symbolic modes like u+s and g-s. To clear these bits for directories with a numeric mode requires an additional leading zero like 00755, leading minus like -6000, or leading equals like =755.
Manual page chmod(1) line 1 (press h for help or q to quit)
```

Se encuentra en la sección 1, ya que arriba nos lo indica con el CHMOD(1)

```
CHMOD(1)
NAME
  chmod - change file mode bits
```

Indica que es un comando de Usuario

2.2

```
-S      sort by file size, largest first
```

El comando “ls -S” es el que ordena los archivos por tamaño

2.3

```
alumno@servidor:~$ sudo find / -name passwd
[sudo] password for alumno:
/usr/share/doc/passwd
/usr/share/lintian/overrides/passwd
/usr/share/bash-completion/completions/passwd
/usr/bin/passwd
/etc/passwd
/etc/pam.d/passwd
alumno@servidor:~$
```

2.4

```
alumno@servidor:~$ touch test_locate.txt
alumno@servidor:~$ locate test_locate.txt
Command 'locate' not found, but can be installed with:
sudo apt install plocate
alumno@servidor:~$ _
```

No lo encuentra porque el comando “locate” no está instalado

2.5

Ejecutamos el comando para instalar y al buscar el archivo de nuevo, sí que nos aparece la ruta

```
alumno@servidor:~$ sudo apt install plocate
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  liburing2
Se instalarán los siguientes paquetes NUEVOS:
  liburing2 plocate
```

```
alumno@servidor:~$ locate test_locate.txt
/home/alumno/test_locate.txt
alumno@servidor:~$
```

3.1

```
alumno@servidor:~$ cd /etc
alumno@servidor:/etc$ ls ~
'dos palabras.txt'  mi_bin  script  test_locate.txt
alumno@servidor:/etc$ _
```

3.2

```
alumno@servidor:~$ cd ../../var/log
alumno@servidor:/var/log$ pwd
/var/log
alumno@servidor:/var/log$ _
```

3.3

```
-rw-r--r-- 1 root root 1737 oct 15 09:56 /etc/passwd
```

3.4

Hace que los datos de los tamaños de los archivos sean más legibles para el Usuario

3.5

```
alumno@servidor:~$ ls -R ~  
/home/alumno:  
'dos palabras.txt'  mi_bin  script  test_locate.txt  
  
/home/alumno/mi_bin:  
alumno@servidor:~$
```

Si lo usáramos en el directorio raíz, nos listaría todos los directorios y los archivos de sistema, esto es peligroso porque la cantidad de archivos que tienen esos directorios podría hacer que el sistema se vuelva más lento ya que le estaríamos pidiendo una cantidad inmensa de datos.

4.1

```
alumno@servidor:~$ mkdir -p proyecto/src proyecto/doc proyecto/bin  
alumno@servidor:~$ tree proyecto  
proyecto  
├── bin  
├── doc  
└── src  
  
4 directories, 0 files  
alumno@servidor:~$
```

Uso el comando tree para que se vea que se han creado los archivos

4.2

```
alumno@servidor:~$ touch ~/notas.txt  
alumno@servidor:~$ mv ~/notas.txt ~/proyecto/doc/README.md  
alumno@servidor:~$ tree  
.  
├── dos palabras.txt  
├── mi_bin  
├── proyecto  
│   ├── bin  
│   ├── doc  
│   │   └── README.md  
│   └── src  
├── script  
└── test_locate.txt  
  
6 directories, 4 files  
alumno@servidor:~$
```

4.3

```
alumno@servidor:~$ cp ~/proyecto/doc/README.md ~/proyecto/bin/
alumno@servidor:~$ tree proyecto
proyecto
├── bin
│   └── README.md
├── doc
│   └── README.md
└── src

4 directories, 2 files
alumno@servidor:~$ rm ~/proyecto/doc/README.md
alumno@servidor:~$ tree proyecto
proyecto
├── bin
│   └── README.md
├── doc
└── src

4 directories, 1 file
alumno@servidor:~$ _
```

4.4

```
alumno@servidor:~$ rmdir ~/proyecto
rmdir: failed to remove '/home/alumno/proyecto': Directory not empty
alumno@servidor:~$
```

No borra el directorio ya que no está vacío

Unos pocos archivos que me devolvió el comando

```
var/log/
var/log/apt/
var/log/apt/history.log
var/log/apt/eipp.log.xz
var/log/apt/term.log
var/log/lastlog
var/log/cloud-init-output.log
var/log/landscape/
var/log/landscape/sysinfo.log
var/log/alternatives.log
var/log/cloud-init.log
var/log/README
var/log/kern.log
var/log/apport.log
var/log/dist-upgrade/
var/log/journal/
var/log/journal/ad660d64f8bf46158b0338bd3cfd3e9f/
```

5.4

```
alumno@servidor:~$ tar -xzf log_backup.tar.gz -C /tmp var/log/syslog
alumno@servidor:~$ ls -l /tmp/var/log/
total 868
-rw-r----- 1 alumno alumno 885364 oct 21 07:00 syslog
alumno@servidor:~$
```

5.5

```
alumno@servidor:~$ touch a.txt b.log c.jpg
alumno@servidor:~$ zip ejercicio5.zip a.txt b.log c.jpg
updating: a.txt (stored 0%)
updating: b.log (stored 0%)
updating: c.jpg (stored 0%)
alumno@servidor:~$ unzip -l ejercicio5.zip
Archive:  ejercicio5.zip
  Length      Date    Time    Name
  -----
      0  2025-10-21  07:39    a.txt
      0  2025-10-21  07:39    b.log
      0  2025-10-21  07:39    c.jpg
  -----
      0
                   3 files
alumno@servidor:~$
```

5.6

```
alumno@servidor:~$ rm a.txt b.log c.jpg
```

```
alumno@servidor:~$ ls -l
total 2940
-rw-rw-r-- 1 alumno alumno      0 oct 16 10:00 'dos palabras.txt'
-rw-rw-r-- 1 alumno alumno    436 oct 21 07:39  ejercicio5.zip
-rw-rw-r-- 1 alumno alumno 2104573 oct 21 07:01  log_backup.tar.gz
drwxrwxr-x 2 alumno alumno   4096 oct 16 10:13  mi_bin
-rwxrwxr-x 1 alumno alumno     12 oct 16 07:10  script
-rw-r----- 1 alumno alumno 885364 oct 21 07:00  syslog
-rw-rw-r-- 1 alumno alumno      0 oct 20 07:44  test_locate.txt
drwxrwxr-x 3 alumno alumno   4096 oct 21 07:18  var
alumno@servidor:~$
```

```
alumno@servidor:~$ unzip ejercicio5.zip
Archive:  ejercicio5.zip
  extracting: a.txt
  extracting: b.log
  extracting: c.jpg
alumno@servidor:~$ ls -l
total 2940
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:39  a.txt
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:39  b.log
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:39  c.jpg
-rw-rw-r-- 1 alumno alumno      0 oct 16 10:00 'dos palabras.txt'
-rw-rw-r-- 1 alumno alumno    436 oct 21 07:39  ejercicio5.zip
-rw-rw-r-- 1 alumno alumno 2104573 oct 21 07:01  log_backup.tar.gz
drwxrwxr-x 2 alumno alumno   4096 oct 16 10:13  mi_bin
-rwxrwxr-x 1 alumno alumno     12 oct 16 07:10  script
-rw-r----- 1 alumno alumno 885364 oct 21 07:00  syslog
-rw-rw-r-- 1 alumno alumno      0 oct 20 07:44  test_locate.txt
drwxrwxr-x 3 alumno alumno   4096 oct 21 07:18  var
```

5.7

```
alumno@servidor:~$ find /var/log -name '*.gz'
find: '/var/log/private': Permission denied
/var/log/dmesg.2.gz
/var/log/dmesg.4.gz
/var/log/dmesg.3.gz
/var/log/dmesg.1.gz
alumno@servidor:~$ zcat /var/log/dmesg.3.gz
```

6.1

```
alumno@servidor:~$ ls -l ~ > ~/misarchivos.txt
alumno@servidor:~$ cat misarchivos.txt
total 2940
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:39 a.txt
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:39 b.log
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:39 c.jpg
-rw-rw-r-- 1 alumno alumno      0 oct 16 10:00 dos palabras.txt
-rw-rw-r-- 1 alumno alumno    436 oct 21 07:39 ejercicio5.zip
-rw-rw-r-- 1 alumno alumno 2104573 oct 21 07:01 log_backup.tar.gz
drwxrwxr-x 2 alumno alumno    4096 oct 16 10:13 mi_bin
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:54 misarchivos.txt
-rwxrwxr-x 1 alumno alumno     12 oct 16 07:10 script
-rw-r----- 1 alumno alumno 885364 oct 21 07:00 syslog
-rw-rw-r-- 1 alumno alumno      0 oct 20 07:44 test_locate.txt
drwxrwxr-x 3 alumno alumno    4096 oct 21 07:18 var
alumno@servidor:~$
```

6.2

```
alumno@servidor:~$ date >> ~/misarchivos.txt
alumno@servidor:~$ cat misarchivos.txt
total 2940
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:39 a.txt
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:39 b.log
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:39 c.jpg
-rw-rw-r-- 1 alumno alumno      0 oct 16 10:00 dos palabras.txt
-rw-rw-r-- 1 alumno alumno    436 oct 21 07:39 ejercicio5.zip
-rw-rw-r-- 1 alumno alumno 2104573 oct 21 07:01 log_backup.tar.gz
drwxrwxr-x 2 alumno alumno    4096 oct 16 10:13 mi_bin
-rw-rw-r-- 1 alumno alumno      0 oct 21 07:54 misarchivos.txt
-rwxrwxr-x 1 alumno alumno     12 oct 16 07:10 script
-rw-r----- 1 alumno alumno 885364 oct 21 07:00 syslog
-rw-rw-r-- 1 alumno alumno      0 oct 20 07:44 test_locate.txt
drwxrwxr-x 3 alumno alumno    4096 oct 21 07:18 var
mar 21 oct 2025 07:56:41 UTC
alumno@servidor:~$ _
```

6.3

```
alumno@servidor:~$ ls -l | grep '^d' | wc -l
2
alumno@servidor:~$ _
```

6.4

```
alumno@servidor:~$ tail -n 10 /etc/passwd | cut -d: -f1
pollinate
polkitd
syslog
uuuid
tcpdump
tss
landscape
fwupd-refresh
usbmux
alumno
alumno@servidor:~$
```

6.5

```
alumno@servidor:~$ ps aux --sort=%cpu | head -n 6
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         2  0.0  0.0      0     0 ?        S    06:34   0:00 [kthreadd]
root         3  0.0  0.0      0     0 ?        S    06:34   0:00 [pool_workqueue_release]
root         4  0.0  0.0      0     0 ?        I<   06:34   0:00 [kworker/R-rcu_g]
root         5  0.0  0.0      0     0 ?        I<   06:34   0:00 [kworker/R-rcu_p]
root         6  0.0  0.0      0     0 ?        I<   06:34   0:00 [kworker/R-slub_]
alumno@servidor:~$
```

6.6

>: Crea un archivo si no existe y le añade texto, pero si escribes más texto y vuelves a ejecutarlo con > se sobrescribe por lo que el primer mensaje ya no existe.

>>: Crea un archivo si no existe y le añade texto, si se escribe más texto y se ejecuta con >> el nuevo texto se añade al final.

```
alumno@servidor:~$ echo "hola" > ejercicio6.txt
alumno@servidor:~$ cat ejercicio6.txt
hola
alumno@servidor:~$ echo "adios" > ejercicio6.txt
alumno@servidor:~$ cat ejercicio6.txt
adios
alumno@servidor:~$ echo "hola" >> ejercicio6_1.txt
alumno@servidor:~$ cat ejercicio6_1.txt
hola
alumno@servidor:~$ echo "adios" >> ejercicio6_1.txt
alumno@servidor:~$ cat ejercicio6_1.txt
hola
adios
alumno@servidor:~$ _
```


6.7

```
alumno@servidor:~$ find /etc -name "*.conf" > config_files.text 2> errors.txt
alumno@servidor:~$ cat config_files.text | head
/etc/apt/apt.conf.d/20apt-esm-hook.conf
/etc/apt/apt.conf.d/20snapd.conf
/etc/adduser.conf
/etc/logrotate.conf
/etc/nsswitch.conf
/etc/e2scrub.conf
/etc/modules-load.d/modules.conf
/etc/depmod.d/ubuntu.conf
/etc/systemd/networkd.conf
/etc/systemd/user.conf
alumno@servidor:~$ cat errors.txt
find: '/etc/credstore': Permission denied
find: '/etc/polkit-1/rules.d': Permission denied
find: '/etc/credstore.encrypted': Permission denied
find: '/etc/ssl/private': Permission denied
find: '/etc/multipath': Permission denied
alumno@servidor:~$
```

7.1

```
alumno@servidor:~$ touch script1.sh
alumno@servidor:~$ nano script1.sh _
```

```
GNU nano 7.2
echo "Nombre de Usuario: $USER"
echo "Directorio de trabajo: $PWD"
```

```
alumno@servidor:~$ ./script1.sh
Nombre de Usuario: alumno
Directorio de trabajo: /home/alumno
alumno@servidor:~$ _
```

7.2

```
alumno@servidor:~$ chmod u+x script1.sh
alumno@servidor:~$ ls -l script1.sh
-rwxrwxr-x 1 alumno alumno 67 oct 22 07:21 script1.sh
alumno@servidor:~$ _
```

Si lo ejecuta otro usuario le saldría un error de permiso denegado, porque no tiene los permisos para ejecutar ese script.

7.3

```
GNU nano 7.2
echo "Nombre de Usuario: $USER"
echo "Directorio de trabajo: $PWD"

if [ "$1" = "hola" ]; then
    echo "mundo"
fi_
```

```
alumno@servidor:~$ ./script1.sh hola
Nombre de Usuario: alumno
Directorio de trabajo: /home/alumno
mundo
alumno@servidor:~$ ./script1.sh adios
Nombre de Usuario: alumno
Directorio de trabajo: /home/alumno
alumno@servidor:~$
```

7.4

```
GNU nano 7.2
if [ "$1" = "hola" ]; then
    echo "mundo"
fi

if [ "$#" -eq 0 ]; then
    echo "Error: Debes proporcionar un argumento"
fi
_
```

```
alumno@servidor:~$ ./script1.sh
Error: Debes proporcionar un argumento
alumno@servidor:~$ ./script1.sh hola
mundo
alumno@servidor:~$ ./script1.sh adios
alumno@servidor:~$ _
```

7.5

```
GNU nano 7.2
if [ "$1" -eq "$2" ]; then
    echo "iguales"
else
    echo "diferentes"
fi
```

```
alumno@servidor:~$ ./script1.sh 1 4
diferentes
alumno@servidor:~$ ./script1.sh 1 1
iguales
alumno@servidor:~$
```

7.6

```
GNU nano 7.2
for archivo in $(ls "$1"); do
    if [[ "$archivo" != *.bak ]]; then
        mv "$1/$archivo" "$1/$archivo.bak"
        echo " Archivo renombrado de $archivo a $archivo.bak"
    fi
done_
```

```
alumno@servidor:~$ tree
```

```
.
├── a.txt
├── b.log
├── c.jpg
├── config_files.txt
├── dos palabras.txt
├── ejercicio5.zip
├── ejercicio6_1.txt
├── ejercicio6.txt
├── errors.txt
├── log_backup.tar.gz
├── ~m
├── mi_bin
├── misarchivos.txt
├── mv
├── prueba
│   └── prueba1.txt
├── script
├── script1.sh
├── syslog
├── test_locate.txt
├── var
│   └── log
│       └── syslog
```

```

alumno@servidor:~$ ./script1.sh ~/prueba
Archivo renombrado de prueba1.txt a prueba1.txt.bak
alumno@servidor:~$ tree
.
├── a.txt
├── b.log
├── c.jpg
├── config_files.text
├── dos palabras.txt
├── ejercicio5.zip
├── ejercicio6_1.txt
├── ejercicio6.txt
├── errors.txt
├── log_backup.tar.gz
├── ~m
├── mi_bin
├── misarchivos.txt
├── mv
├── prueba
│   └── prueba1.txt.bak

```

8.1

```

alumno@servidor:~$ cut -d: -f7 /etc/passwd | sort | uniq
/bin/bash
/bin/false
/bin/sync
/usr/sbin/nologin
alumno@servidor:~$ _

```

Uso el -f7 para que me saque por pantalla solo los shells de los usuarios

8.2

```

alumno@servidor:~$ ps aux | grep '^root' | wc -l
86
alumno@servidor:~$

```

8.3

```
alumno@servidor:~$ ls -l /etc | grep 'oct' > october_files.txt
```

Muestra de algunos resultados

```
drwxr-xr-x 2 root root      4096 oct 15 09:29 console-setup
drwxr-xr-x 2 root root      4096 oct 20 07:45 cron.daily
drwxr-xr-x 3 root root      4096 oct 15 09:37 default
drwxr-xr-x 4 root root      4096 oct 15 09:38 dpkg
-rw-r--r-- 1 root root     1853 oct 17 2022 ethertypes
drwxr-xr-x 4 root root      4096 oct 15 09:37 fonts
-rw-r--r-- 1 root root       657 oct 15 09:37 fstab
-rw-r--r-- 1 root root       823 oct 20 07:45 group
-rw-r--r-- 1 root root       808 oct 15 09:56 group-
drwxr-xr-x 2 root root      4096 oct 15 09:30 grub.d
-rw-r----- 1 root shadow    697 oct 20 07:45 gshadow
-rw-r----- 1 root shadow    685 oct 15 09:56 gshadow-
-rw-r--r-- 1 root root        9 oct 15 09:37 hostname
-rw-r--r-- 1 root root      223 oct 15 09:37 hosts
drwxr-xr-x 2 root root      4096 oct 15 09:38 init.d
drwxr-xr-x 6 root root      4096 oct 15 09:36 kernel
-rw-r--r-- 1 root root    22503 oct 20 07:45 ld.so.cache
drwxr-xr-x 2 root root      4096 oct 15 09:36 ld.so.conf.d
-rw-r--r-- 1 root root        17 oct 15 09:56 locale.conf
-rw-r--r-- 1 root root     9563 oct 15 09:56 locale.gen
drwxr-xr-x 2 root root      4096 oct 15 09:38 logrotate.d
```

8.4

```
alumno@servidor:~$ ls /etc/*[0-9]*
/etc/e2scrub.conf /etc/mke2fs.conf /etc/sensors3.conf

/etc/dbus-1:
session.d system.d

/etc/iproute2:
bof_pinning ematch_map group nl_protos rt_dsfield rt_protos rt_protos.d rt_realms rt_scopes rt_tables r

/etc/libnl-3:
classid pktloc

/etc/polkit-1:
rules.d

/etc/python3:
debian_config

/etc/python3.12:
sitecustomize.py

/etc/rc0.d:
K01cryptdisks K01cryptdisks-early K01iscsid K01open-iscsi K01open-vm-tools K01plymouth K01unattended-upgrad

/etc/rc1.d:
K01iscsid K01open-iscsi K01open-vm-tools K01ufw K01uuidd

/etc/rc2.d:
S01appport S01console-setup.sh S01cron S01dbus S01grub-common S01open-vm-tools S01plymouth S01rsync S01sys

/etc/rc3.d:
S01appport S01console-setup.sh S01cron S01dbus S01grub-common S01open-vm-tools S01plymouth S01rsync S01sys

/etc/rc4.d:
S01appport S01console-setup.sh S01cron S01dbus S01grub-common S01open-vm-tools S01plymouth S01rsync S01sys

/etc/rc5.d:
S01appport S01console-setup.sh S01cron S01dbus S01grub-common S01open-vm-tools S01plymouth S01rsync S01sys

/etc/rc6.d:
K01cryptdisks K01cryptdisks-early K01iscsid K01open-iscsi K01open-vm-tools K01plymouth K01unattended-upgrad

/etc/udisks2:
mount_options.conf.example udisks2.conf

/etc/X11:
xkb Xsession.d
alumno@servidor:~$ _
```

8.5

```
alumno@servidor:~$ find /usr/bin -executable ! -user root  
alumno@servidor:~$
```

8.6

```
alumno@servidor:~$ bzip2 -c /var/log/syslog > syslog.bz2  
alumno@servidor:~$ gzip -c /var/log/syslog > syslog.gz  
alumno@servidor:~$ ls -l syslog.bz2 syslog.gz  
-rw-rw-r-- 1 alumno alumno 88390 oct 22 09:42 syslog.bz2  
-rw-rw-r-- 1 alumno alumno 214697 oct 22 09:43 syslog.gz  
alumno@servidor:~$ ls -lh syslog.bz2 syslog.gz  
-rw-rw-r-- 1 alumno alumno 87K oct 22 09:42 syslog.bz2  
-rw-rw-r-- 1 alumno alumno 210K oct 22 09:43 syslog.gz  
alumno@servidor:~$
```

```
alumno@servidor:~$ time bzip2 -dc ~/syslog.bz2 > /dev/null  
  
real    0m0,022s  
user    0m0,020s  
sys     0m0,000s  
alumno@servidor:~$ time gzip -dc ~/syslog.gz > /dev/null  
  
real    0m0,008s  
user    0m0,007s  
sys     0m0,000s
```

8.7

```

alumno@servidor:~$ touch /tmp/archivo1.txt
alumno@servidor:~$ ln -sf /tmp/archivo1.txt ~/enlace.txt
alumno@servidor:~$ ls -l ~/enlace.txt
lrwxrwxrwx 1 alumno alumno 17 oct 22 09:54 /home/alumno/enlace.txt -> /tmp/archivo1.txt
alumno@servidor:~$ tar -cvhf /tmp/home_deref.tar -C "$HOME"
tar: Cowardly refusing to create an empty archive
Try 'tar --help' or 'tar --usage' for more information.
alumno@servidor:~$ tar -cvhf /tmp/home_deref.tar -C "$HOME" .

```

```

alumno@servidor:~$ tar -cvhf /tmp/home_deref.tar -C "$HOME" .
./
./october_files.txt
./ssh/
./ssh/authorized_keys
./dos palabras.txt
./local/
./local/share/
./local/share/nano/
./mi_bin/
./b.log
./~m
./ejercicio6.txt
./ejercicio5.zip
./bash_logout
./misarchivos.txt
./mv
./config_files.text
./c.jpg
./test_locate.txt
./sudo_as_admin_successful
./bashrc
./script
./a.txt
./log_backup.tar.gz
./enlace.txt
./syslog
./syslog.bz2
./errors.txt
./var/
./var/log/
./var/log/syslog
./lessht
./syslog.gz
./prueba/
./prueba/prueba1.txt.bak
./cache/
./cache/motd.legal-displayed
./ejercicio6_1.txt
./prueba1.txt
./profile
./script1.sh
alumno@servidor:~$ tar -tvf /tmp/home_deref.tar | grep enlace.txt || true
-rw-rw-r-- alumno/alumno      0 2025-10-22 09:53 ./enlace.txt
alumno@servidor:~$

```

8.8

```

ruta="$1"

if [ -f "$ruta" ]; then
    echo "'$ruta' es un archivo regular"
elif [ -d "$ruta" ]; then
    echo "'$ruta' es un directorio"
else
    echo "'$ruta' no existe"
fi

```

```

alumno@servidor:~$ ./script1.sh /etc/passwd
'/etc/passwd' es un archivo regular
alumno@servidor:~$ ./script1.sh /etc
'/etc' es un directorio
alumno@servidor:~$ ./script1.sh /ruta
'/ruta' no existe
alumno@servidor:~$

```

8.9

```

GNU nano 7.2
#!/bin/bash

mkdir /test_dir 2>&1

if [ $? -eq 0 ]; then
    echo "Se ha creado el directorio con exito."
else
    echo "Error: no se pudo crear el directorio, permiso denegado."
fi

```

```

alumno@servidor:~$ ./script1.sh
mkdir: cannot create directory '/test_dir': Permission denied
Error: no se pudo crear el directorio, permiso denegado.
alumno@servidor:~$ sudo ./script1.sh
[sudo] password for alumno:
Se ha creado el directorio con exito.
alumno@servidor:~$ _

```


8.10

```
GNU nano 7.2
#!/bin/bash

for arg in "$@"; do
if [[ $arg =~ ^[0-9]+$ ]];then
    if [ "$arg" -gt 10 ]; then
        echo "$arg"
    fi
fi
done_
```

```
alumno@servidor:~$ ./script1.sh 6 9 17 34 0 61
17
34
61
alumno@servidor:~$
```