

**UNIVERSIDAD DE LAS FUERZAS
ARMADAS-ESPE SEDE SANTO DOMINGO**

**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN - DCCO-SS
CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN**

PERIODO : 202450

ASIGNATURA : Sistemas Operativos

TEMA : Tarea 2:Grupal “Guia”

NOMBRES : Juan David Jiménez Romero

NIVEL-PARALELO : 15310

DOCENTE : Ing. Javier Cevallos. MSc.

FECHA DE ENTREGA : 10/07/2024

SANTO DOMINGO - ECUADOR

2024



```
Activities Terminal Jul 10 21:09 vboxuser@10: ~
root@10:~/scripts# nl archivo.sh
1  #!/bin/sh
2  echo "cual es tu nombre"
3  read nombre
4  echo "Que estas haciendo, $nombre"
5  read hace
6  echo "Genial, yo tambien estoy $hace !"

root@10:~/scripts#
```

El comando “nl + (nombre archivo)” este muestra el contenido de un archivo de texto ya sea .txt o .sh o entre otros; en Debian sin necesidad de poner el comando “nl -b (nombre archivo)” para que muestre las líneas con numeración.



```
Activities Terminal Jul 10 21:13 vboxuser@10: ~
root@10:~/scripts# cat -n archivo.sh
1  #!/bin/sh
2  echo "cual es tu nombre"
3  read nombre
4  echo "Que estas haciendo, $nombre"
5  read hace
6  echo "Genial, yo tambien estoy $hace !"
7
8
9
10
11
12
13
14
15
16
root@10:~/scripts#
```

El comando “cat -n (nombre archivo)” este sirve para lo mismo que el comando “nl” pero esta muestra enumerado todas las líneas incluso las que están en blanco como se puede apreciar en la imagen.

```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 10 21:17
vboxuser@10: ~

root@10:~/scripts# nl -ba archivo.sh
1  #!/bin/sh
2  echo "cual es tu nombre"
3  read nombre
4  echo "Que estas haciendo, $nombre"
5  read hace
6  echo "Genial, yo tambien estoy $hace !"
7
8
9
10
11
12
13
14
15
16
root@10:~/scripts#
```

El comando “nl -ba (nombre archivo)” este sirve para realizar lo mismo que el comando “cat -n” porque así mismo muestra con enumeración las líneas en blanco.

```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 10 21:19
vboxuser@10: ~

root@10:~/scripts# od archivo.sh
0000000 020443 061057 067151 071457 005150 061545 067550 021040
0000020 072543 066141 062440 020163 072564 067040 066557 071142
0000040 021145 071012 060545 020144 067556 061155 062562 062412
0000060 064143 020157 050442 062565 062440 072163 071541 064040
0000080 061541 062551 062156 026157 022040 067556 061155 062562
0000100 005042 062562 062141 064040 061541 005145 061545 067550
0000120 021040 062507 064556 066141 020054 067571 072040 066541
0000140 064542 067145 062440 072163 074557 022040 060550 062543
0000160 020440 005042 005012 005012 005012 005012
0000200
0000216
root@10:~/scripts#
```

El comando “od (nombre archivo)” este muestra los archivos en formato octal el cual es un formato que se se almacenan los datos como puede ser el hexadecimal o el binario y este solo va del 0 al 7 siendo que tienen un total de 8 tipos de datos.

```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 10 21:39
vboxuser@10: ~

root@10:~/scripts# od -c archivo.sh
0000000 # ! / b i n / s h \n e c h o "
0000020 c u a l   e s   t u   n o m b r e
0000040 e " \n r e a d   n o m b r e \n e
0000060 c h o "   Q u e   e s t a s   h a
0000100 a c i e n d o ,   $ n o m b r e
0000120 " \n r e a d   h a c e \n e c h o
0000140 "   G e n i a l ,   y o   t a m
0000160 b i e n   e s t o y   $ h a c e
0000200 ! " \n \n \n \n \n \n \n \n \n \n
0000216
root@10:~/scripts#
```

El comando “od -c (nombre archivo)” esta muestra asi mismo en octal los datos, pero esta vez muestra los caracteres que lo componen asi es fila donde se ve los caracteres que forman el texto del archivo.

```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 10 21:44
vboxuser@10: ~
root@10:~/scripts# od -cc archivo.sh
0000000 # / b i n / s h \n e c h o "
# / b i n / s h \n e c h o "
0000020 c u a l e s t u n o m b r
c u a l e s t u n o m b r
0000040 e " \n r e a d n o m b r e \n e
e " \n r e a d n o m b r e \n e
0000060 c h o " Q u e e s t a s h
c h o " Q u e e s t a s h
0000100 a c i e n d o , $ n o m b r e
a c i e n d o , $ n o m b r e
0000120 " \n r e a d h a c e \n e c h o
" \n r e a d h a c e \n e c h o
0000140 " G e n i a l , y o t a m
" G e n i a l , y o t a m
0000160 b i e n e s t o y $ h a c e
b i e n e s t o y $ h a c e
0000200 ! " \n \n \n \n \n \n \n \n \n \n
! " \n \n \n \n \n \n \n \n \n \n
0000216
root@10:~/scripts#
```

El comando “od -cc (nombre archivo)” esta muestra así mismo en formato octal los caracteres del archivo solo que esta vez los ubica uno encima del otro.

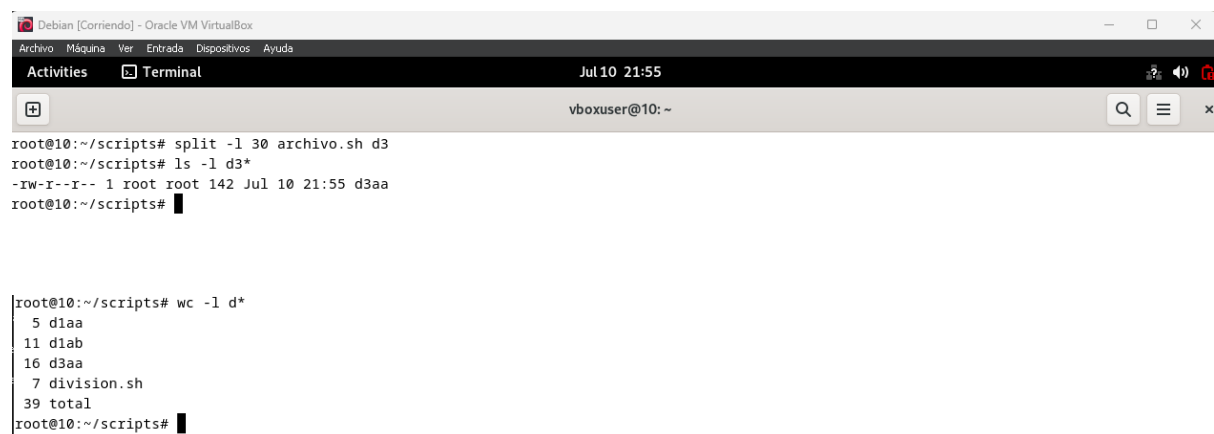
```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 10 21:48
vboxuser@10: ~
cadenanombre.sh division.sh fecha.sh funcion.sh leer.sh operaciones.sh suma viruschiste.sh
root@10:~/scripts# split -b 100 archivo.sh
root@10:~/scripts# ls
archivo.sh cadena.sh fecha1.sh funcionerror.sh leer1.sh multiplicacion.sh resta.sh suma.sh xaa
cadenanombre.sh division.sh fecha.sh funcion.sh leer.sh operaciones.sh suma viruschiste.sh xab
root@10:~/scripts# ls -l
total 72
-rw-r--r-- 1 root root 142 Jul 4 17:58 archivo.sh
-rw-r--r-- 1 root root 126 Jul 4 19:06 cadenanombre.sh
-rw-r--r-- 1 root root 50 Jul 4 19:03 cadena.sh
-rw-r--r-- 1 root root 153 Jul 4 19:32 division.sh
-rw-r--r-- 1 root root 115 Jul 4 18:39 fecha1.sh
-rw-r--r-- 1 root root 84 Jul 4 18:34 fecha.sh
-rw-r--r-- 1 root root 109 Jul 4 18:22 funcionerror.sh
-rw-r--r-- 1 root root 103 Jul 4 18:17 funcion.sh
-rw-r--r-- 1 root root 156 Jul 4 18:59 leer1.sh
-rw-r--r-- 1 root root 113 Jul 4 18:27 leer.sh
-rw-r--r-- 1 root root 159 Jul 4 19:28 multiplicacion.sh
-rw-r--r-- 1 root root 216 Jul 4 19:15 operaciones.sh
-rw-r--r-- 1 root root 153 Jul 4 19:24 resta.sh
-rw-r--r-- 1 root root 157 Jul 4 19:19 suma
-rw-r--r-- 1 root root 100 Jul 4 19:10 suma.sh
-rw-r--r-- 1 root root 108 Jul 4 18:47 viruschiste.sh
-rw-r--r-- 1 root root 100 Jul 10 21:47 xaa
-rw-r--r-- 1 root root 42 Jul 10 21:47 xab
root@10:~/scripts# split -b 100 archivo.sh d1
root@10:~/scripts# ls -l d1*
-rw-r--r-- 1 root root 100 Jul 10 21:47 diaa
-rw-r--r-- 1 root root 42 Jul 10 21:47 diab
root@10:~/scripts#
```

El comando “split -b (número de divisiones posibles) (nombre archivo) (iniciador de la división del archivo seleccionado)” este comando lo que realiza es dividir como dice el nombre traducido del ingles que separa por partes el archivo.



```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Activities  Terminal
Jul 10 21:53
vboxuser@10: ~
root@10:~/scripts# split -C 100 archivo.sh
root@10:~/scripts# ls -l d1*
-rw-r--r-- 1 root root 100 Jul 10 21:47 d1aa
-rw-r--r-- 1 root root 42 Jul 10 21:47 d1ab
root@10:~/scripts#
```

El comando “split -C (número de divisiones) (nombre archivo) (indicador)” este sirve para dividir un archivo, pero con la condición de no dividir las líneas de este.



```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Activities  Terminal
Jul 10 21:55
vboxuser@10: ~
root@10:~/scripts# split -l 30 archivo.sh d3
root@10:~/scripts# ls -l d3*
-rw-r--r-- 1 root root 142 Jul 10 21:55 d3aa
root@10:~/scripts#

root@10:~/scripts# wc -l d*
 5 d1aa
11 d1ab
16 d3aa
 7 division.sh
39 total
root@10:~/scripts#
```

Con el comando “split -l (número de divisiones) (nombre archivo) (indicador)” este sirve para dividir un archivo, pero únicamente hasta llegar a las líneas indicadas por el usuario.

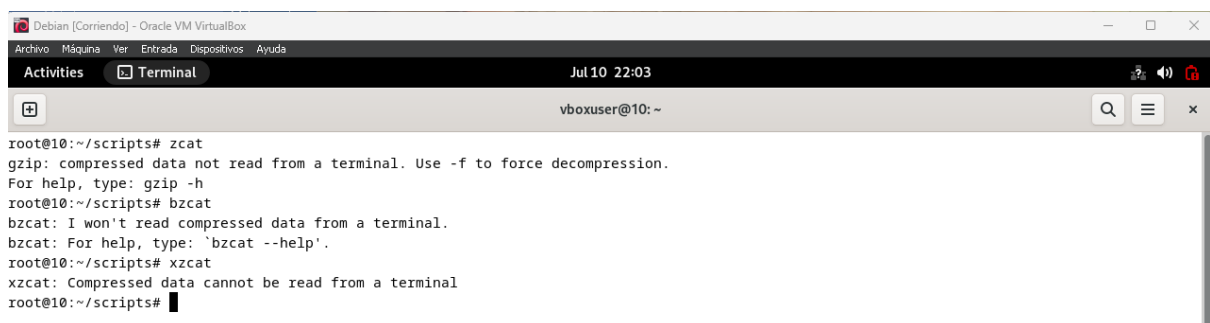


```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 10 22:00
vboxuser@10: ~

root@10:~/scripts# paste archivo.sh resta.sh
#!/bin/sh          #!/bin/bash
echo "cual es tu nombre"      echo "ingrese el var1:"
read nombre      read var1
echo "Que estas haciendo, $nombre"      echo "ingrese el var2:"
read hace      read var2
echo "Genial, yo tambien estoy $hace !" result=$(( $var1 - $var2 ))
      echo "La resta del var1 y var2 es:" $result

root@10:~/scripts#
```

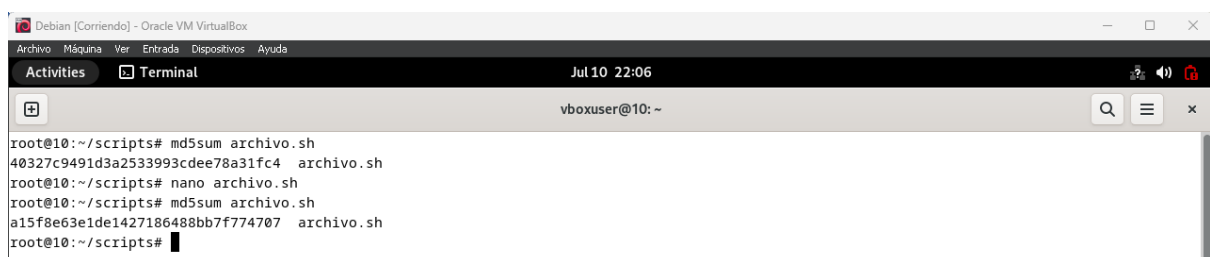
El comando “paste (nombre archivo 1) (nombre archivo 2)” este lo que realiza es funcionar el contenido de los 2 archivos el cual se muestra en pantalla dependiendo del orden que se ingrese de la mezcla.



```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 10 22:03
vboxuser@10: ~

root@10:~/scripts# zcat
gzip: compressed data not read from a terminal. Use -f to force decompression.
For help, type: gzip -h
root@10:~/scripts# bzip2
bzip2: I won't read compressed data from a terminal.
bzip2: For help, type: 'bzip2 --help'.
root@10:~/scripts# xzcat
xzcat: Compressed data cannot be read from a terminal
root@10:~/scripts#
```

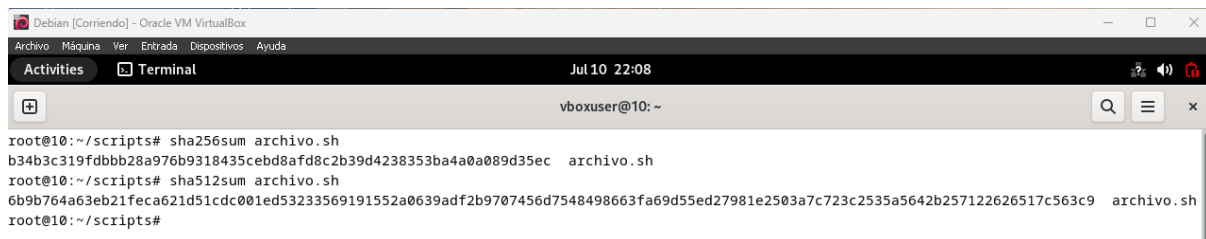
Con los comandos “zcat (nombre archivo)” / “bzip2 (nombre archivo)” / “xzcat (nombre archivo)” este funciona para mostrar el contenido de archivos comprimidos en formato Gzip, Bzip2 y Xz este haciendo que sea posible ver el contenido de los archivos sin descomprimirlos.



```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 10 22:06
vboxuser@10: ~

root@10:~/scripts# md5sum archivo.sh
40327c9491d3a2533993cdee78a31fc4  archivo.sh
root@10:~/scripts# nano archivo.sh
root@10:~/scripts# md5sum archivo.sh
a15f8e63e1de1427186488bb7f774707  archivo.sh
root@10:~/scripts#
```

Con el comando “md5sum (nombre archivo)” este funciona para darnos como así decirlo una marca del archivo que es única dependiendo del archivo, si este archivo llega a ser modificado esta marca digital se va a cambiar.

A screenshot of a terminal window titled "Debian [Corriendo] - Oracle VM VirtualBox". The terminal shows the user 'vboxuser@10: ~' at the prompt. The user enters 'sha256sum archivo.sh' and receives a long hexadecimal string. Then, the user enters 'sha512sum archivo.sh' and receives an even longer hexadecimal string. The terminal interface includes a menu bar with 'Archivo', 'Máquina', 'Ver', 'Entrada', 'Dispositivos', and 'Ayuda'. The status bar shows 'Jul 10 22:08' and system icons for volume and network.

```
root@10:~/scripts# sha256sum archivo.sh
b34b3c319fdbbb28a976b9318435cebd8afd8c2b39d4238353ba4a0a089d35ec  archivo.sh
root@10:~/scripts# sha512sum archivo.sh
6b9b764a63eb21feca621d51cdc001ed53233569191552a0639adf2b9707456d7548498663fa69d55ed27981e2503a7c723c2535a5642b257122626517c563c9  archivo.sh
root@10:~/scripts#
```

Los comandos “sha256sum (nombre archivo)” y “sha512sum (nombre archivo)” este realiza la misma función de darnos una marca digital de archivo solamente que dependiendo de los números que serían el 256 y 512 va a mostrar más caracteres de la marca digital.

A screenshot of a terminal window titled "Debian [Corriendo] - Oracle VM VirtualBox". The terminal shows the user 'vboxuser@10: ~' at the prompt. The user enters 'mkdir ejemplo', then 'cd ejemplo/'. They then run 'ls -l' and 'ls -lah' to see the directory's contents and permissions. Finally, they run 'mkdir dir1 dir2 dir3' to create subdirectories. The terminal interface includes a menu bar with 'Archivo', 'Máquina', 'Ver', 'Entrada', 'Dispositivos', and 'Ayuda'. The status bar shows 'Jul 10 22:12' and system icons for volume and network.

```
root@10:~# mkdir ejemplo
root@10:~# cd ejemplo/
root@10:~/ejemplo# ls -l
total 0
root@10:~/ejemplo# ls -lah
total 8.0K
drwxr-xr-x  2 root root 4.0K Jul 10 22:11 .
drwx----- 10 root root 4.0K Jul 10 22:11 ..
root@10:~/ejemplo# mkdir dir1 dir2 dir3
root@10:~/ejemplo# ls -l
total 12
drwxr-xr-x  2 root root 4096 Jul 10 22:12 dir1
drwxr-xr-x  2 root root 4096 Jul 10 22:12 dir2
drwxr-xr-x  2 root root 4096 Jul 10 22:12 dir3
root@10:~/ejemplo#
```

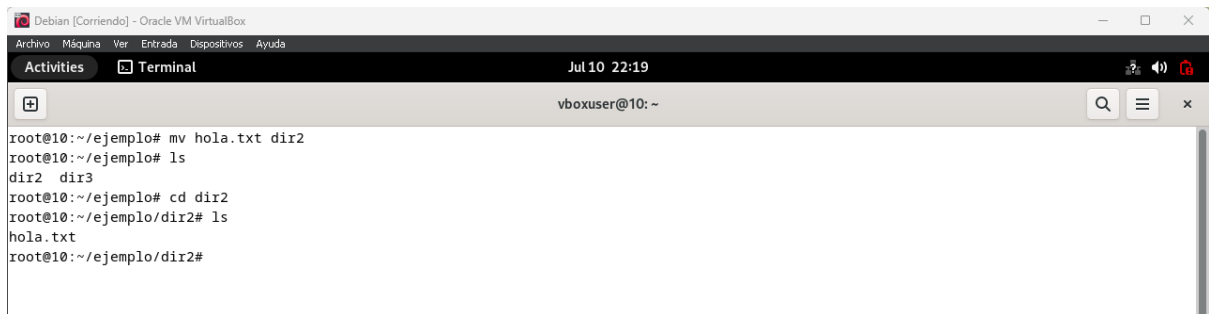
El comando “mkdir (nombre directorio)” este sirve para crear directorios vacíos donde se pueden ir añadiendo archivos o más directorios.

A screenshot of a terminal window titled "Debian [Corriendo] - Oracle VM VirtualBox". The terminal shows the user 'vboxuser@10: ~' at the prompt. The user enters 'ls' to see the current directory. Then, they enter 'rm -r dir1' to remove the directory. They run 'ls' again to confirm. Next, they enter 'touch hola.txt' to create a file. Finally, they run 'ls' to see the new file. The terminal interface includes a menu bar with 'Archivo', 'Máquina', 'Ver', 'Entrada', 'Dispositivos', and 'Ayuda'. The status bar shows 'Jul 10 22:14' and system icons for volume and network.

```
root@10:~/ejemplo# ls
dir1 dir2 dir3
root@10:~/ejemplo# rm -r dir1
root@10:~/ejemplo# ls
dir2 dir3
root@10:~/ejemplo# touch hola.txt
root@10:~/ejemplo# ls
dir2 dir3 hola.txt
root@10:~/ejemplo#
```

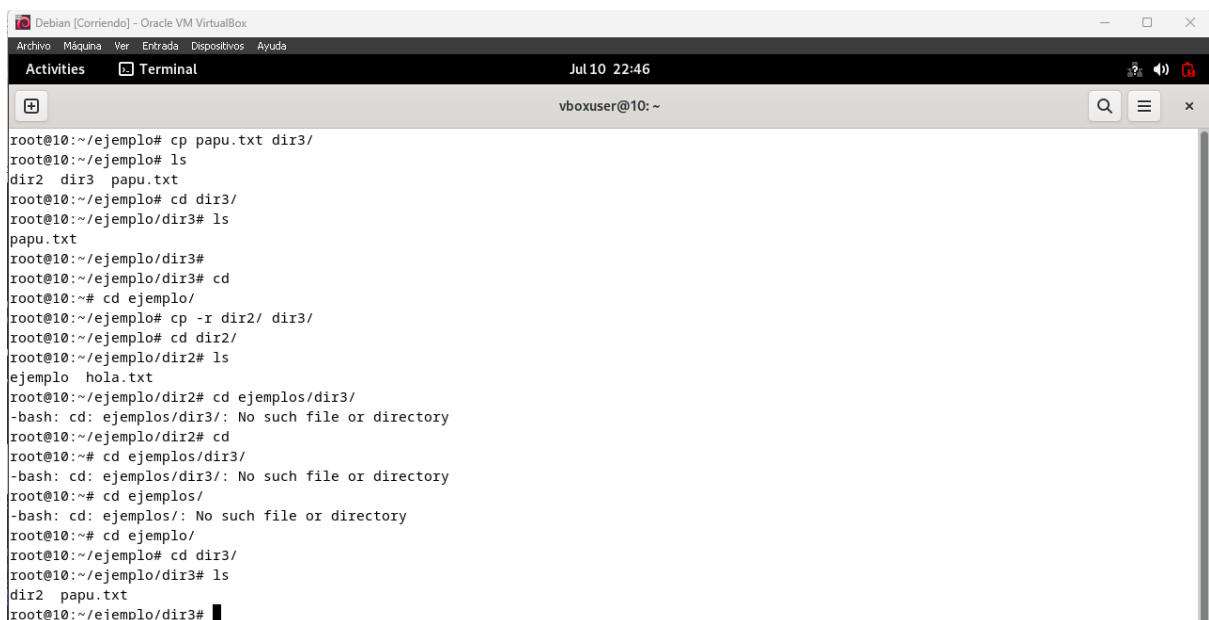
Con el comando “rm -r (nombre directorio)” este elimina un directorio que se encuentre vacío en este caso el directorio dir1 que recién fue creado.

Con el comando “touch (nombre archivo +. tipo de archivo)” este realiza la función de crear archivos de cualquier tipo mientras tenga al final el tipo de archivo que sea.



```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 10 22:19
vboxuser@10: ~
root@10:~/ejemplo# mv hola.txt dir2
root@10:~/ejemplo# ls
dir2 dir3
root@10:~/ejemplo# cd dir2
root@10:~/ejemplo/dir2# ls
hola.txt
root@10:~/ejemplo/dir2#
```

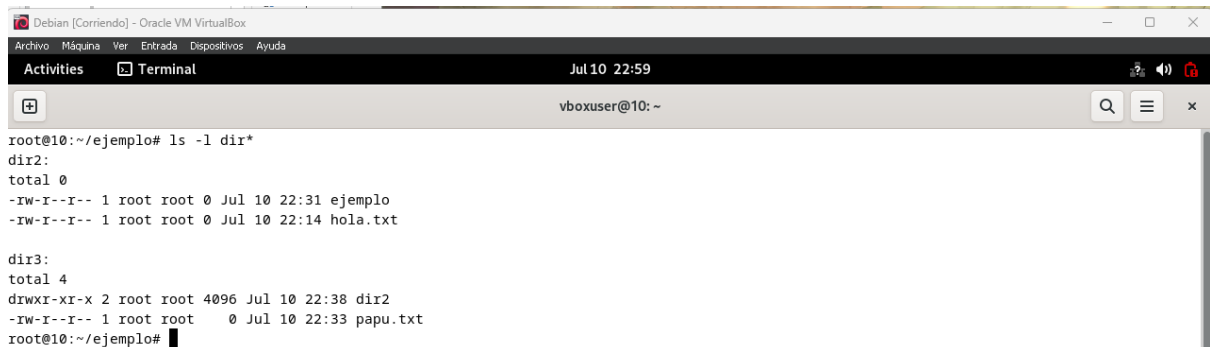
El comando “mv (nombre archivo) (dirección directorio donde se desea mover)” este comando sirve para mover archivos de un directorio a otro siendo mv sus siglas debido a la palabra en ingles move que significa mover.



```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 10 22:46
vboxuser@10: ~
root@10:~/ejemplo# cp papu.txt dir3/
root@10:~/ejemplo# ls
dir2 dir3 papu.txt
root@10:~/ejemplo# cd dir3/
root@10:~/ejemplo/dir3# ls
papu.txt
root@10:~/ejemplo/dir3#
root@10:~/ejemplo/dir3# cd
root@10:~/ejemplo# cd ejemplo/
root@10:~/ejemplo# cp -r dir2/ dir3/
root@10:~/ejemplo# cd dir2/
root@10:~/ejemplo/dir2# ls
ejemplo hola.txt
root@10:~/ejemplo/dir2# cd ejemplos/dir3/
-bash: cd: ejemplos/dir3/: No such file or directory
root@10:~/ejemplo/dir2# cd
root@10:~/ejemplo# cd ejemplos/dir3/
-bash: cd: ejemplos/dir3/: No such file or directory
root@10:~/ejemplo# cd ejemplos/
-bash: cd: ejemplos/: No such file or directory
root@10:~/ejemplo# cd ejemplo/
root@10:~/ejemplo# cd dir3/
root@10:~/ejemplo/dir3# ls
dir2 papu.txt
root@10:~/ejemplo/dir3#
```

El comando “cp (nombre de archivo) (direccion que se desea copiar)” esta copia un archivo y lo ubica en la dirección indicada.

El comando “cp -r (dirección 1) (dirección 2)” esta copia un directorio y lo pega dentro de otro directorio.



```
root@10:~/ejemplo# ls -l dir*
dir2:
total 0
-rw-r--r-- 1 root root 0 Jul 10 22:31 ejemplo
-rw-r--r-- 1 root root 0 Jul 10 22:14 hola.txt

dir3:
total 4
drwxr-xr-x 2 root root 4096 Jul 10 22:38 dir2
-rw-r--r-- 1 root root 0 Jul 10 22:33 papu.txt
root@10:~/ejemplo#
```

El uso de los * en el comando ls realiza la practica función de buscar únicamente los archivos que cumplan la condición de búsqueda que sería la palabra que se encuentre antes del *.

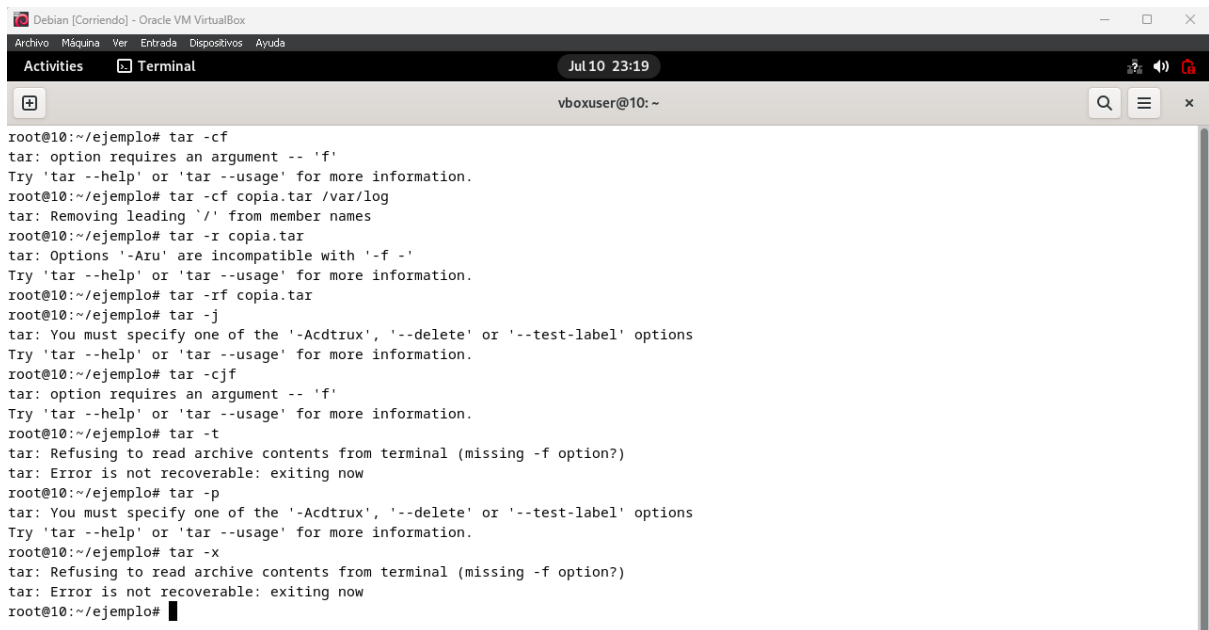
Si realizamos la misma función únicamente con el * sin una palabra atrás mostrara todo incluso lo que este dentro del contenido de los directorios que estén dentro.

En caso de poner ** pero con una palabra dentro ejemplo: *.txt* este va a mostrar únicamente los archivos que tengan esa extensión.



```
root@10:~/ejemplo# ls -l sopa
ls: cannot access 'sopa': No such file or directory
root@10:~/ejemplo# ls -l sopa?.*
-rw-r--r-- 1 root root 0 Jul 10 23:09 sopa1.mp3
-rw-r--r-- 1 root root 0 Jul 10 23:10 sopa1.txt
-rw-r--r-- 1 root root 0 Jul 10 23:10 sopa2.mp3
-rw-r--r-- 1 root root 0 Jul 10 23:10 sopa2.txt
root@10:~/ejemplo# ls
dir2 dir3 papu.txt sopa1.mp3 sopa1.txt sopa2.mp3 sopa2.txt sopa.mp3 sopa.txt
root@10:~/ejemplo#
```

El uso de las ¿? Al momento de realizar búsquedas sirve para al momento de tener archivos con nombre exacto pero con distinta enumeración muestre en todos los archivos que se tengan ese nombre + la numeración el añadir un .* realiza la función de buscar también todos los archivos con distinto formato.

A screenshot of a terminal window titled "Debian [Corriendo] - Oracle VM VirtualBox". The terminal shows a series of tar commands and their outputs. The user is root@10:~/ejemplo. The commands and outputs are: 1. tar -cf: "tar: option requires an argument -- 'f'. Try 'tar --help' or 'tar --usage' for more information." 2. tar -cf copia.tar /var/log: "tar: Removing leading '/' from member names" 3. tar -r copia.tar: "tar: Options '-Aru' are incompatible with '-f -'. Try 'tar --help' or 'tar --usage' for more information." 4. tar -rf copia.tar: "tar: You must specify one of the '-Acdrux', '--delete' or '--test-label' options. Try 'tar --help' or 'tar --usage' for more information." 5. tar -j: "tar: option requires an argument -- 'f'. Try 'tar --help' or 'tar --usage' for more information." 6. tar -t: "tar: Refusing to read archive contents from terminal (missing -f option?). tar: Error is not recoverable: exiting now" 7. tar -p: "tar: You must specify one of the '-Acdrux', '--delete' or '--test-label' options. Try 'tar --help' or 'tar --usage' for more information." 8. tar -x: "tar: Refusing to read archive contents from terminal (missing -f option?). tar: Error is not recoverable: exiting now". The terminal ends with root@10:~/ejemplo#.

```
root@10:~/ejemplo# tar -cf
tar: option requires an argument -- 'f'
Try 'tar --help' or 'tar --usage' for more information.
root@10:~/ejemplo# tar -cf copia.tar /var/log
tar: Removing leading '/' from member names
root@10:~/ejemplo# tar -r copia.tar
tar: Options '-Aru' are incompatible with '-f -'
Try 'tar --help' or 'tar --usage' for more information.
root@10:~/ejemplo# tar -rf copia.tar
tar: You must specify one of the '-Acdrux', '--delete' or '--test-label' options
Try 'tar --help' or 'tar --usage' for more information.
root@10:~/ejemplo# tar -j
tar: option requires an argument -- 'f'
Try 'tar --help' or 'tar --usage' for more information.
root@10:~/ejemplo# tar -t
tar: Refusing to read archive contents from terminal (missing -f option?)
tar: Error is not recoverable: exiting now
root@10:~/ejemplo# tar -p
tar: You must specify one of the '-Acdrux', '--delete' or '--test-label' options
Try 'tar --help' or 'tar --usage' for more information.
root@10:~/ejemplo# tar -x
tar: Refusing to read archive contents from terminal (missing -f option?)
tar: Error is not recoverable: exiting now
root@10:~/ejemplo#
```

El comando “tar” es mayoritariamente usado para la extraccion de archivos comprimidos de varios tipos.

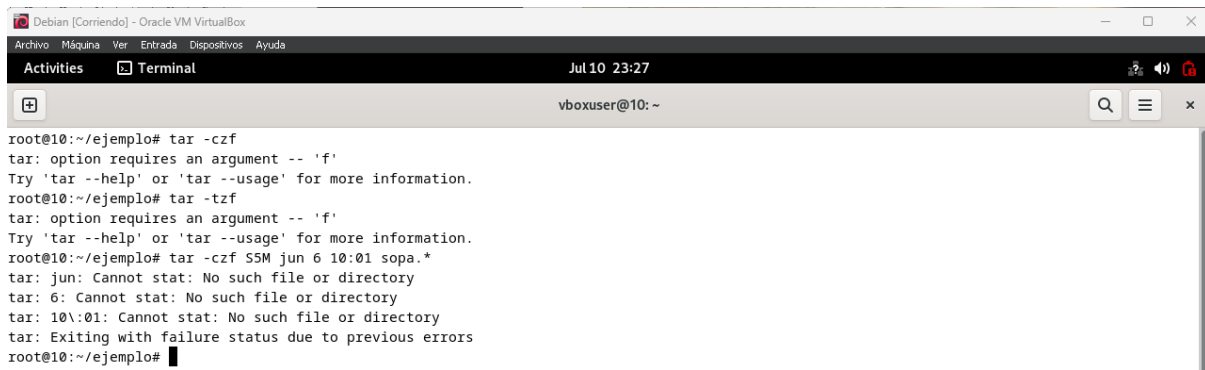
El comando “tar -cf” este es el metodo basico para el uso de tar el -c es de compacta y el -f es para extraer

El comando “tar -cf copia.tar/var/log” es para realizar una referencia que guarde las rutas del archivo, donde también se puede usar el -p en vez de realizar el uso de comando - cf copia.tar/var/log.

El comando “tar -r” este comando sirve para añadir archivos dentro de un archivo que se encuentre compactado.

El comando “tar -t” muestra la informacion del contenido que se encuentre dentro de un archivo comprimido.

El comando “tar -j” puede comprimir o descomprimir archivos de tipo bzip2.


A screenshot of a terminal window titled "Debian [Corriendo] - Oracle VM VirtualBox". The window has a menu bar with "Archivo", "Máquina", "Ver", "Entrada", "Dispositivos", and "Ayuda". Below the menu bar is a status bar showing "Jul 10 23:27" and system icons. The terminal itself shows a prompt "root@10:~/ejemplo#" and the following commands and output:

```
root@10:~/ejemplo# tar -czf
tar: option requires an argument -- 'f'
Try 'tar --help' or 'tar --usage' for more information.
root@10:~/ejemplo# tar -tzf
tar: option requires an argument -- 'f'
Try 'tar --help' or 'tar --usage' for more information.
root@10:~/ejemplo# tar -czf S5M jun 6 10:01 sopa.*
tar: jun: Cannot stat: No such file or directory
tar: 6: Cannot stat: No such file or directory
tar: 10\01: Cannot stat: No such file or directory
tar: Exiting with failure status due to previous errors
root@10:~/ejemplo#
```

El comando “tar -czf” crea una copia de un archivo de que este comprimido mientras que el comando “tar -tzf” este nos muestra el contenido que tiene la copia realizada.

El comando “Tar -czf S(date +%F)-I.tar.gz películas/ -N 2024-06-01” este comando realiza la función de guardar información para una fecha específica.

El comando “tar -p” establece las rutas absolutas para el archivo mientras el comando “tar -x” las expande.

A screenshot of a terminal window titled "Debian [Corriendo] - Oracle VM VirtualBox". The window has a menu bar with "Archivo", "Máquina", "Ver", "Entrada", "Dispositivos", and "Ayuda". Below the menu bar is a status bar showing "Jul 10 23:55" and system icons. The terminal itself shows a prompt "root@10:~/ejemplo#" and the following commands and output:

```
root@10:~/ejemplo# ls *.gz | cpio -o > ficheros.cpio
ls: cannot access '*.gz': No such file or directory
1 block
root@10:~/ejemplo# cpio -i < ficheros.cpio
1 block
root@10:~/ejemplo# find sopa | cpio -p FICHEROS
find: 'sopa': No such file or directory
0 blocks
root@10:~/ejemplo#
```

El comando “cpio” copia los ficheros desde un único archivo mientras que el comando “cpio -o” este crea un archivo que contenga el contenido que le especifiquemos que tenga.

El comando “cpio -i” este extrae en el directorio el contenido de un fichero que se desee.

El comando “cpio -p” este crea una estructura del directorio a otra ubicación.



```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 11 00:07
vboxuser@10: ~
root@10:~/ejemplo# dd if=~/ejemplo/dir2 of=sopa.mp3 bs=1M count=100
dd: error reading '/root/ejemplo/dir2': Is a directory
0+0 records in
0+0 records out
0 bytes copied, 0.000377522 s, 0.0 kB/s
root@10:~/ejemplo# ls -lh sopa.mp3
-rw-r--r-- 1 root root 0 Jul 11 00:06 sopa.mp3
root@10:~/ejemplo# file sopa.mp3
sopa.mp3: empty
root@10:~/ejemplo#
```

El comando “dd” copia informacion de un archivo de bajo nivel, este se trabaja con funciones if que sirve para marcar el origen del archivo y el of que sirve para marcar el destino es este, bs para el tamaño del bloque que se quiere copiar.

El comando “ls -lh (nombre fichero)” sirve para comprobar la integridad del archivo copia de bajo nivel

El comando “file” se encarga de mostrar los tipos de datos que contiene el fichero que se desea inspeccionar.



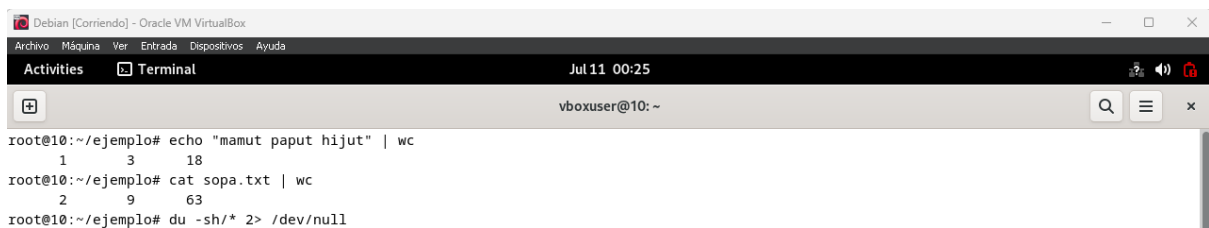
```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 11 00:14
vboxuser@10: ~
root@10:~/ejemplo# gzip copia
gzip: copia: No such file or directory
root@10:~/ejemplo# gzip -k copia
gzip: copia: No such file or directory
root@10:~/ejemplo# gzip -d copia
gzip: copia.gz: No such file or directory
root@10:~/ejemplo#
```

Como se mencionó el uso de los comandos “gzip – gunzip – xz – bzip2” tienen la funcion de extraer archivos de ese tipo de archivos, pero en esta ocacion se mostrará los comandos para copiar los archivos comprimidos o eliminar al momento de extraer el archivo deseado siendo que todo tienen los mismos comandos, este siendo este “-k” el que conserva el archivo original al momento de extraer y el comando “-d” para descomprimir con el nombre original del archivo.



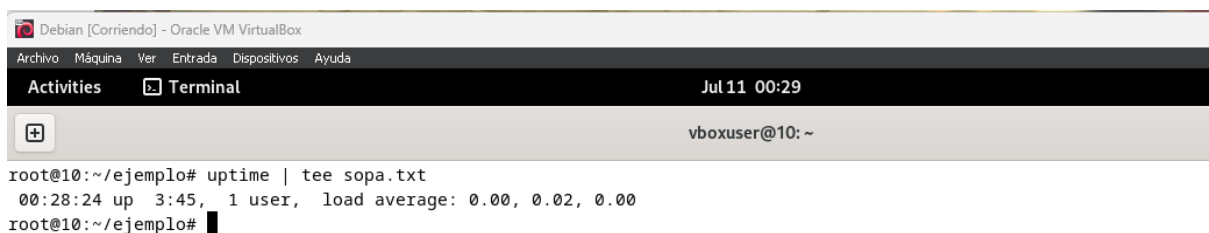
```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 11 00:21
vboxuser@10: ~
root@10:~/ejemplo# echo "Sabes porque se extinguieron los mamuts" > sopa.txt
root@10:~/ejemplo# cat sopa.txt
Sabes porque se extinguieron los mamuts
root@10:~/ejemplo# echo "Porque faltaban Paputs" >> sopa.txt
root@10:~/ejemplo# cat sopa.txt
Sabes porque se extinguieron los mamuts
Porque faltaban Paputs
root@10:~/ejemplo#
```

El comando “echo “mensaje” > (nombre archivo)” este funciona para ingresar un texto dentro de un archivo, si se desea agregar más contenido se realiza la misma función solo añadiéndole un > después del > sin espacio.



```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 11 00:25
vboxuser@10: ~
root@10:~/ejemplo# echo "mamut paput hijut" | wc
1      3      18
root@10:~/ejemplo# cat sopa.txt | wc
2      9      63
root@10:~/ejemplo# du -sh/* 2> /dev/null
```

El comando “echo “mensaje” | wc” este muestra el contenido en fila, palabras y letras siendo ese el orden de salida que se muestra, este tambien puede usarse para archivos que tengan contenido dentro y funciona de la misma funcion solo que con el uso de “cat (nombre archivo) | wc” mientras el comando “du” muestra el peso de cada tuberia en la carpeta actual.



```
Debian [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Activities Terminal Jul 11 00:29
vboxuser@10: ~
root@10:~/ejemplo# uptime | tee sopa.txt
00:28:24 up 3:45, 1 user, load average: 0.00, 0.02, 0.00
root@10:~/ejemplo#
```

El comando “tee” es un tipo de comando que se encarga de realizar la visualización de la información de un archivo, este mostrándonos el tiempo que se realiza la función al archivo el número de usuarios que lo usan y la carga en el sistema de este.

```
Debian [Corriendo] - Oracle VM VirtualBox
Actividades Terminal
Jul 11 00:35
vboxuser@10: ~

root@10:~/ejemplo# echo sopa | xargs touch
root@10:~/ejemplo# ls -l
total 84732
-rw-r--r-- 1 root root 86743040 Jul 10 23:18 copia.tar
drwxr-xr-x 2 root root 4096 Jul 10 22:31 dir2
drwxr-xr-x 3 root root 4096 Jul 10 22:38 dir3
-rw-r--r-- 1 root root 512 Jul 10 23:53 ficheros.cpio
-rw-r--r-- 1 root root 0 Jul 10 22:32 papu.txt
-rw-r--r-- 1 root root 130 Jul 10 23:26 S5M
-rw-r--r-- 1 root root 0 Jul 11 00:33 sopa
-rw-r--r-- 1 root root 0 Jul 10 23:09 sopa1.mp3
-rw-r--r-- 1 root root 0 Jul 10 23:10 sopa1.txt
-rw-r--r-- 1 root root 0 Jul 10 23:10 sopa2.mp3
-rw-r--r-- 1 root root 0 Jul 10 23:10 sopa2.txt
-rw-r--r-- 1 root root 0 Jul 11 00:06 sopa.mp3
-rw-r--r-- 1 root root 61 Jul 11 00:28 sopa.txt
root@10:~/ejemplo# tail
^C
root@10:~/ejemplo# tail -n3 /etc/passwd
Debian-gdm:x:113:122:Gnome Display Manager:/var/lib/gdm3:/bin/false
vboxuser:x:1000:1000:vboxuser,,,:/home/vboxuser:/bin/bash
papu:x:1001:1001:Papu,6942,962745941,962745941,hola:/home/papu:/bin/bash
root@10:~/ejemplo# tail -n3 /etc/passwd | cut -d":" -f1
Debian-gdm
vboxuser
papu
root@10:~/ejemplo#
```

El comando “xargs” sirve para la ejecución de cualesquiera parámetros que esté usando la información de una tubería en este caso revisamos las tuberías que existen en los archivos “sopa” creados anteriormente.

Con el comando “tail” este sirve para administrar usuarios donde con el comando “tail -n3 /etc/passwd” sirve para ver la ubicación donde se encuentra la dirección de la contraseña del usuario.

Este comando también sirve para la creación de los directorios con el comando mkdir siendo una manera más rápida de crear varios directorios.