

TERMINAL Y LINEA DE COMANDOS

TERMINAL: ES UNA INTERFAZ GRAFICA QUE NOS MUESTRA EL PROMT. ESTE ALOJA UNA LINEA

SHELL O LINEA DE COMANDOS : ES UN PROGRAMA QUE TOMA PARAMETROS Y LOS TRANSMITE

POWERSHELL ES DE WINDOWS
BASH SHELL SE UTILIZA EN LINUX
Z SHELL ES LAS MAC DE 1995

COMANDO

- * un programa ejecutable que se compilo y podemos ejecutarlos. Se guardan en las rutas / *user* / bin
- * un comando de utilidad shell que viene por defecto dentro del programas
- * una funcion de shell externa a los programas que vienen incluidos en la shell
- * un alias, un comando creado de forma temporal

COMANDOS

virgulilla	HACE REFERENCIA AL HOME, HOME ES LA CARPETA DONDE SE GUARDAN LOS ARCHIVOS DE US		
ls		lista carpetas (en azul) y archivos (en blanco)	
ls -l	long	lista mostrando cuando fue creado y cuanto pesan en bytes	
ls -lh	lectura huma	lista mostrando cuanto pesan en MB, GB,	
ls -al		lista todos los archivos. Incluso los Ocultos. Son los archivos que tengan un "Pu	
la -a		muestra todos los archivos pero sin listarlos	
ls carpeta		lista la carpeta pero sin movernos de la ubicación	
ls -IS		lista ordenando de mayor tamaño a menor	
ls -lr		lista en sentido opuesto	
cd "nombre_carpeta"		moverse al directorio	
cd O cd + virgulilla		te lleva al home	
Cd ..		retrocede un directorio	
Cd ../..		retrocede tantos directorios como indique	
pwd		muestra la ruta de ficheros donde nos encontramos	/home/lucas/personalProje
Cd "+ ruta"		te leva a la ruta indicada	Cd /home/lucas/personalP
		para indicar una ruta comenzamos con /	

clear O cntrl+L limpiar la pantalla de la terminal // control+L saca de nuestra visión todo lo que e

Control + c CANCELA CUALQUIER PROCEDIMIENTO QUE ESTE CORRIENDO

file “archivo con su extensi muestra una descripción del archivo indicado con Tab autocompleta el n

Opciones -

tecla TAB Realiza un autocompletado del nombre de archivo o directorio que se encuentre

Presionar doble TAB En caso que existan varias opciones de nombre para una palabra nos muestra e

teclas ARRIBA o ABAJO muestra los comandos utilizados

history muestra un historial de comandos utilizados adjuntandoles un número de ejecuc

! + número de comando nos permite ejecutar el comando que vimos con el comando history. Ejemplo ! 7

sudo apt-get install tree instalar el comando tree si no se encuentra instalado

tree muestra los archivos ordenado por niveles

tree -L 2 limita el orden de archivos en los niveles indicados

mkdir “directorio” crear un directorio

mkdir carpeta1 carpeta2 carpeta3 crea carpetas multiples

touch “nombre” crear un archivo

touch file1 file2 file 3 crear archivos multiples

cp cp file1 file1_copia crear una copia del archivo1 en el mismo directorio

cp -r (recursividad) copiamos un directorio y todo su contenido

mv mv file1_copia ,, mover el archivo un directorio atrás

mv file1 file2 renombrar archivo

mv carpeta1 carpeta2 renombrar directorio

cat archivo.txt ver el contenido de dicho archivo

ELIMINAR ARCHIVOS O CARPETAS

rm file eliminar archivo

rm -r carpeta “recursiva” eliminar carpeta que contiene archivos

rm -i file eliminar con una interaccion previa para confirmar

rm -ri carpeta

elimina el directorio generando una interacción previa

COMANDOS DE EXPLORACION DE TEXTO

head	+ nombre del archivo y su extension	muestra las primeras 10 lineas
	head texto.txt -n 15	muestra las primeras 15 lineas
tail		muestra las ultimas
	tail texto.txt -n 15	muestra las ultimas
less	less texto.txt	muestra todo el archivo
	si presionamos / podemos buscar palabras Q es para salir	
xdg-open	texto.txt	abre el archivo en su programa predeterminado
nautilus	nautilus Descargas/	permite abrir la interfaz de la carpeta donde nos encontramos abre la ruta especificada

ALIAS

type {cd}	ver que tipo de comando es
alias l="ls -lh"	definir y crear un alias de uso temporal mientras la consola este ejecutandose
help {comando}	help cd
alias	Visualizar todos los alias definidos en el perfil, ver los creados

WILDCARDS

* Caracteres especiales para definir patrones de búsqueda

Ls *.{extension}	Ls *.txt	busca en base a la extensión
ls {caracteres del nombre}* ls *		

ls {nombre}?
ls {nombre}??
ls {nombre}???

busca los caracteres + un carácter aleatorio mas

Ls [caracteres]*
Ls [[:upper:]]*
Ls -d [[:upper:]]*

busca en base a los archivos que comiencen con los
busca por primer carácter en mayus
busca en el directorio atual

Ls [[:lower:]]*

busca por primer carácter en minus

find -name *.py

REDIRECCIONAR UN STDOUT o STDERR

UN stdin(con un filedescripter designado como 0) es una entrada que puede ser por teclado o direccionada por un archivo

Con el comando > reedirijo la información de salida del comando que veo en pantalla a un archivo .txt (Si existe lo sobre
Con el doble >> puedo añadir información a un archivo existente sin eliminar lo anterior.

Con 2> o 2>> redirecciono los stderr a un archivo.

Este comando se utiliza después del nombre de archivo de destino “ 2>&1 ” Asegura que Redirecciona cualquier sea la i

ls {nombre de directorio} > archivo.txt
Pw {nombre de directorio} >> archivo.txt

CREO EL ARCHIVO Y LE ENVÍO LA INFORMACIÓN
CREA O AÑADE SI ES QUE YA EXISTE LA INFORM

ls {“cualquier carácter que genere error”} 2> error.txt
ls {“cualquier carácter que genere error”} 2>> error.txt

ENVÍA DETALLES DE UN STD ERROR
IDEM, pero añade info.

ls {scrip} > archivo.txt 2>&1
ls {scrip} >> archivo.txt 2>&1

REDIRECCIONA CUALQUIER SALIDA, SEA EL STD
IDEM PERO SIN ELIMINAR EL CONTENIDO

echo “hola”

GENERA UN STDOUTPUT

cat archivo.txt

Este comando hace uso de la función < stdin para mostrar el conte

Cat < archivo.txt
cat archivo.txt error.txt

CONCATENA LA SALIDA DE VARIOS ARCHIVOS. REALIZANDC

ESCRIBIR UN ARCHIVO DESDE LA CONSOLA DE COMANDOS: cat >

Cat > nuevo_archivo.txt Genera una línea donde podemos escribir sobre el archivo indicado. Salimos co

Cat >> nuevo_archivo.txt Agregamos líneas de texto al archivo indicado sin sobrecribir lo que ya existe.

Cat < archivo.txt Muestra en pantala el archivo como un output

PIPE OPERATOR y comando TEE

COMANDO QUE CONVIERTE EL EL STDOUT COMO UN STDINPUT PARA OTRO COMANDO

Con operador “ | ” pipe operator redirecciona la salida de un comando y la arroja como entrada de otro comando

Ls – lh | less

Ls – lh | tee output.txt | less

Ls – lh | sort | tee output.txt | less

comando tee lee la entrada y toma la salida creando un archivo

Ls | tee -a archivo.txt

echo “texto_de_prueba” | tee -a archivo.txt

anulamos que el comando tee sobrescriba el contenido del archivo agregando la opción -a
agrega la linea echo al final del archivo

EJECUTAR COMANDOS CONCATENADOS CON OPERADORES DE CONTROL

SIMBOLOS RESERVADOS POR LA TERMINAL PARA EJECUTAR MÁS DE UN COMANDO DE MANERA CONCATE

ls; mkdir holi; cal	“,”	De forma sincrona se ejecutan uno seguido del otro en el mismo plano
Ls & date & cal	“&”	De forma asincrona se abre una línea de comando (shell) por cada comando.
mkdir test && cd test	“&&”	De forma condicional se ejecuta un comando SI Y SOLO SI el comando anterior es exitoso
cd hhashka touch archivo.txt	“ OR “	Omite errores en la ejecución del primer comando y pasa al siguiente

MANEJAR PERMISOS

Para ver los permisos utilizamos el comando:

ls -l “long”

La cadena

El primer

El permiso

Luego de

Los perm

Estos p

Esta re

> Tipos de archivos

Atributo	Tipo de archivo
-	Un archivo normal.
d	Un directorio.
l	Un link simbólico.
b	Un archivo de bloque especial. Son archivos que manejan la información de los bloques de datos como una USB.

a directorio, | para link simbólico y b para arc

s usuarios

rite, x para execute.

entrarse activado su valor binario es 1.

s hacer su conversión para su correspondien

Diferencia de permisos entre archivos y directorios



Permiso	Archivo	Directorio
r	Permite abrir y leer un archivo.	Permite listar el contenido de un directorio solo si el permiso de ejecución (x) también está activo.
w	Permite escribir en un archivo; sin embargo, este atributo no permite cambiar el nombre de los archivos o eliminarlos. La capacidad de eliminar o cambiar el nombre de los archivos está determinado por los atributos del directorio.	Permite que los archivos dentro de un directorio sean creados, eliminados y renombrados si también se establece el atributo de ejecución.
x	Permite que un archivo sea tratado como un programa y pueda ser ejecutado.	Permite entrar al directorio.

> Tipo de modo

Dueño			Grupo			World		
rwx			r-x			r-x		
1	1	1	1	0	1	1	0	1

> Modo octal

Dueño			Grupo			World		
rwx			r-x			r-x		
1	1	1	1	0	1	1	0	1
7			5			5		

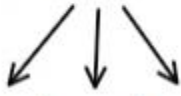
> Modo octal

Octal	Binario	Permisos
0	000	---
1	001	--X
2	010	-W-
3	011	-WX
4	100	r--
5	101	r-X
6	110	rw-
7	111	rwX

drwxrwxrwx

d = Directory
r = Read
w = Write
x = Execute

chmod 777


 rwx | rwx | rwx
 Owner | Group | Others

7	rwX	111
6	rw-	110
5	r-X	101
4	r--	100
3	-WX	011
2	-W-	010
1	--X	001
0	---	000



> Modo simbólico

Símbolo	Significado
u	Solo para el usuario.

g	Solo para el grupo.
o	Solo para otros (es el <i>world</i>) .
a	Aplica para todos.

LA DESIGNACIÓN DE PERMISOS CON CHMOD

LA DESIGNACIÓN DE PERMISOS SE REALIZAN MEDIANTE EL COMANDO `chmod` EL CUAL RECIBE LAS INDICACI

`mkdir sandbox` Creamos una carpeta y un texto a modo de ejemplo

`> mitexto.txt`

```
total 4
-rwxr--r-- 1 lucas lucas 25 may  3 20:36 mitexto.txt
-rw-r--r-- 1 root  root   0 may  3 22:50 rootfile
lucas@lucas-System-Product-Name:~/curso_terminal/sandbox$
```

`cat mitexto.txt` vemos el contenido

`ls -l mitexto.txt` accedemos a los permisos del archivo

`Chmod 755 mitexto.txt` cambiamos los permisos del archivo con el modo octal

`chmod u-r mitexto.txt` El modo mas sencillo es el modo simbolico para quitar los permisos

definimos el usuario (en este caso u)
con el signo de resta – quitamos el permiso (r,w,x)

chmod u+r mitexto.txt con el signo de suma + asignamos el permiso (r,w,x)

chmod u+x,g+r,o+r mitexto.txt

Asignamos permisos a los distintos usuarios de manera simultanea

Sobreescribir todos los permisos de un usuario es con el comando =

chmod go = w mitexto.txt Deshabilita todos los permisos y solo asigna lo indicado tras el signo =

CAMBIAR DE USUARIO – CAMBIAR CONTRASEÑA

whoami con el comando whoami conocemos el usuario actual
id otorga el nombre de nuestro usuario con informacion extra

su {NOMBRE DE USUARIO} SWITCH USER para cambiar de usuario
su root cambiar a usuario root (pero en Ubuntu no funciona)

sudo su PARA UBUNTU nos permite cambiar a root luego de introducir la contraseña
sudo -i cambia a usuario root

Creamos un archivo y volvemos a nuestro usuario para ver como no podemos modificar un archivo creado con el usuario

SOLO EL USUARIO ROOT TIENE PERMITIDO CAMBIAR LA CONTRASEÑA DE ROOT O CUALQUIER OTRO USUARIO

Para poder obtener permisos de usuario root de forma temporal utilizamos el comando

sudo sudo rm rootfile para poder remover el archivo rootfile

passwd modificar contraseña de usuario. Luego de este comando el sistema
sudo passwd root cambiar la contraseña del usuario que se indique(en este caso root)

VARIABLES DE ENTORNO (crear alias de forma permanente,

LINK SIMBOLICOS son un tipo de archivo que hacen referencia a una ruta

ln -s { RUTA } { NOMBRE DEL LINK SIMBOLICO}

In -s curso_terminal/sandbox Link_To_Sandbox crea un link llamado Link_To_Sandbox que nos dirige a esa carpet

se ve con ls -l y se accede con cd

VER TODAS LAS VARIABLES DE ENTORNO CONFIGURADAS

```
printenv
env
```

IMPRIMIR UNA VARIABLE DE ENTORNO

```
echo ${NOMBRE DE LA VARIABLE}      echo $HOME
```

VER LA RUTA DE DONDE SE ENCUENTRAN LOS BINARIOS

```
echo $PATH
```

MODIFICAR LAS VARIABLES DE ENTORNO Y CONCATENAR PATH

Ls -la y buscamos el archivo .bashrc (para el caso de bashshell)

Code .bashrc para abrirlo con Visual Studio Code (o un editor de texto)

Al abrir el archivo de configuración:

****podemos crear alias**

```
alias l='ls -lh'
```

Crear una variable de entorno se realiza con MAYUSCULA

```
PLATZI_MESSAGE="Hola amigos"
```

bash En la terminal cargamos el comando bash para que se actualicen los cambios realizados

Utilizando el comando “ alias ” podremos ver los alias creados de para nuestro usuario

MODIFICAR LA VARIABLE DE ENTORNO PATH ANEXANDO UN BINARIO

```
PATH=$PATH:{nueva ruta a anexar}
```

```
PATH=$PATH:/home/lucas/curso_terminal/bin
```

este ejemplo anexa a la variable ya existente la nueva ruta

COMANDOS DE BUSQUEDA: which Y find

Which + { nombre de un binario } which code = devuelve la ubicación del binario de VSC

Comando FIND ayuda a encontrar un archivo especificando la ruta desde donde lo buscamos, seguido del nombre utiliz:

```
Find + { la ruta desde donde lo quiero buscar. La ubicación actual es “ ./ ” } + -name + { nombre del archivo }
```

EJEMPLOS:

```
Find ./ -name file
```

Buscamos todos los archivos de nombre file

Find ./ -name *.txt Buscamos todos los archivos .txt utulizando el wildcar *

Find ./ -name *.txt | less Repetimos la instrucción y utilizamos el comando pipe operator para sumar la in

Utilizamos el modificador -type para buscar por tipo de archivos.
-type recibe como parámetros las opciones " f " para archivos y " d " para directorios.

Find ./ -type d -name Docum* Buscamos un directorio que se llame Docum...

Find ./ -type f -name '*.log' Buscamos los archivos de extensión .log

Buscamos archivos por tamaño con el modificador -size seguido del Tamaño requerido

Find ./ -size 20M Buscamos archivos mayores al tamaño de 20Megabytes

EJERCICIO: Buscamos todos los archivos .txt luego redireccionamos el ouput a un archivo, luego contatenamos la acció

Find ./ -type f -name '*.txt' > ejercicio_archivos_txt.txt && echo "EJERCICIO COMPLETADO. ARCHIVO CF

COMANDO DE BÚSQUEDA PARA COINCIDENCIAS DENTRO DE ARCHIVOS: grep

Grep también es un comando para ser utilizado en servidores y hacer búsquedas en los console log y en archivos de pro

Sintaxis: grep + { lo que se busca } + { nombre del archivo }

grep Towers movies.csv Ejemplo: Buscamos la palabra "Towers" dentro de un archivo .csv

grep Towers movies.csv

Con el modificaro -i "Ignore case sensitive" ignoramos si se trata de mayus o minus

grep -i the movies.csv

Contar el número de ocurrencias con el modificador -c

grep -c the movies.csv Devuelve el número de veces que aparece la palabra comenzando

grep -ci the movies.csv Realiza la cuenta ignorando si es mayus o mins

Buscando casos donde NO COINCIDE con la palabra con el modificador -v

grep -vi towers movies.csv

WC Comando para contar cuantas palabras hay.

Utiliza los modificadores -l, -w, -c
wc movies.csv

wc -l	Cuenta el número de líneas
wc -w	Cuenta el número de palabras
wc -c	Cuenta el número de bytes

COMANDOS DE UTILIDADES DE RED

ifconfig	Ver información de red. De no estar instalado el paquete a utilizar es: sudo apt install net-tools
ping ping www.google.com	Visualizar si una página esta activa. Detenerla con control+c
curl curl www.google.com Curl www.google.com > index.html	Visualizar / Traer la información de una página web
wget wget www.google.com	Traer y descargar un archivo directo al PC Trae la información y la guarda en un archivo en el directorio que nos encontrarr
tracert tracert www.google.com	Visualiza todas las direcciones de IP que recorreremos para llevar a un sitio
netstat -i	visualiza los dispositivos de red

COMPRESIÓN DE ARCHIVOS: Formatos tar , tar.gz, zip

Formato .tar

Sintaxis:
tar -cvf { nombre del archivo.tar } {archivo a comprimir}

Comando tar y sus modificadores -c para comprimir, v para que nos muestre un output, f para indicar que es un FILE
Luego indicamos el nombre del archivo comprimido a crear y seguido el archivo a comprimir/o directorio

tar -cvf ToCompress.tar ToCompress

Formato .gz muy eficiente para archivos de texto plano

tar -cvzf { nombre del archivo.tar } {archivo a comprimir} decorador z indica que se realizara en formato gz

Para Descomprimir:

Sintaxis: Tar-xvf { archivo.tar }

tar -xvf ToCompress.tar En formato tar

tar -xzvf ToCompress.tar.gz En formato .gz

Formato .zip

zip -r (manera recursiva) ToCompressInZip.zip ToCompress Comprimir

unzip ToCompressInZip.zip Descomprimir

Formato.rar

rar comprimir
unrar Descomprimir

MANEJO DE PROCESOS

ps VER LOS PROCESOS EN EJECUCIÓN EN LA TERMINAL ACTUAL
Se muestran con un numero de PID.

Kill + PID para cerrar definitivamente procesos
Por defecto el comando kill utiliza la señal 15, llamada SIGTERM. Si el comando falla podemos utilizar kill -9 +PID, haciendo

top VISUALIZA LOS PROCESOS Y SU CONSUMO DE RECURSOS

tecla h Ver los comandos a utilizar
tecla q Salir del menú
tecla u Filtrar por usuario:

PROCESOS EJECUTANDOSE EN FOREGROUND Y EN BACKGROUND

CONTROL + C Cancelar un proceso
CONTROL + D Terminar el input de un proceso y lo guarda
CONTROL + z Detiene un proceso (ejemplo cat > para editar un texto) y lo envía al Background

jobs Ver todos los procesos corriendo en el background

Fg + numero de proceso Continuar un proceso que se encuentra suspendido en el background trayendolo

Bg + número de proceso Detenido Enviar un proceso que este corriendo en el Foreground al Background

“ & “ Con el decorador & lo que hacemos es enviar al Background un proceso una vez ejecutado
por ejemplo cat > mi_nota.txt & . Luego vemos con jobs el proceso Detenido en el Background

EDITORES DE TEXTO EN LA TERMINAL - VIM

Instalar editor de texto VIM con administrador de paquetes de UBUNTU

```
sudo apt install vim
```

vim Para entrar a VIM

:q + Enter Salir de VIM (en modo normal)

Crear un archivo con el comando: vim NOMBRE_DE_ARCHIVO.extensión

Cambiar a modo INSERTAR presionando la tecla " i "

Volver al modo NORMAL para recorrer la pantalla Tecla ESC

Para buscar algo usamos la tecla /

Para guardar " :w"

Para guardar y salir " :wq"

Para salir sin guardar " :qa " Se indica al querer cancelar el proceso con Control + C

Para salir de manera forzosa " :wq! "

CONTROL + Z para volver a la terminal dejando el proceso suspendido en el background

Podemos verlo desde la consola con comando cat archivo.py

" ./ " punto slash permite ejecutar el archivo indicado a continuación del slash. Esto para los que posean un permiso de e

PERSONALIZAR LA TERMINAL

DE COMANDOS (SHELL)

: AL SISTEMA OPERATIVO PARA REALIZAR UNA ACCION

USUARIO

nto" antes de su nombre

cts
rojects

este escrito en pantalla pero si subimos podemos verlo

nombre de algún archivo que se encuentre en el directorio

en nuestra ubicación
estas opciones

ción

2

ontramos

caracteres indicados

```
ERROR > 0 >>
```

ivo. EL sistema tiene dos posibles salidas: un stdout (1) que realiza y muestra la petición o un stderr (2) que arroja un error (describe)

información de salida. Sea un error o una stdout

DE SALIDA
MACIÓN DE SALIDA SIN ELIMINAR LO QUE CONTENGA EL ARCHIVO

2. Dado que sin utilizar el 2> el errores no se graban en el archivo

STDOUT o STDERR

enido como un stdout

EL OUTPUT DE UNO TRAS OTRO

n Control +C

o escribiendo un archivo

ido del archivo y solo lo añade

[REDACTED]

NADA, DE FORMA SINCRONA O ASINCRONA

se ejecuta de manera satisfactoria

nte

[REDACTED]

hivo de bloque especial que contiene información de dispositivos como ser USB o disco duro

nte manipulación



ONES EN MODO OCTAL O PODEMOS UTILIZAR EL MODO SIMBOLICO PARA INDICAR QUE CATEGORIA DE PER

[REDACTED]

o root sin tener los permisos

RIO

na solicitará la contraseña nueva y luego su confirmación
ot)

[REDACTED]

a



ando el modificador -name

strucción less y verlo en páginas

ón con la impresión de un mensaje en pantalla.

READO”



ogramación donde buscamos una línea específica.

en minus.

[REDACTED]

nos. (.txt)

[REDACTED]

.E



endo mención a la señal SIGKILL



d asignando un número de proceso []

o al Foreground



ejecución (x)



or.

MISOS ESTAMOS MANIPULANDO.