Tendrán que investigar por su cuenta cómo funcionan los distintos comandos (como

grep, cat, sort, head), conectores de comandos como & ; pipes | y redirecciones ><.

Les dejamos un pequeño machete como ayuda ([Bite Size Linux.pdf)](https://drive.google.com/file/d/1QFROtJe_TtXnQqAf656BFQgQ8AyyYwpC/view). Otro recurso que les

puede ser de utilidad es el sitio explainshell.com, para comandos que encuentren

online y no sepan qué hacen.

**Grep:**

Searches the named input FILEs for lines containing a match to the given PATTERN. By default, grep prints the matching lines.

Se utiliza para buscar patrones de texto dentro de archivos o la salida de otros comandos.

Es una herramienta fundamental en la manipulación y búsqueda de texto en el shell, permitiendo filtrar y encontrar información rápidamente en archivos y flujos de datos.

Sintaxis básica: grep [opciones] patrón [archivo(s)]

EJ: grep "texto" archivo.txt (Busca la palabra texto dentro de archivo.txt y muestra todas las líneas que contienen esa palabra.

Principales opciones de grep:

* (-i), grep -i "texto" archivo.txt : busqueda no sensible a mayus.
* (-r), grep -r "texto" directorio/: busqueda recursiva dentro de un directorio
* (-l), grep -l "texto" directorio/: muestra solo los nombres de los archivos que contienen el patron, no las lineas.
* (-E), grep -r "texto" directorio/: permite usar expresiones regulares extendidas.
* …

Expresiones regulares: grep es muy poderoso cuando se usa con expresiones regulares, lo que permite buscar patrones complejos. Por ejemplo, buscar líneas que comiencen con "Error" seguido de cualquier cosa:

grep "^Error" archivo.txt

### **Uso combinado:**

Puedes combinar grep con otros comandos usando tuberías (|). Por ejemplo, para buscar procesos que contengan "ssh":

ps aux | grep ssh

**CAT:**

Utilizada para concatenar y mostrar el contenido de archivos.

Sintaxis basica: cat [opciones] [archivo(s)]

Ej simple: cat archivo.txt

>Esto muestra el contenido de archivo.txt en la salida estandar (terminal).

Funciones principales:

* Mostrar el contenido de un archivo:

cat archivo.txt

* Concatenar archivos: (para combinar el contenido de varios archivos en uno solo)

cat archivo1.txt archivo2.txt > combinado.txt

* Mostrar el contenido de varios archivos:

cat archivo1.txt archivo2.txt

* Crear un nuevo archivo:

cat > nuevoarchivo.txt.

Después de ejecutar este comando, puedes escribir el contenido que desees en nuevo\_archivo.txt. Para terminar y guardar el archivo, presiona Ctrl+D.

* Agregar contenido a un archivo existente:

cat >> archivo.txt

Esto permitirá añadir texto al final de archivo\_existente.txt sin sobrescribir el contenido existente.

Uso combinado:

cat archivo.txt | grep "palabra"

cat se utiliza a menudo junto con otras herramientas de línea de comandos. Por ejemplo, puedes usar cat para mostrar el contenido de un archivo y luego pasarlo a otro comando mediante una tubería (|)

**Sort:**

Se utiliza para ordenar líneas de texto en archivos o en la salida de otros comandos. sort organiza las líneas en orden alfabético o numérico, dependiendo de las opciones especificadas.

Sintaxis basica:

sort [opciones] [archivo(s)]

Ej: sort archivo.txt

Esto ordena alfabéticamente las líneas de archivo.txt y las muestra en la salida estándar.

Funciones principales:

* Ordenar en orden inverso:

sort -r archivo.txt

* Ordenar numericamente:si las lineas contienen numeros se ordenan numericamente

sort -n archivo.txt

* Ordenar por columna especifica: Si tienes un archivo con varias columnas (por ejemplo, separado por espacios o tabulaciones), puedes ordenar por una columna específica. Esto ordena el archivo por la segunda columna.

sort -k 2 archivo.txt

* Ordenar ignorando mayus y minus: sort -f archivo.txt
* Ordenar y eliminar duplicados: Para eliminar líneas duplicadas después de ordenar, puedes combinar sort con uniq:

sort archivo.txt | uniq

o en una sola operacion:

sort -u archivo.txt

* Especificar un delimitador de campo:Si tus columnas están separadas por un delimitador específico (como una coma), puedes indicarlo con la opción -t

sort -t, -k 2 archivo.csv

* GUardar la salida en un archivo: Puedes redirigir la salida ordenada a un archivo en lugar de mostrarla en pantalla:

sort archivo.txt > archivo\_ordenado.txt

-i, --ignore-nonprinting consider only printable characters

Se pueden **ordenar dos o más parámetros simultáneamente** con el comando sort.

Sintaxis: sort -k<campo1>,<campo1>[<tipo>] -k<campo2>,<campo2>[<tipo>] [archivo]

**-k<campo1>,<campo1>**: Define la primera clave de ordenación, que comienza en <campo1> y termina en <campo1>. Puedes añadir un tipo de orden (n para numérico, r para reverso) al final.

**-k<campo2>,<campo2>**: Define la segunda clave de ordenación, que comienza en <campo2> y termina en <campo2>. Similar a la primera clave, puedes definir el tipo de orden aquí.

**WC (WORD COUNT):**

Se utiliza para contar líneas, palabras y bytes (caracteres) en un archivo o entrada de texto.

Sintaxis basica: wc [opciones] [archivo(s)]

**Head:**

Muestra las primeras líneas de un archivo o de la salida de otro comando. Por defecto, muestra las primeras 10 líneas de un archivo, pero puedes especificar el número exacto de líneas que deseas ver.

Sintaxis: head [opciones] [archivo(s)]

Principales funciones:

* Especificar el numero de lineas que queres ver:

head -n 5 archivo.txt

* mostrar num especifico de bytes:

head -c 20 archivo.txt

* mostrar las primeras lineas de varios archivos:

head archivo1.txt archivo2.txt

* Combinar con otros comandos:

ps aux | head

**TR:**

se utiliza para traducir, reemplazar o eliminar caracteres en la entrada estándar. Su nombre proviene de la palabra "translate" (traducir).

Sintaxis: tr [opciones] SET1 [SET2]

**SET1**: Conjunto de caracteres que deseas traducir o eliminar.

**SET2**: Conjunto de caracteres con los que deseas reemplazar los caracteres en SET1. Si no se proporciona SET2, tr eliminará los caracteres en SET1.

### **Opciones de tr**

* **-d**: Elimina los caracteres que están en el conjunto SET1.
* **-c**: Complementa el conjunto SET1, es decir, actúa sobre los caracteres que no están en SET1.

**CUT:**

se usa para extraer secciones de cada línea de archivos de texto o entrada estándar. Es útil para extraer columnas específicas de datos delimitados por caracteres, como comas o tabulaciones.

SINTAXIS: cut [opciones] [archivo]

OPCIONES:

* -d DELIM: Especifica el delimitador que separa los campos. El valor por defecto es la tabulación.

cut -d';' -f2 archivo.txt

* -f LIST: Selecciona los campos que deseas extraer, separados por delimitador. Puedes especificar un único campo, un rango de campos, o una lista de campos.
  + - **Campo único**:cut -d';' -f2 archivo.txt
    - **Rango de campos:** cut -d';' -f1-3 archivo.txt
* ….

**CURL:**

se utiliza para transferir datos desde o hacia un servidor utilizando diversos protocolos, como HTTP, HTTPS, FTP, y más. Su nombre proviene de "Client URL".

SINTAXIS: curl [opciones] [URL]

**AWK:**

Se utiliza para analizar y manipular datos en archivos de texto o entrada estándar.

SIntaxis: awk 'patrón { acción }' [archivo]

**patrón**: Condición que se debe cumplir para que se ejecute la acción.

**acción**: Código que se ejecuta cuando el patrón se cumple.

**archivo**: Archivo de entrada (si no se especifica, awk leerá de la entrada estándar).

**IP:**

Una herramienta poderosa para la configuración y gestión de redes. El comando ip permite manipular la configuración de redes, rutas y dispositivos de red de manera detallada.

Sintaxis: ip [opciones] objeto [subcomando] [opciones]

**[opciones]**: Opciones generales del comando.

**objeto**: El tipo de objeto que quieres manipular, como link, addr, route, etc.

**[subcomando]**: La acción específica que deseas realizar sobre el objeto.

**[opciones]**: Opciones específicas del subcoman

| !!!!!! no uses ifconfig/ipconfig, ya quedaron obsoletos!!!!!!!!!1!!!!!11!!!! para algo esta ip!!! |
| --- |