

## MANIPULACIÓN DE ARCHIVOS DE TEXTO.

Ingrese a un shell de línea de comandos de su equipo .

### Determinando el tipo de un archivo

En muchas ocasiones es extremadamente útil determinar que clase de contenido tiene un archivo antes de utilizarlo. Para determinar el tipo de un archivo determinado se utiliza el comando **file**.

Sintaxis: **file** <opciones> <archivo>

- Ingrese el comando **file --help** y a continuación <ENTER>. (1)¿Para qué sirven las opciones del comando file?
- Ingrese el comando **file /etc/passwd** y a continuación <ENTER>. (2)¿De qué tipo de archivo se trata?
- Ingrese el comando **file /bin/bash** y a continuación <ENTER>. (3)¿Y este archivo?
- Ingrese el comando **file /dev/tty1** y a continuación <ENTER>. (4)¿Qué tipo de archivo es?
- (5)¿De qué tipo es el archivo **/dev/sda**? ¿Y el archivo **/home**? ¿Y el archivo **/dev/psaux**?

### Mostrando por pantalla el contenido de un archivo de texto

En incontables ocasiones el usuario se verá en la necesidad de ver por pantalla de manera rápida y simple el contenido de un archivo de texto dado.

Una de las posibles alternativas es el uso del comando **more**. El comando **more** tiene la finalidad de mostrar el contenido de un archivo de texto por pantalla de manera paginada (deteniéndose cada vez que se llena la pantalla).

Sintaxis: **more** <opciones> <archivo>

- Ingrese el comando **more /etc/hosts** y a continuación <ENTER>. (6)¿Ve el contenido del archivo?
- Ahora Ingrese el comando **more /etc/services** y a continuación <ENTER>. (7)¿Ve el contenido del archivo? ¿En qué se diferencia el comportamiento del comando **more**?
- Presione la barra espaciadora. Compárelo con el efecto de presionar <ENTER>.

Otra manera de ver el contenido de un archivo de texto es el uso del comando **less**. El comando **less** tiene la finalidad de mostrar el contenido de un archivo de texto por pantalla de manera paginada (deteniéndose cada vez que se llena la pantalla). El comando **less** permite ver la pantalla siguiente con las mismas teclas que **more** y, si lo deseamos, a diferencia de **more**, podemos ver la pantalla anterior usando las teclas del cursor o bien mediante <b> e <y>.

- Ahora Ingrese el comando **less /etc/services** y a continuación <ENTER>. (8)¿En qué se diferencia del comportamiento del comando **more**?

### Mostrando por pantalla sólo parte del contenido de un archivo de texto

Cuando trabajemos con archivos de texto cuyo tamaño sea considerable (esto es algo realmente común en las tareas habituales de un administrador de servidores Linux) en ocasiones sólo deseamos ver parte de un archivo de texto (usualmente las *n* primeras líneas o las *n* últimas líneas).

Visualizando el comienzo de un archivo

Para visualizar las *n* primeras líneas de un archivo de texto se utiliza el comando **head**.

Sintaxis: **head** <opciones> <archivo>

- Ingrese el comando **head --help** y a continuación <ENTER>. ¿Para qué sirven las opciones del comando **head**?

- Ingrese el comando **head /etc/services** y a continuación <ENTER>. **(9)**¿Cuántas líneas del archivo **/etc/services** le mostró?
- Ahora Ingrese el comando **head -n 15 /etc/services** y a continuación <ENTER>. **(10)**¿Cuántas líneas del archivo **/etc/services** le mostró ahora?

### Visualizando el final de un archivo

Para visualizar las n últimas líneas de un archivo de texto se utiliza el comando **tail**.

Sintaxis: **tail <opciones> <archivo>**

- Ingrese el comando **tail --help** y a continuación <ENTER>. ¿Para qué sirven las opciones del comando **tail**?
- Ingrese el comando **tail /etc/services** y a continuación <ENTER>. **(11)**¿Cuántas líneas del archivo **/etc/services** le mostró?
- **(12)** ¿Cómo haría para que el comando **tail** muestre las últimas **n** líneas?

### Buscando patrones dentro de un archivo de texto

Muchas veces necesitamos encontrar las líneas de un archivo de texto que contienen una determinada subcadena o patrón de texto. Para mostrar las líneas que contienen un patrón de texto se utiliza el comando **grep**.

Sintaxis: **grep <opciones> <patrón> <archivo>**

- Ingrese el comando **grep --help** y a continuación <ENTER>. ¿Para qué sirven las opciones del comando **grep**?
- Ingrese el comando **grep <nombre\_usuario> /etc/passwd** y a continuación <ENTER>. **(13)**¿Qué haría este comando?¿Qué información mostró?¿Porqué mostró solamente la línea alumno?
- Ingrese el comando **grep bash /etc/passwd** y a continuación <ENTER>. **(14)**¿Qué haría este comando?¿Qué información mostró?
- Ahora el comando **grep -v bash /etc/passwd** y a continuación <ENTER>. **(15)**¿Qué haría este comando?¿Qué información mostró?¿Nota la diferencia?

### Contando las líneas/palabras/bytes de un archivo

Para contar cuántas líneas/palabras/bytes tiene un archivo de texto se utiliza el comando **wc**.

Sintaxis: **wc <opciones> <archivo>**

- Ingrese el comando **wc --help** y a continuación <ENTER>. ¿Para qué sirven las opciones del comando **wc**?
- Ingrese el comando **wc /etc/passwd** y a continuación <ENTER>. **(16)**¿Qué haría este comando? ¿Qué información mostró sobre el archivo **/etc/passwd**? Pruebe este comando con las opciones **-l**, **-c** y **-w** y compare cada una de las tres salidas

### Separando (partiendo) un archivo

Para partir un archivo en dos o mas partes de igual tamaño se emplea el comando **split**.

Sintaxis: **split <opciones> <archivo> <prefijo>**

- Ingrese el comando **split --help** y a continuación <ENTER>. ¿Para qué sirven las opciones del comando **split**?
- Ingrese el comando **split -l 10 /etc/passwd partido** y a continuación <ENTER>. ¿Qué haría este comando? Liste los archivos del directorio actual. ¿En cuantas partes separo el archivo **password**? Determine cuántas líneas tiene cada parte usando el comando **wc**. Borre todas las partes del archivo **partido**.
- Ingrese el comando **split -b 10 /etc/passwd partido** y a continuación <ENTER>. ¿Qué haría este comando?Liste los archivos del directorio actual. ¿En cuántas partes separó el

archivo password? ¿Son más que las de la opción `-l 10`, por qué? Borre todas las partes del archivo **partido**.

- Ingrese el comando `split -b 10k /etc/passwd partido` y a continuación <ENTER>. ¿Qué haría este comando? Liste los archivos del directorio actual. ¿En cuántas partes separó el archivo password? ¿Son más o menos que las de la opción `-b 10`? ¿por qué? Borre todas las partes del archivo **partido**.

### **Uniando (concatenando) varios archivos**

Para concatenar uno o más archivos y visualizarlos en la salida estándar (pantalla) se utiliza el comando `cat`.

Sintaxis: `cat <opciones> <archivo1> <archivo2> ... <archivon>`

- Ingrese el comando `cat --help` y a continuación <ENTER>. ¿Para qué sirven las opciones del comando `cat`?

- Ingrese el comando `split -l 10 /etc/passwd partido` y a continuación <ENTER>.

- Ingrese el comando `cat partidoaa` y a continuación <ENTER>. ¿Que ocurrió?

- Ingrese el comando `cat partidoaa partidoab` y a continuación <ENTER>. ¿Y ahora? ¿Nota cómo funciona el comando `cat`?

### **Otros editores de línea de comando**

Existen multitud de editores de línea de comandos para Unix/Linux que podríamos utilizar. El elegir uno en particular dependerá de gustos personales y disponibilidad de los mismos en el sistema que estemos usando.

En particular cabe destacar un conjunto de editores que se crearon con la finalidad de resultar familiares a los usuarios del procesador de textos WordStar para DOS.

Estos son:

- Nano.
- Pico.
- Joe.

La disponibilidad de alguno o algunos de éstos dependerá de la distribución de Linux y las opciones de instalación de la misma.

Además existen gran numero de editores orientados a brindar facilidades a los programadores, ejemplo de estos son: Emacs y jed.