**Terminal y sus comandos.**

La terminal no es otra cosa más que una herramienta que por medio de comandos nos ayuda a poder manejar mejor un sistema como Linux. Con estos comandos podemos llegar a salvar archivos eliminar, acomodar o hasta crearlos. La terminal es algo bastante efectivo y muy necesario para muchos usuarios.

A continuación mencionaremos los 10 comandos mas básicos que puedes utilizar en una terminal:

**Sudo**

Este comando te da permisos de superusuario en la terminal de Linux, necesario para instalar o modificar partes importantes del sistema.

Ejemplo:

Digamos que busco abrir Firefox con permisos de superusuario, sólo tendría que escribir

Sudo firefox

Y así ejecutaré Firefox como superusuario.

**Apt-get**

Este comando es uno de los más utilizados en Ubuntu y cuenta con una gran variedad de aplicaciones, por ejemplo: ***install***.

Con ***apt-get install*** podemos realizar la instalación de programas mediante la terminal de Linux a nuestro sistema de una forma bastante simple, tan sólo escribiendo el nombre del paquete que deseemos instalar (que en muchas ocasiones es igual al nombre del programa) normalmente acompañado por *sudo*, realizar la instalación de un programa quedaría así:

Sudo apt-get install programa

Por ejemplo, para instalar Gimp, sólo hay que escribir

Sudo apt-get install gimp

Y esperar a que se instale el programa.

**Cd**

Este es el comando utilizado para cambiar de directorios en la terminal, su uso es bastante simple y consiste en escribir el directorio que deseemos después de escribir “Cd”.

Por ejemplo, para acceder a la carpeta “*Documentos*” de mi sistema:

Cd/home/usuario/Documentos

Y para regresar al directorio anterior sólo tienes que escribir Cd sin ninguna dirección después.

**Mkdir**

La función de mkdir no es más que la de crear directorios dentro de una carpeta, por ejemplo, si deseo crear “*Carpeta1*” dentro de “*Documentos*“, debería escribir

Mkdir/home/usuario/Documentos/Carpeta1

Pero eso no es todo lo que puedes hacer con *mkdir*, ya que también puedes crear arboles completos (esto es, directorios dentro de otros directorios), utilizando la función -p, por ejemplo

Mkdir-p/home/usuario/carpeta/uno/dos/

Y daría como resultado la creación del directorio “*carpeta*” dentro de “*usuario*“, y a su vez, tendría dentro el directorio “*uno*“, que dentro tendría el directorio “*dos*”

**Rm**

Este es el comando utilizado para borrar archivos y carpetas en la terminal de Linux, aunque es sencillo de usar, esto mismo lo convierte en un comando peligroso si no se usa con cuidado. Para utilizarlo sólo tienes que escribir rm seguido del directorio completo donde se encuentra el archivo que buscas eliminar.

Por ejemplo, para eliminar el archivo “*archivo.txt*” ubicado en la carpeta “*Descargas*” de mi usuario, escribiría

Rm/home/usuario/Descargas/archivo.txt

Y el archivo sería eliminado sin opción a recuperación (Es decir, no irá a la papelera de reciclaje, aunque aún podrá recuperarse con programas de recuperación de archivos)

**Man**

Uno de los comandos que más puede ayudarte en tu uso de la terminal de Linux es man. Este comando nos permite obtener el manual de instrucciones del comando que deseemos, por ejemplo.

Man sudo

Nos mostrará el manual de uso de sudo; man es aplicable con prácticamente todos los comandos.

**Clear**

Y para terminar, el comando clear. Este comando no afecta realmente al sistema, pero si nos permite limpiar la terminal de todo lo que se ha escrito, para que quede como al momento de abrirla.

Para utilizar este comando, sólo tienes que escribir

Clear

Sin más, e inmediatamente se borrará toda la información de la terminal. También puedes reemplazarlo por el atajo de teclado Ctrl+L.

**Cat**

Cat (de concatenar), es una maravillosa utilidad que nos permite visualizar el contenido de un archivo de texto sin la necesidad de un editor. Para utilizarlo solo debemos mencionarlo junto al archivo que deseamos visualizar:

$ cat prueba.txt

**Ls**

Ls (de listar), permite listar el contenido de un directorio o fichero. La sintaxis es:

$ ls /home/directorio

El comando ls tiene varias opciones que permiten organizar la salida, lo que resulta particularmente útil cuando es muy grande. Por ejemplo, puedes usar -apara mostrar los archivos ocultos y -l para mostrar los usuarios, permisos y la fecha de los archivos. Así como para todos los comandos Linux, estas opciones pueden combinarse, terminando en algo como:

Touch

Touch crea un archivo vacío, si el archivo existe actualiza la hora de modificación. Para crear el archivo prueba1.txt en /home, seria:

$ touch /home/prueba1.txt