

Caso práctico

María esta delante del ordenador junto a **Juan**.

-**María**, ¿Sabes utilizar Linux?

-Sí un poco, lo tengo instalado en mi equipo y lo suelo utilizar bastante.

-Pues me ha dicho **Ada** que tengo que utilizarlo ¿Me ayudas?

-¡Claro que sí! Vamos a tu equipo y lo utilizamos.



[Ministerio de Educación y Formación Profesional](#). (Dominio público)

Materiales formativos de FP Online propiedad del Ministerio de Educación y Formación Profesional.

[Aviso Legal](#)

1.- Unix y GNU/Linux. Historia. Distribuciones.Libro.

Caso práctico

Juan va a instalar Linux en su equipo con **Maria**.

-He visto en Internet que hay muchas distribuciones de Linux y no sé la que utilizar.

-Existen muchas distribuciones y cada una tiene ventajas o desventajas sobre el resto. Lo importante es saber para qué queremos utilizar Linux y utilizar la distribución que mejor se adapta a nuestra necesidades. Mira vamos a ver las distribuciones más importantes para seleccionar la que mejor se adapta a tus necesidades.



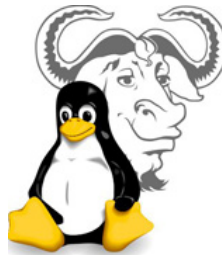
1.1.- Historia de GNU-Linux.

En los años 80 del pasado siglo, se introducen los PC (Personal Computer). Los estudiantes, en las grandes computadoras de la Universidad, utilizaban el sistema operativo UNIX y programaban en C. Pero, la mayor parte de las prácticas las tenían que realizar en papel, por no disponer de un sistema operativo similar para los PC.

De ahí, que el estudiante finlandés Linus Torvalds, de la Universidad de Helsinki, en el año 1991, diseñó un Sistema Operativo para PC desde cero, pero que funcionara con los mismos comandos que Unix. De esta forma, los estudiantes podían practicar Unix en sus PC de casa. Para ello comenzó trabajando sobre el código fuente de Minix (un pequeño UNIX para PC).

El estudiante Linus, liberó el código. Se denominó Linux, como contracción de Linus y Unix. La respuesta de los programadores fue contundente. Pronto todos querían aportar sus conocimientos para que Linux se convirtiera en un sistema operativo estable, robusto y potente. Finalmente llegó la primera versión estable delKernel, la versión 1.0.

De esta forma, Linux se unió al proyecto GNU de Richard Stallman de software libre (concepto explicado en la unidad 1). De esta forma, nació GNU-Linux como sistema operativo libre.



Materiales originales del MEC de FP a Distancia

En la imagen, se representan las mascotas que representan a GNU y a Linux. El animal ñu representa a GNU y el pingüino Tux, representa a Linux.

Hay que hacer hincapié, es que GNU-Linux utiliza en general los mismos comandos que UNIX, pero no son el mismo sistema operativo. Al contrario que GNU-Linux, el sistema UNIX no es libre. De hecho, el acrónimo GNU, significa "GNU no es Unix"

Hoy en día se pueden encontrar sistemas GNU/Linux en multitud de sistemas: desde grandes servidores pasando por PCs y llegando hasta los dispositivos móviles (Android parte de un kernel Linux)

También, sistemas operativos muy conocidos como MAC OS y Solaris, están basados en Unix.

Tux es el nombre de la mascota oficial de Linux. Fue creada por Larry Ewing en 1996. La idea de que la mascota del kernel Linux fuera un pingüino, provino del mismo Linus Torvalds, ya que, según se cuenta, cuando era niño le picó un pingüino, y le resultó simpática la idea de asociar un pingüino a su proyecto.

1.2.- Distribuciones de GNU-Linux.

Una **distribución** Linux, consiste en 2 partes:

- ✓ **Kernel** o núcleo del Sistema Operativo.
- ✓ **Software** que acompaña la distribución.

El sistema operativo lo forma el kernel, que es todo lo necesario para que funciones el Sistema Operativo, pero al sistema operativo le acompaña software adicional que completa la distribución.

Para obtener más información sobre el kernel, ver el siguiente [enlace](#).

Una vez instalada una distribución GNU-Linux tenemos instaladas la mayor parte de aplicaciones que utiliza un usuario habitual: suite ofimática (LibreOffice u OpenOffice), lector pdf, navegador web (Mozilla), compresor, programa de vídeo, grabador de dvd (Brasero),...

Asimismo, al instalar GNU-Linux, se suele instalar la mayoría del hardware con sus drivers.

Puesto que el kernel y las aplicaciones son libres, cualquier empresa o particular puede crear su propio sistema operativo, creando su propia distribución; que lógicamente tendrá que ser libre. Por ese motivo, hay muchísimas distribuciones.

Hay **3 distribuciones principales**, de las que suelen derivar el resto: **Red Hat, Debian y Slackware**.

A partir del año 2003, nacen algunas distribuciones muy conocidas hoy día, como Ubuntu y Fedora.

La diferencia, entre Red Hat y Fedora, es que Fedora solo contiene software libre, mientras que Red Hat puede contener software gratuito, no libre. Además, Red Hat es distribución comercial (se cobra por soporte comercial y técnico), mientras que Fedora no lo es, siendo mantenida por la comunidad. Para ello, Red Hat colabora con Fedora.

Basada en Red Hat, tenemos CentOS, una distribución dirigida principalmente para servidores.

Ubuntu es una distribución relativamente reciente, nace en el 2004, y se ha implantado mayoritariamente. Ubuntu deriva de Debian y su éxito se debe a haber desarrollado un entorno gráfico, más fácil para los usuarios de Windows.

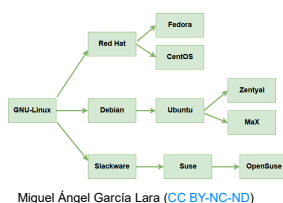
Zentyal, es una distribución española que ha cogido mucha fuerza, basada en Ubuntu tiene una interfaz web que facilita la administración de los servicios de red y está dirigida hacia las Pymes. Más información en el siguiente [enlace](#).

De Slackware, parte Suse, inicialmente alemana. Suse es comprada por la norteamericana Novell en el año 2003. Al igual que Red Hat, también libera una nueva distribución llamada OpenSuse, basada solo en software libre.

En España, varias Comunidades Autónomas han desarrollado su propio GNU-Linux: MaX (Madrid), LinEx (Extremadura), MoLinux (Castilla-La Mancha, molinos), Guadalinux (Andalucía). Algunas de estas distribuciones, se han dejado de desarrollar como MoLinux.

Información sobre [Fedora](#) y [OpenSuse](#).

En la imagen siguiente, se ven las distribuciones principales, junto con algunas españolas (MaX y Zentyal):



Para saber más

En el siguiente [vínculo](#) se puede ver un mapa completo de las distribuciones de Linux.

1.3.- Versiones de Ubuntu y Fedora.

Versiones de Ubuntu

Las versiones de Ubuntu, se basan en 2 números que representan el año y mes. Así, por ejemplo Ubuntu 17.10 significa que es de octubre del 2017.

En Ubuntu, se mantienen 2 versiones con distinto soporte:

- ✓ Versiones con soporte de 9 meses. Sale una cada medio año (salen en abril y octubre). No son versiones estables. La última es **Ubuntu 18.10** (octubre del 2018).
- ✓ Versiones con soporte extendido LTS. Es una versión más estable, y se mantiene el soporte de actualizaciones durante 5 años. Salen cada 2 años. La versión LTS más moderna es **Ubuntu 18.04 LTS** (abril del 2018)
Estas versiones se van actualizando, de forma, que en la actualidad, se pueden descargar la 18.04.1 LTS, 16.04.5 LTS o 14.04.5 LTS. Si tenemos instalada Ubuntu 14.04LTS, tenemos actualizaciones hasta abril del 2019. En ese momento, tendríamos que actualizar a una versión más moderna.
Lo habitual, es que instalemos versiones LTS, más estables y con mayor periodo de actualizaciones.
- ✓ [Vínculo](#) de descarga de las últimas versiones LTS y estándar.
- ✓ [Vínculo](#) de descarga de todas las versiones con vida útil.

Versiones según servicios que ofrece:

En versiones LTS, se puede descargar **Ubuntu 18.04 LTS Desktop y Ubuntu 18.04 LTS Server**

Como indican sus nombres, la versión Desktop (escritorio) está pensada para las estaciones de trabajo o PC habituales y la versión Server para los servidores.

La versión Server, instala más servicios, aunque una Desktop, también se podrá convertir en servidor en el futuro instalando los servicios necesarios.

Requisitos recomendados Ubuntu 18.04 LTS:

- ✓ Procesador 2 GHz dual core como mínimo
- ✓ Memoria RAM 2 GB
- ✓ 25 GB de espacio libre en disco duro

Versiones de Fedora

También sale una cada medio año. Son números correlativos, la primera versión salió en el 2003. La versión más nueva es Fedora 29.
[Página](#) de descarga de Linux Fedora 29.

1.4.- Entorno gráfico X-Windows.

Caso práctico

Juan y Carlos tienen que terminar un trabajo.

-Juan que escritorio más chulo tiene ¿Qué has hecho con tu ordenador?

-He instalado Linux en mi ordenador y mira, hay muchos entorno gráficos que podemos utilizar.



Uno de los elementos que ha propiciado la gran expansión de los sistemas GNU/Linux en empresas y hogares es la utilización de entornos gráficos sencillos y amigables. Los sistemas GNU/Linux cuentan con diversos entornos gráficos, muy potentes, que permiten utilizar el sistema fácilmente.

X-Windows (o *sistema de ventanas X* en castellano) es el nombre por el que se conoce al entorno gráfico usado por los sistemas Unix. Desarrollado desde mediados de la década de los 80 en el MIT (Instituto Tecnológico de Massachussets) se encuentra actualmente en su versión 11, por lo que normalmente suele ser referenciado como X11. El grupo de desarrollo inicial ha ido dando lugar en el tiempo a diversos sucesores: X Consortium (desde 1994 a 1996), The Open Group (de 1997 a 1998), XFree 86 (de 1992 a 2004) y X.org (desde 2004). X.org Foundation se fundó a partir de un grupo de desarrolladores y actualmente se encarga de desarrollar y coordinar el entorno X.

X-Windows proporciona una interfaz gráfica (GUI) al mundo de Linux. X-Windows, al igual que el sistema operativo Windows de Microsoft, ofrece una forma de manejo de algunos de los elementos de interacción más comunes como ventanas, cuadros de diálogo, botones y menús. X-Windows es quien proporciona las capacidades gráficas que hacen de las plataformas basadas en Linux la elección para el desarrollo de muchas aplicaciones de ingeniería y diseño, y es x-Windows lo que hace posible que Linux sea un competidor serio en el mercado de los sistemas operativos para PC.

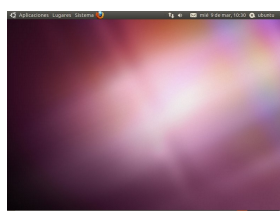
De forma simple, se puede decir que x-Windows es una interfaz gráfica completa para Linux y, por extensión, para Unix. X-Windows está compuesto por dos elementos principales: el **servidor X**, encargado de mostrar visualmente los elementos en la pantalla de forma totalmente independiente del sistema operativo, y el **gestor de ventanas**, cuyo objetivo es la gestión y administración de las ventanas mostradas para las aplicaciones, su apariencia, creación, colocación en la pantalla, etcétera. De esta forma x-Windows es capaz de distribuir el procesamiento de las aplicaciones siguiendo el paradigma cliente-servidor: el servidor provee los servicios para acceder a teclado, ratón y pantalla, mientras que los clientes son las aplicaciones que toman estos recursos para poder interactuar con los usuarios.

Esta forma de diseño *en dos partes* es lo que origina que existan diferentes implementaciones de gestores de ventanas, con diferentes características funcionales y visuales. El *servidor X*, como puedes imaginar es altamente portable y en el caso de Ubuntu permite utilizar los tres principales **entornos de escritorio GUI** (*Graphical User Interfaces* o *Interfaces Gráficas de Usuario*). GNOME está orientado a la simplicidad, KDE ofrece un mayor conjunto de aplicaciones así como posibilidades de caracterización por defecto, o Xfce está optimizado para su uso con requisitos hardware bajos. Aparte de los citados, es posible utilizar otros entornos de escritorio como *Fluxbox*, *Sugar* o *LXDE*.

Entornos gráficos más utilizados.

ENTORNOS GRÁFICOS
KDE
GNOME
Xfce

Tal y como puede ver en la siguiente figura, por defecto Ubuntu Desktop utiliza el entorno gráfico GNOME.



Miguel Ángel García Lara ([CC BY-ND](#))

Entorno gráfico de Ubuntu.

2.- Ejercicio. Instalación de Ubuntu en una máquina virtualLibro.

Caso práctico

-Hola **Maria**, ya he descargado la distribución Ubuntu y la he grabado en un CD. Ahora me toca instalarla en el equipo pero no me atrevo a hacerlo sólo.

-Estupendo, vamos a instalarlo en tu equipo y así aprendes.



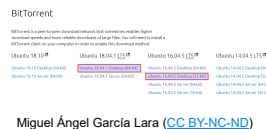
2.1.- Instalación de Ubuntu 18.04 LTS Desktop.

PASO 1. Descarga de la iso de Ubuntu

La iso se descargó en la tarea de la unidad de trabajo 2.
Si no lo hiciste entonces, el [vínculo](#) de descarga de la iso de Ubuntu.

Al igual que en la instalación de Windows, se recomienda descargar versión de 64 bits solo si tenemos al menos 8GB de RAM en nuestra máquina anfitrión.

Si vamos a instalar versión de 32 bits, descargaremos Ubuntu 16.04, pues Ubuntu no ha desarrollado versión de 32 bits para la versión 18.04.



PASO 2. Crear la máquina virtual y montar iso

Estos pasos son iguales, que cuando se instaló la máquina con Windows 10, por lo que aquí no se explican. En caso de duda mirar el libro I de la unidad 1.

Creamos la máquina *Ubuntu1804Sistemas* con memoria RAM 2GB (si se instala 32 bits, 1536 MB) y disco duro 100GB.

Montar en configuración de la máquina la iso de Ubuntu.

PASO 3. Instalar Ubuntu en la nueva máquina

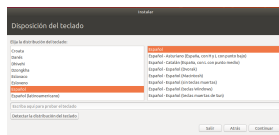
Arrancamos la máquina virtual con la iso de Ubuntu. Cuando esté iniciando, nos aparece la pantalla de bienvenida y nos pregunta idioma y si queremos instalar o probar.

Probar, sirve para arrancar cualquier máquina (tenga instalado Windows, Linux o incluso sin sistema operativo) con el CD y trabajar en Ubuntu sin haberlo instalado.

En nuestro caso seleccionamos **Español e Instalar Ubuntu**.

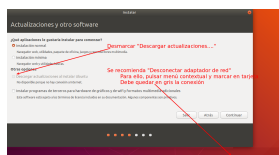


La siguiente pantalla, es para seleccionar idioma del teclado, ya debe aparecer **Español**. Pulsamos **Continuar**.

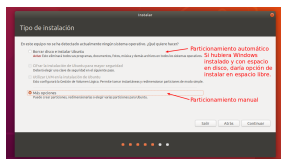


Para iniciar la instalación, el sistema muestra una pantalla preguntando si queremos una instalación normal o mínima. Además pregunta si queremos que descargue actualizaciones. En nuestro caso, realizamos **instalación normal y desmarcamos "Descargar actualizaciones..."** para evitar que la instalación se alargue en el tiempo.

También se recomienda desconectar la tarjeta de red de la máquina virtual (para evitar que el sistema pierda tiempo, intentando conectar la red).

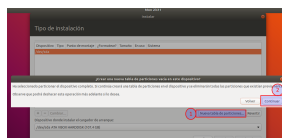


El siguiente paso es realizar el particionamiento del disco duro. Para ello, se puede seleccionar la opción Borrar y usar el disco entero en el caso de querer hacer el particionamiento automático, o se puede seleccionar “Más opciones” para especificar las particiones de forma manual. En nuestro caso, seleccionamos “**Más opciones**” para tener control total sobre las particiones a crear manualmente. Pulsar **Continuar**.



Miguel Ángel García Lara (CC BY-NC-ND)

En la siguiente pantalla, debemos pulsar “**Crear nueva tabla de particiones**” y **Siguiente**. Nos avisa de que los datos se borrarán. Al pulsar “Crear nueva tabla...” lo que estamos haciendo es crear el sector MBR explicado en la unidad 3.

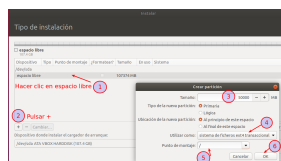


Miguel Ángel García Lara (CC BY-NC-ND)

Se van a crear **2 particiones**: la primera de 50GB para el sistema y la segunda va a ser la partición swap.

Para crear la **partición de sistema de 50GB** se seguirán los pasos siguientes:

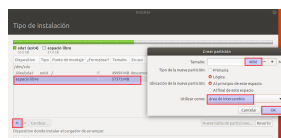
- ✓ Marcar espacio libre (todo el disco está libre)
- ✓ Pulsar + para añadir partición en espacio libre. Se abre la ventana “Crear partición”
- ✓ Rellenar tamaño, le decimos 50000MB (redondeamos 1GB=1000MB)
- ✓ Seleccionamos sistema de ficheros de Linux: ext4
- ✓ Seleccionamos el punto de montaje / Esto significa, que en esta partición vamos a instalar todo el sistema de ficheros de Linux (la raíz del sistema, anotada como /)
- ✓ Pulsamos OK



Miguel Ángel García Lara (CC BY-NC-ND)

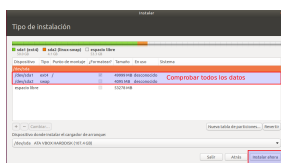
Creamos ahora la **partición swap** o área de intercambio. Como nuestra máquina virtual tiene poca memoria RAM, ponemos como tamaño de la partición el doble de memoria RAM.

Para crear la partición, seguimos los mismos pasos. Configuramos **4096MB** como tamaño, **primaria y área de intercambio**.



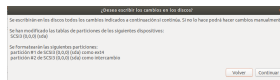
Miguel Ángel García Lara (CC BY-NC-ND)

En la siguiente imagen, se ven las 2 particiones creadas. Son particiones primarias, reconocidas por Linux como sda1 y sda2. Una vez creadas las dos particiones se pulsa **Instalar ahora**.



Miguel Ángel García Lara (CC BY-NC-ND)

Al pulsar Instalar ahora, nos aparece una ventana informando que si pulsamos Continuar, se guardarán todos los cambios. Se pulsa **Continuar**.



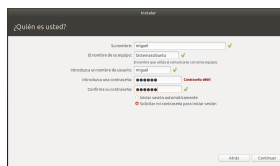
Miguel Ángel García Lara ([CC BY-NC-ND](#))

Posteriormente, nos pide seleccionar la zona horaria donde nos encontramos. Pulsar **Continuar**.



Miguel Ángel García Lara ([CC BY-NC-ND](#))

A continuación se nos pide crear una cuenta de usuario. Para ello, introduce tu **nombre**, nombre del equipo **SistemasUbuntu**, y password, tal como aparecen en la captura. Para acabar pulsar Continuar. Este usuario "tu_nombre" podrá realizar la administración de Ubuntu, pues tiene poderes para convertirse en superusuario o root. Se explicará esto con más detalle en el libro 5.D.



Miguel Ángel García Lara ([CC BY-NC-SA](#))

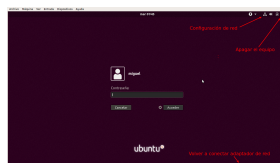
Ahora se realiza la instalación del sistema, que tarda de 10 a 20 minutos, según el sistema que tengamos. Para finalizar la instalación pulsar Reiniciar Ahora.



Miguel Ángel García Lara ([CC BY-NC-SA](#))

Cuando se está apagando, el sistema da un mensaje en el que hay que pulsar Intro. Este mensaje, sirve para decir que debemos extraer el CD. Esto significa, que al ser una máquina virtual, se desmonta la iso. Pero si estuviéramos instalando Ubuntu en una máquina real, al pulsar Intro se abriría la unidad del DVD automáticamente. (Así, se evitaría, arrancar con el CD de Ubuntu al iniciar el equipo).

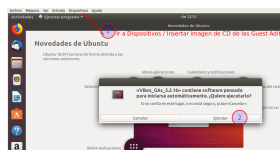
Una vez reiniciado el equipo, hay que pulsar Intro para que aparezca la pantalla de inicio de sesión. Una vez introducido el nombre de usuario y contraseña ya se puede empezar a utilizar el sistema.



Miguel Ángel García Lara ([CC BY-NC-SA](#))

PASO 4. Instalar Guest Additions

Realizamos la instalación de Guest Additions, igual que la hicimos en Windows.



Miguel Ángel García Lara ([CC BY-NC-SA](#))

Una vez instalado, si todo ha ido bien, ya tendremos la pantalla ajustada a nuestra máquina anfitrión.

3.- Primeros pasos en Linux.

Caso práctico

-Hola **Ana**. Estoy utilizando Linux y me han dicho que tu llevas utilizándolo desde hace mucho tiempo. La verdad es que estoy un poco perdido, porque es la primera vez.  Ana esta estudiando el ciclo DAIM.

-**Juan**, Linux es muy fácil de utilizar pero hay que hacerlo poco a poco. Mira, lo más importante que tienes que saber ahora es utilizar el sistema, conocer su estructura e instalar o quitar aplicaciones. Te lo enseño y así puedes empezar a utilizar bien el sistema.

Una de las grandes ventajas de los sistemas GNU/Linux es que se adapta completamente al nivel de conocimientos del usuario. Hoy en día, a través de los asistentes y los entornos gráficos es posible utilizar fácilmente los sistemas GNU/Linux sin necesidad de tener amplios conocimientos sobre el sistema.

Por supuesto, cuantos más conocimientos tengas mejor puedes aprovechar las prestaciones del sistema. A continuación se van a comentar las tareas más frecuentes en los sistemas GNU/Linux.

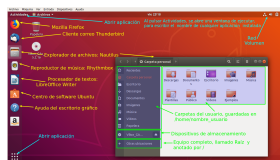
3.1.- Entorno gráfico y directorios de Linux.

Interfaz gráfica

La primera vez que se utiliza Ubuntu, es habitual sentirse perdido, pero es muy fácil movernos por el escritorio gráfico. El escritorio incorporado en Ubuntu, se basa en Gnome.

Nada más instalar Ubuntu ya tenemos un montón de aplicaciones instaladas: LibreOffice (suite ofimática), visor de documentos pdf, calculadora, editores de texto plano, compresor de archivos, Brasero (programa para grabar CD y DVD)

Se muestra captura del escritorio gráfico, donde se explica para qué sirve cada icono. Lo señalado en amarillo, son accesos directos, que se pueden eliminar y añadir de forma similar a como se realiza en Windows.



Miguel Ángel García Lara (CC BY-SA)

El navegador o explorador de archivos se llama **nautilus**.

Si tenemos varias ventanas abiertas, podemos navegar entre ellas con la combinación **Alt + Tab**.

El centro de software de Ubuntu, sirve para instalar y/o desinstalar las aplicaciones de forma gráfica. Ubuntu puede instalar un montón de software libre, siendo muy sencillo, instalar desde esta ventana. Cuando se instala la aplicación, Ubuntu busca el software en páginas web configuradas para ello, llamadas "repositorios de Ubuntu".



Miguel Ángel García Lara (CC BY)

Primeras diferencias entre Windows y Linux

Entre los sistemas operativos Microsoft Windows y los sistemas Gnu-Linux, hay algunas diferencias importantes, que nos ayudan a entender el sistema de archivos de Linux.

Microsoft Windows vs Unix y GNU-Linux

Microsoft Windows	Unix y GNU-Linux
Un árbol de directorios por cada unidad lógica. Si tenemos 3 particiones montadas, las tenemos en C:\ D:\ E:\ donde cada árbol tiene su raíz.	Árbol de directorios único, el directorio padre es / y a partir de él, cuelgan todos los directorios, incluidas las unidades de almacenamiento. Así, por ejemplo, el CD de las Guest Additions, se encuentra montado en: /media/tu_nombre/VBox_Gas_5.2.16
No diferencia mayúsculas y minúsculas. En una carpeta hoja.txt, Hoja.txt y HOJA.txt son el mismo archivo.	Diferencia mayúsculas y minúsculas. En una carpeta pueden existir hoja.txt, Hoja.txt y HOJA.txt, pues son archivos distintos.
En Windows, las extensiones de archivos son necesarias. Por ejemplo los archivos ejecutables se distinguen por su extensión exe, com, bat	En Linux, hay muchos archivos sin extensión. Los archivos ejecutables se distinguirán por sus permisos)
Los comandos, se pueden escribir con mayúsculas y minúsculas. Pues cuando ejecutamos un comando, realmente llamamos al archivo ejecutable con ese nombre. Y da lo mismo llamar a copy que a COPY pues son el mismo archivo. Si queremos abrir el explorador de archivos, podemos escribir explorer, Explorer, EXPLORER	Los comandos se escriben siempre en minúsculas, pues llamaríamos a distintos archivos ejecutables. Para evitar confusión, en Unix/Linux se crearon todos los ficheros ejecutables con minúsculas. Si queremos abrir el explorador de archivos, hay que escribir nautilus. No Nautilus, ni NAUTILUS.
El símbolo para los directorios es \ Ejemplo: la ruta del usuario juan es C:\users\juan	El símbolo para los directorios es / Ejemplo: la ruta del usuario juan es /home/juan
Se utiliza el signo / para las opciones de los comandos	Se utiliza el signo - para las opciones de los comandos

Directorios más importantes de Linux

Cuando nos enfrentamos por primera vez en Linux, nos encontramos perdidos, por llamarse los directorios de forma muy distinta a Windows.

- ✓ **/bin y /usr/bin:** contienen la mayoría de los ficheros ejecutables y comandos. Se llama bin, porque representa a los binarios puros. Se les llama así a los ejecutables, queriendo decir que son binarios puros, con 0 y 1 que solo entienden las máquinas.
- ✓ **/etc:** contiene los ficheros de configuración del sistema. Suelen ser archivos de texto plano editables.
- /home:** contiene el directorio home de todos los usuarios salvo el del usuario root. Por ejemplo, la carpeta del usuario juan, se encuentra en /home/juan.
- ✓ **/root:** contiene el directorio home del usuario root.
En Linux el usuario root representa al administrador, también se le llama superusuario.
- ✓ **/usr:** Aquí se instalan las aplicaciones de usuarios. Su tamaño puede ser elevado, usr viene de user.
- ✓ **/tmp y /usr/tmp:** Almacenan los ficheros temporales.
- ✓ **/boot:** contiene los ficheros necesarios para el arranque del sistema.
- ✓ **/dev:** contiene los archivos de dispositivo: discos, usb, puertos, terminales; dev viene de device (dispositivo)
Equivale a decir que tiene los drivers de los dispositivos.
Ejemplos de ficheros de dispositivos:
/dev/sda representa al primer disco duro (sata disk a)
/dev/tty1 representa la primera terminal de texto
/dev/cdrom representa el archivo de la unidad de dvd.
- ✓ **/mnt y /media:** Se utilizan para montar los dispositivos de almacenamiento o unidades lógicas. Cuando se montan automáticamente, se montan en /media. Cuando las montamos manualmente, se realiza de forma habitual en /mnt (mount)
- ✓ **/var:** se guardan los archivos con información del sistema, datos de aplicaciones. En /var/log se encuentran los archivos de registro.

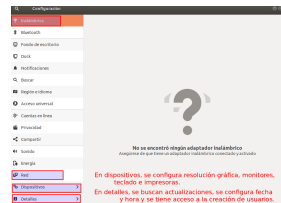
Observaciones:

- ✓ En el explorador de archivos nautilus, **"Otras ubicaciones" representa al /**. Al abrir otras ubicaciones se puede explorar todo el árbol de directorios.
- ✓ Hemos comentado que hay un único árbol raíz. De forma que un pendrive montado, se verá en /media/tu_usuario/label_dispositivo
- ✓ Cuando hemos instalado Linux en el anterior libro, hemos creado 2 particiones: / y swap, pero se pueden crear más particiones, utilizando esas particiones para un directorio concreto.
Por ejemplo, al instalar, un servidor se suelen crear estas 5 particiones: /, swap, /home, /boot, /tmp
¿Para qué sirve?
/home Como por defecto, los usuarios solo pueden escribir en /home, nunca llenarían la partición /
/boot Tener los archivos de arranque separados del resto del sistema. (al igual, que hizo Windows automáticamente al instalar)
/tmp Que los archivos temporales, no nos colapsen la partición /

3.2.- Configuración de Ubuntu.

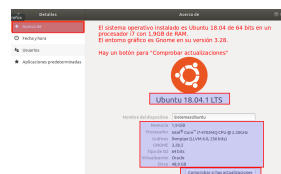
Configuración del sistema en entorno gráfico

Toda la configuración de Ubuntu se encuentra en la ventana “Configuración”. Pulsar en Actividades y escribir Configuración. En la ventana Configuración, se tiene acceso a varios submenús. Los más importantes son “Red”, “Dispositivos” y “Detalles”. En “Dispositivos”, se configura resolución gráfica, monitor, teclados, impresoras y otros dispositivos. En “Detalles” se tiene acceso a la creación de usuarios, así como a actualizar el sistema.



Miguel Ángel García Lara (CC BY-NC-SA)

Si pulsamos en Detalles. aparece la imagen siguiente.



Miguel Ángel García Lara (CC BY-NC-SA)

Creación de un usuario nuevo “alumno”

Como ejemplo, se va a crear el usuario alumno. Pinchar en “Detalles” y seleccionar “Usuarios”. Hay que desbloquear el programa, para poder crear y eliminar usuarios.



Miguel Ángel García Lara (CC BY-NC-SA)

Una vez desbloqueado, pulsamos en “Añadir usuario”. Se nos abre una nueva ventana, donde rellenamos nombre de usuario, configuramos la contraseña y pulsamos “Añadir”. Cuando creamos usuarios de forma gráfica, la contraseña tiene que tener cierta complejidad.

Por defecto el usuario creado es estándar. Es decir, no podrá convertirse a superusuario o root.



Miguel Ángel García Lara (CC BY-NC-SA)

4.- Terminal de comandos de GNU-Linux

El intérprete de comandos o `shell` del sistema es la interfaz entre el usuario y el sistema operativo. La función del shell es recibir las órdenes del usuario a través de la línea de comandos, interpretarlas, ejecutarlas y mostrar su resultado.

Resulta muy útil aprender a utilizar el shell del sistema ya que aunque al principio puede parecer un poco difícil, resulta fundamental para obtener el máximo rendimiento del sistema. El shell permite interactuar directamente con el sistema y con sus ficheros de configuración

4.1.- Interfaces de texto: terminales o consolas de texto.

Desde la interfaz gráfica iniciada, se puede iniciar la aplicación terminal, para ello pulsamos en Actividad y escribimos terminal. Además, en la mayoría de las versiones de Linux, aparte de la sesión gráfica, se inician varias terminales de texto. Estas sesiones gráficas o terminales de texto, llamadas tty1, tty2, ..., tty6 se muestran en pantalla, pulsado las teclas "Ctrl+Alt+F1", ..., "Ctrl+Alt+F6". Cada terminal puede estar iniciada por distintos usuarios. En Ubuntu 18.04 tty1 y tty2 son sesiones gráficas (por defecto se inicia la terminal gráfica tty2). Desde tty3 hasta tty6 son terminales de texto. Con la sesión gráfica iniciada, podemos cambiar a tty3 con Ctrl+Alt+F3 e iniciar sesión con otro usuario, y volver a la gráfica pulsando Ctrl+Alt+F2. En otros Linux, las terminales cambian, incluso en las versiones anteriores de Ubuntu, tty1 a tty6 representan terminales de texto, mientras las gráficas son tty7 y tty8.

Recordar que al utilizar VirtualBox, la combinación "Ctrl + Alt" se sustituye por la tecla anfitrión de VirtualBox "Ctrl derecha" (de las 2 teclas de control que hay en el teclado, la que hay a la derecha). De forma, que para iniciar sesión en tty1 en una máquina virtual se utiliza "Ctrl derecha + F1"

Sintaxis de los comandos

La sintaxis general de los comandos es:
comando [opciones][parámetros]

Ejemplo de comando:

Si queremos conocer lo que hay en el directorio home de alumno, con información larga y mostrando archivos ocultos, ejecutaremos el comando siguiente (las 2 formas son válidas):

```
ls -l -a /home/alumno  
ls -la /home/alumno
```

✓ Comando:

ls es el comando que muestra lo que tiene un directorio (posteriormente vemos el comando de forma más detallada). Los comandos siempre se escriben en minúsculas, como se ha comentado antes.

✓ Opciones del comando:

La opción l sirve para que la información de cada archivo y directorio sea larga o extendida. La opción a sirve para mostrar los archivos ocultos.

Cada opción se escribe con un guion delante, o con un único guion, en este caso las opciones no se separan con espacio, siendo equivalentes: -l -a y -la

✓ Parámetros:

Donde vamos a realizar la acción, en este caso en el directorio home de alumno: /home/alumno

Primeros comandos

Comenzamos con unos primeros comandos fáciles de utilizar, antes de entrar a los comandos de ficheros y directorios.

passwd usuario

Comando para cambiar la contraseña del usuario.

El sistema pide la contraseña antigua y después hay que introducir la nueva contraseña 2 veces.

exit

Comando para salir de la sesión.

Ejemplos:

Si tenemos una terminal abierta en la sesión gráfica, exit cierra la terminal.

Si estamos en la terminal de texto tty1, y escribimos exit, el sistema finaliza la sesión y vuelve a mostrar la petición de login para iniciar una nueva sesión.

man y help

La ayuda de un comando se puede solicitar de 2 formas distintas, por ejemplo, para pedir la ayuda del comando passwd, se puede ejecutar de las 2 formas siguientes:

1. **passwd --help** (atención, antes de help, hay un doble guion)
2. **man passwd**

El comando **help** ofrece la ayuda de forma resumida, mientras que **man** ofrece la ayuda completa del manual de Linux. Para desplazarnos por la ayuda podemos utilizar además de los cursores, las teclas RePg y AvPg. Para salir de nuevo al prompt o Shell del sistema se pulsa la tecla q (q viene de quit).

Usuarios de Linux. Trabajar como administrador. Cambios de usuario.

Trabajar como administrador

Cuando se ha instalado Linux, hemos creado un usuario "tu_nombre". Este usuario, tiene poderes para convertirse en superusuario o root.

Cualquier usuario posterior que se cree, no tendrá esos poderes, salvo que se configuren.

De momento, es importante tener claro que ese primer usuario creado "tu_nombre" y "root" son 2 usuarios distintos. Esto no es así en Windows, el usuario que se creó al instalar Windows era administrador, aunque debido al UAC Control de usuarios, se le pida identificación.

Además, en nuestro Ubuntu, hemos creado un tercer usuario "alumno", creado en anterior libro y que no tiene poderes para ser root.

sudo comando_a_ejecutar

Si queremos ejecutar un solo comando como root (superusuario o administrador) utilizamos sudo.

La terminal nos pedirá la contraseña de nuestro usuario, y ejecutará el comando con la identidad de root y sus permisos.

sudo su

Si queremos realizar varias comandos como root, es más cómodo cambiarnos temporalmente a root, para ello se ejecuta sudo su. En los siguientes comandos seguimos siendo root hasta que salgamos con el comando exit.

Es muy importante ser root sólo cuando sea necesario. Para entenderlo, los archivos tienen un propietario, que es quien lo ha creado. Si soy el usuario miguel y creo un archivo, tendré permisos para cambiarlo. Pero si creo el archivo siendo root, el usuario miguel no tendrá permisos para cambiar el fichero.

Cambiar a otro usuario. Comando su

Se puede cambiar a cualquier usuario en la terminal, utilizando:

su nombre_usuario

Al introducir el anterior comando, se solicita el password del usuario, y se cambia la sesión de la terminal a ese usuario.

su root

De la misma forma, **su root** sirve para cambiar la sesión a root.

Por defecto, esta opción no funciona en las distribuciones Ubuntu, pues el usuario root viene deshabilitado porque no tiene contraseña. Por ello, por defecto para ser root hay que utilizar obligatoriamente sudo. Sin embargo, en otras distribuciones Linux, viene deshabilitado sudo, por lo que hay que utilizar su.

Para que el usuario root, pueda tener sesión propia en Ubuntu, solo hay que poner contraseña a root con:

sudo passwd root

Leyenda para los ejemplos que se muestre

Ejemplo:

```
miguel@SistemasUbuntu:~$ cat /etc/shadow
```

```
cat: /etc/shadow: Permiso denegado
```

#Como usuario miguel se intenta mostrar el archivo /etc/shadow en pantalla. El sistema devuelve permiso denegado

```
miguel@SistemasUbuntu:~$ sudo cat /etc/shadow
```

```
[sudo] contraseña para miguel:
```

```
root:!:17848:0:99999:7:::
```

```
.....
```

#Se ejecuta el mismo comando pero con sudo, de forma que el sistema solicita el password de miguel, (usuario que puede convertirse en root). Se muestra la información del fichero en pantalla (este fichero tiene las contraseñas encriptadas de los usuarios)

```
miguel@SistemasUbuntu:~$ sudo su
```

#Con sudo su, se cambia a root, de forma que cuando devuelve el prompt o Shell, aparece la primera palabra root (que significa que está conectado el usuario root).

```
root@SistemasUbuntu:/home/miguel# cat /etc/shadow
```

```
root:!:17848:0:99999:7:::
```

```
.....
```

#Como ya somos root, podemos mostrar el archivo sin utilizar sudo

```
root@SistemasUbuntu:/home/miguel# mkdir carpeta1
```

#Creamos carpeta1 en directorio actual

```
root@SistemasUbuntu:/home/miguel# exit
```

```
exit
```

#Salimos de root, y volvemos al usuario miguel con exit (fijarse en shell)

```
miguel@SistemasUbuntu:~$ mkdir carpeta2
```

#Creamos carpeta2 en directorio actual

```
miguel@SistemasUbuntu:~$ ls -l
```

```
total 52
```

```
drwxr-xr-x 2 root root 4096 nov 22 12:24 carpeta1
```

```
drwxr-xr-x 2 miguel miguel 4096 nov 22 12:24 carpeta2
```

```
.....
```

#Listamos nuestro directorio actual, al listar se ve carpeta1 y carpeta2, pero se ven sus propietarios, donde el propietario de carpeta1 es root, y el propietario de carpeta2 es miguel

Varios comandos sencillos

poweroff

Apagar el equipo. En la mayoría de las distribuciones, hay que ser root.

reboot

Reiniciar el equipo.

who

Devuelve todos los usuarios conectados al equipo, bien en distintas terminales, bien desde la red.

echo

Igual que en Windows, echo mensaje devuelve en pantalla el mensaje.

pwd

Devuelve el directorio actual en el que nos encontramos.

clear

Limpia la pantalla (equivalente al cls de Windows)

Inicio de sesión del usuario en Linux. Directorio /home

Cuando un usuario inicia sesión en una consola de Linux, accede por defecto a su directorio \$HOME. En el directorio \$HOME de un usuario, solo tiene acceso a escribir ese usuario y los administradores. Por defecto, en Ubuntu, si se permite la lectura de los archivos de otros usuarios.

Si el usuario se llama alumno, su directorio \$HOME es /home/alumno

Cuando un usuario se encuentra en su \$HOME, en la mayoría de las shell aparece el símbolo ~ en lugar de /home/usuario

Significado de los parámetros del Prompt o Shell del sistema

```
root@localhost:/etc#  
1 2 3 4
```

1. Estamos conectados con el usuario root.
2. El nombre del ordenador es localhost
3. El directorio actual es /etc. Si aparece ~ significa que estamos en el directorio HOME del usuario.
4. La # identifica que el usuario conectado es un administrador. Cuando el usuario conectado no es administrador, se visualiza \$.

Ejemplo:

```
miguel@SistemasUbuntu:~/Documentos$ pwd  
/home/miguel/Documentos
```

#El comando pwd devuelve la ruta actual completa, como estamos en ~/Documentos y ~ representa \$HOME, la ruta completa en /home/miguel/Documentos

```
miguel@SistemasUbuntu:~/Documentos$ su alumno  
alumno@SistemasUbuntu:/home/miguel/Documentos$
```

#Al cambiar al usuario alumno, se ve los cambios en los parámetros 1 (usuario) y 3 (directorio actual)

4.2.- Comandos de directorios.

En este epígrafe se estudian los comandos para crear y eliminar directorios. En el siguiente los de ficheros, al final se incluye un ejemplo con la ejecución de varios comandos.

Antes de comenzar, veamos algunas analogías y diferencias entre Linux y Windows:

- ✓ La utilización de rutas absolutas y relativas, es igual en Linux que en Windows.
- ✓ Linux utiliza también los comodines *, ? con el mismo significado que Windows.
- ✓ En Linux, un archivo o directorio oculto es simplemente el que su nombre empieza por el carácter punto ".".
- ✓ En Linux, un archivo ejecutable no se distingue por su extensión, sino por sus permisos, donde se podrá configurar lectura, escritura y ejecución.

pwd

Muestra el directorio actual con su ruta absoluta

cd

Cambiar de directorio (change directory)

Ejemplos:

cd .. Este comando cambia al directorio padre

cd / Este comando cambia al directorio raíz

Estamos en el directorio /home/usuario2 y queremos cambiar al directorio home del usuario1. ¿Cómo se haría con ruta absoluta y con ruta relativa?

usuario2@SistemasUbuntu:~\$ **cd /home/usuario1** Usando ruta absoluta

usuario2@SistemasUbuntu:~\$ **cd ../usuario1** Usando ruta relativa

mkdir

Crear nuevos directorios

rmdir

rm -rf

El comando rmdir borra un directorio, pero tiene que estar vacío.

Para borrar un directorio con todo su árbol de ficheros y subdirectorios tenemos que utilizar **rm -rf**

En realidad, **rm** es el comando para borrar ficheros, pero para borrar un árbol de directorios, hay que utilizar este comando, con estas opciones, pues con rmdir no es posible.

tree

Muestra la información jerárquica de un directorio, con sus ficheros y subdirectorios.

ls

El comando ls lista (list) los archivos de un directorio

Sintaxis del comando: **ls [-laRtr] [ruta]**

Este comando tiene muchas opciones, las más importantes son:

-l La opción l, muestra información larga (long) o detallada de cada archivo (propietario, fecha, permisos)

-a La opción a, sirve para que al listar incluya los archivos ocultos. En Linux, un archivo o directorio oculto, es cuando su nombre empieza por el carácter punto. Por ejemplo, si listamos **ls -la** en nuestro \$HOME, vemos que hay un montón de archivos que empiezan por . como: .bashrc, .cache, .profile

-R La opción R muestra la información de los subdirectorios, es decir del árbol entero (R de recursive)

-t La opción t muestra los archivos ordenados por fecha (t de time)

-r La opción r, muestra el orden inverso. Por defecto al listar aparecen los archivos ordenados por nombre de la a a la z. Con la opción r, lo harían de la z a la a (r de reverse). Fijarse, que aunque hemos dicho que los comandos siempre se usan en minúsculas, sus opciones sí que las hay en mayúsculas y minúsculas con significados distintos.

Ejemplos:

- ✓ Obtener todos los archivos y directorios del sistema, incluyendo ocultos con información detallada: **ls -laR /**
 - ✓ Mostrar contenido del directorio /var/log en formato largo y mostrando archivos ocultos: **ls -la /var/log**
- Significado de la información que devuelve la opción **-l** (información larga o detallada)

drwxr-xr-x 2 alumno alumno 4096 2012-04-15 23:43 Descargas

1 2 3 4 5 6 7 8

1. Si el primer carácter es una "d", es un directorio. (en el ejemplo es el directorio descargas) Si no es "d" es un fichero. En el caso de fichero puede aparecer::

Un guion "-" que son los archivos normales o regulares

Una "b" que significa archivo de bloque, se usa en los dispositivos, por ejemplo los discos duros, donde las transferencias de información se realizan en bloques (o trozos)

Una "c" que significa archivo de carácter, se usa en los dispositivos de carácter a carácter, por ejemplo las terminales tty1, tty2,...

Una "l" es un enlace simbólico o blando, igual que los accesos directos de Windows.

2. Los otros 9 caracteres son los permisos de fichero. Lo vemos con detalle en la unidad 6. De momento, saber que si aparece la x en esos permisos, significa que el archivo es ejecutable (eXecute).

3. El siguiente campo, significa que tiene 2 enlaces duros. En Linux hay 2 tipos de enlaces: duros y blandos.

4. El usuario propietario del fichero es alumno.

5. El grupo propietario del fichero es alumno.

6. Tamaño del fichero, en bytes.

7. Fecha y Hora: Indica la fecha y la hora de creación o modificación del fichero.

8. Nombre del fichero o directorio.

Observación: Parece repetitivo decir que el propietario es alumno, y el grupo propietario es alumno. Pero no es así, en Linux todo usuario pertenece a un grupo principal, que por defecto, se llama igual que el. Cuando se creó el usuario alumno se realizaron 3 cosas:

1. Se creó el grupo alumno.

2. Se creó el usuario alumno.

3. Se introdujo al usuario alumno dentro del grupo alumno.

A partir de ahí, todos los ficheros que cree alumno, pertenecen al usuario alumno y al grupo alumno.

En la siguiente unidad se crearán grupos con distintos nombres que los usuarios.

4.3.- Comandos de ficheros

Editor de texto plano:

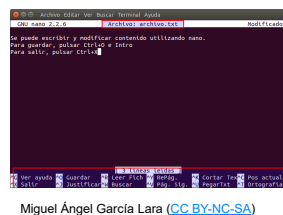
Si estamos en gráfico, Ubuntu incorpora el editor de texto plano gedit (equivalente al bloc de notas o notepad de windows)
En terminal, hay muchos, los más conocidos son: nano, vi, emacs.

En Ubuntu están instalados **nano y vi**. En entorno comando, desde el primer UNIX se utilizó el editor vi, muy completo pero que se utiliza solo con comandos. En nuestro caso, vamos a utilizar el editor nano, con menos opciones que el vi, pero fácil de utilizar.

Ejemplo: nano archivo.txt abre el editor nano. Si existía archivo.txt, muestra su contenido en pantalla. Si no existía, muestra vacío el contenido. En cualquier caso, podemos añadir o modificar el contenido que queramos.

Para salir y guardar hay que utilizar las combinaciones de tecla que aparecen en la misma ventana del editor en la zona de abajo.

Se muestra captura del editor nano (se ha ejecutado nano archivo.txt).



Miguel Angel García Lara (CC BY-NC-SA)

touch

Crea un fichero vacío, si ya existía, actualiza su hora de modificación.

Por ejemplo, touch archivo.txt crea archivo.txt sin contenido en el interior.

Visualización de fichero en terminal:

Los comandos cat, less y more sirven para visualizar archivos de texto en pantalla (equivalen al comando type en Windows)

Las diferencias entre ellos, es que cat muestra el contenido entero, devolviendo el Shell del sistema. Sin embargo, less y more, muestran el contenido del fichero, se puede subir y bajar con los cursores, y para salir se debe pulsar q (quit)

El comando head muestra las 10 primeras líneas de un fichero.

El comando tail muestra las 10 últimas líneas de un fichero.

Comando type

Realiza una acción totalmente distinta que en Windows. En Linux, muestra donde se encuentra el ejecutable de un comando.

Ejemplo:

¿Dónde se encuentra el fichero ejecutable del comando touch?

alumno@sisistemasUbuntu:~\$ type touch

touch is /usr/bin/touch El archivo touch se encuentra en /usr/bin/touch

Comando cp

Sintaxis: cp [-R] origen destino

El comando cp copia archivos o directorios, se pueden copiar con otro nombre.

-R La opción "R" copia recursivamente, toda la estructura de subdirectorios y ficheros. Copia el árbol entero.

Ejemplos:

- ✓ Estando en \$HOME de alumno, copiar el archivo "archivo.txt" en el \$HOME de miguel con el nombre copia.txt.
Utilizando ruta relativa:
alumno@sisistemasUbuntu:~\$ cp archivo.txt ../miguel/copia.txt
Utilizando ruta absoluta:
alumno@sisistemasUbuntu:~\$ cp /home/alumno/archivo.txt /home/miguel/copia.txt
- ✓ Copiar todos los archivos .txt del directorio home de usuario 1 al subdirectorio datos con la extensión .dat
cp /home/usuario1/*.txt /home/usuario1/datos/*.dat
- ✓ Copiar toda la estructura del directorio home de usuario 1 al directorio home de usuario 2 (con ruta absoluta)
cp -R /home/usuario1 /home/usuario2

Comando mv

Sintaxis: mv origen destino

El comando mv mueve archivos o directorios, se puede cambiar el nombre a la vez.

También sirve para renombrar ficheros o directorios.

Comando rm

Sintaxis: rm [-irf] fichero

El comando rm borra ficheros (remove)

-i La opción i pide confirmación antes de borrar cada archivo.

-r La opción r borra recursivamente, por lo que borra directorios enteros.

-f La opción f fuerza el borrado de los archivos, sin preguntar

Para borrar un directorio con una sola orden, y sin tener que confirmar cada borrado, se ejecuta rm -rf

Ejemplo:

Borrar el directorio hoy situado dentro del directorio home de alumno con ruta absoluta:
alumno@sistemasUbuntu:~\$ rm -rf /home/alumno/hoy

Direccionamientos “>”, “>>”, “2>”, “2>>”

Los operadores de direccionamiento > y >> se utilizan de la misma forma que en Windows. La salida esperada en la terminal; se redirecciona a un archivo.

El operador > sobrescribe el archivo y el operador >> añade a lo que ya tuviera.

Cuando se ejecuta un comando, se devuelven 2 salidas distintas en terminal: la esperada o estándar (buffer stdout) y los errores (buffer stderr). Mientras que los operadores > y >> redireccionan la salida estándar, los operadores 2> y 2>> redireccionan el buffer de errores.

4.4.- Ejemplo de comandos de directorios y ficheros.

En este ejemplo, se ejecutan varios comandos explicados en este libro.
Para probarlo, inicia tu Ubuntu en tu sesión "tu_nombre"

miguel@SistemasUbuntu:~\$ **su alumno**

Contraseña: Se ha iniciado terminal en la sesión de miguel. Cambiamos con el comando su al usuario alumno.

alumno@SistemasUbuntu:/home/miguel\$ **pwd**

/home/miguel Aunque hemos cambiado a sesión de alumno, seguimos estando en \$HOME de miguel, /home/miguel

alumno@SistemasUbuntu:/home/miguel\$ **cd /home/alumno**

Nos cambiamos al \$HOME de alumno

alumno@SistemasUbuntu:~\$ **nano archivo.txt**

Se escribe dentro cualquier cosa.

alumno@SistemasUbuntu:~\$ **cat archivo.txt**

Se puede escribir y modificar contenido utilizando nano

Para guardar, pulsar Ctrl+O e Intro

Para salir, pulsar Ctrl+X

Se muestra en pantalla el contenido escrito

alumno@SistemasUbuntu:~\$ **ls -l**

total 16

-rw-rw-r-- 1 alumno alumno 118 nov 26 22:17 archivo.txt

-rw-r--r-- 1 alumno alumno 8980 nov 22 13:04 examples.desktop

Hay 2 archivos en /home/alumno. El archivo creado, archivo.txt es de alumno, y tiene 118 bytes (118 caracteres)

alumno@SistemasUbuntu:~\$ **mkdir carpeta**

Se crea el directorio "carpeta"

alumno@SistemasUbuntu:~\$ **mv archivo.txt carpeta/otroNombre.txt**

Se mueve archivo.txt a la carpeta creada, guardándolo con otro nombre, y utilizando rutas relativas.

alumno@SistemasUbuntu:~\$ **echo añadir >> carpeta/otroNombre.txt**

Se añade la palabra "añadir" al final del archivo otroNombre.txt

alumno@SistemasUbuntu:~\$ **cp carpeta/otroNombre.txt carpeta/copia.txt**

Se realiza una copia del archivo, la llamamos copia.txt

alumno@SistemasUbuntu:~\$ **cat carpeta/copia.txt**

Se puede escribir y modificar contenido utilizando nano

Para guardar, pulsar Ctrl+O e Intro

Para salir, pulsar Ctrl+X

añadir

Se muestra el contenido de copia.txt

alumno@SistemasUbuntu:~\$ **echo sólo esta línea > carpeta/copia.txt**

Se sobrescribe copia.txt con "sólo esta línea"

alumno@SistemasUbuntu:~\$ **cat carpeta/copia.txt**

sólo esta línea

Se muestra el contenido de copia.txt

alumno@SistemasUbuntu:~\$ **cd carpeta**

Se cambia al directorio carpeta

alumno@SistemasUbuntu:~/carpeta\$ **pwd**

/home/alumno/carpet

La ruta actual es ~/carpeta es decir, /home/alumno/carpet

alumno@SistemasUbuntu:~/carpeta\$ **cd /home/miguel**

Se cambia al directorio \$HOME de miguel

alumno@SistemasUbuntu:/home/miguel\$ **rm -rf ../alumno/carpet**

Se borra la carpeta utilizando ruta relativa

alumno@SistemasUbuntu:/home/miguel\$ **ls -l /home/alumno**

total 12

-rw-r--r-- 1 alumno alumno 8980 nov 22 13:04 examples.desktop

Se lista el \$HOME de alumno. Se comprueba que ya no está la carpeta ni los archivos creados en el ejemplo.

alumno@SistemasUbuntu:/home/miguel\$ **exit**

exit

#Con exit se sale de la sesión de alumno, y se vuelve a la de miguel

#Guardar en archivo.txt "Calendario del mes actual" y el propio calendario, utilizando cal

miguel@SistemasUbuntu:~\$ **echo Calendario del mes actual > archivo.txt**

miguel@SistemasUbuntu:~\$ **cal >> archivo.txt**

miguel@SistemasUbuntu:~\$ **cat archivo.txt**

Calendario del mes actual

Diciembre 2018

do lu ma mi ju vi sa

1

2 3 4 5 6 7 8

9 10 11 12 13 14 15

16 17 18 19 20 21 22

23 24 25 26 27 28 29

30 31

#Direccionamientos. Para guardar en listado.txt todo el árbol de Linux, en formato largo y con ocultos, se ejecuta:

miguel@SistemasUbuntu:~\$ **ls -laR / > listado.txt** # Fijarse en las opciones: a (mostrar ocultos) y R (recursiva)

#El archivo listado se obtiene de forma correcta, pero además en la terminal aparecen errores porque no se tiene permiso para listar en muchos directorios

Vamos a direccionar las 2 salidas (la esperada y los errores) en 2 archivos distintos.

miguel@SistemasUbuntu:~\$ **ls -laR / > listado.txt 2> error.txt**

#Aparte de listado.txt, guardamos los errores en error.txt

miguel@SistemasUbuntu:~\$

4.5.- Instalación de aplicaciones.

Sólo los administradores pueden instalar aplicaciones. Para instalar programas en Linux se utilizan gestores de paquetes que facilitan la instalación.

Gestores de paquetes

Los gestores de paquetes de Linux, buscan de forma automática en los repositorios de software. Los repositorios de software, son servidores web, (páginas web) que facilitan ese software. Además, si para instalar esa aplicación, es necesario instalar otras previas, también las busca e instala. A esto se le dice que instala las dependencias necesarias. Gracias a estos gestores de paquetes, la instalación de software en Linux, es muy fácil.

1. Distribuciones basadas en Debian. Paquetes Debian o _____deb Gestores apt-get, aptitude y apt.

Cuando se instalan aplicaciones en Linux, tenemos que tener muy presente si estamos trabajando con una distribución basada en Debian, Red Hat o Suse, pues utilizan paquetes de instalación distintos. Como se comentó, Ubuntu está basada en Debian.

En las distribuciones Debian más modernas se utiliza el instalador de paquete apt. Anteriormente, se utilizaba apt-get. El instalador apt-get se mejoró con aptitude, pero en la actualidad apt es el más completo. Los comandos a utilizar son prácticamente iguales en todos los casos.

Sintaxis de apt

- ✓ Para instalar un paquete: `#apt install nombre_paquete`
- ✓ Para desinstalar: `#apt remove nombre_paquete`
- ✓ Si no sabemos el nombre exacto de un paquete, buscamos con: `#apt search nombre_aproximado`
- ✓ Para actualizar los repositorios: `#apt update`

Cambio de versión de Sistema Operativo:

Para cambiar a un sistema operativo más moderno, por ejemplo, para cambiar de Ubuntu 16.04 LTS a 18.04 LTS, se ejecuta en este orden:

```
$ sudo apt update (para actualizar repositorios)
$ sudo apt upgrade (para actualizar sistema operativo y aplicaciones)
$ sudo update-manager -d
```

Instalación gráfica:

En Ubuntu tenemos un icono directo al Centro de software de Ubuntu.

Esta aplicación gráfica se basa en apt, de forma que las aplicaciones que podremos instalar son las mismas a través de apt que con el Centro de software de Ubuntu.

Repositorios del software

Se ha comentado que apt busca en los repositorios, que son servidores web. ¿Dónde se encuentran guardadas estas direcciones web?, estas direcciones web se encuentran configuradas en el archivo `/etc/apt/sources.list`

Al instalar aplicaciones, algunas veces tenemos que añadir el repositorio en este archivo.

2. Distribuciones basadas en Red Hat. Paquetes Red Hat o _____rpm.

Herramienta yum

En las distribuciones Red Hat y derivadas: Fedora, CentOS se utiliza el gestor yum. Este instalador funciona igual que apt.

El programa yum instala los archivos rpm, instalando de forma automática las dependencias.

Sintaxis de yum

Instalar con yum: `#yum install nombre_paquete`
Desinstalar con yum: `#yum remove nombre_paquete`
Buscar un paquete: `#yum search nombre-paquete`
Actualización todos los paquetes: `#yum update`

Instalación sin gestores de instalación

Instalar paquetes _____deb o _____rpm sin gestores de instalación

También se puede realizar la instalación de paquetes _____deb o _____rpm directamente, sin utilizar gestores de instalación. Pero en este caso, la instalación suele ser más pesada.

Al instalar un paquete, busca las dependencias, sino tiene dependencias sin instalar, lo instala, pero si tiene, dice cuáles son esas dependencias y no lo instala. La forma de proceder, es instalar primero esas dependencias, y después el paquete que queremos. Esto suele ser tedioso, porque al instalar esas dependencias suele requerir otras.

Comando dpkg (para archivos _____deb en distribuciones Debian y derivadas)

Instalar un paquete deb: `#dpkg -i paquete.deb`

Desinstalar un paquete rpm: `#dpkg -r paquete.deb`

Comando rpm (para archivos _____rpm en distribuciones Red Hat y derivadas)

Instalar un paquete rpm: `#rpm -ivh paquete.rpm`

Desinstalar un paquete rpm: `#rpm -e paquete.rpm`

Instalar paquetes de código fuente en cualquier Linux

Los programas se pueden bajar de internet en código fuente. Básicamente, significa que ese programa está escrito en lenguaje C pero no está compilado.

Lo primero que hay que hacer es utilizar `tar -xvzf` para desempaquetar y descomprimir.

Después, nos cambiamos a la carpeta descomprimida, y ejecutamos los comandos siguientes:

```
#!/configure
#make
#make install
```

Para saber más

En la guía de Ubuntu se explica como añadir aplicaciones, entre las formas de hacerlo está cómo hacerlo con synaptic, lee el siguiente artículo que te resultará interesante.

[Añadir aplicaciones en Ubuntu](#)

Autoevaluación

¿Qué comando hay que ejecutar para actualizar el sistema?

- ☐ `apt-get upgrade`
- ☐ `apt-get update`
- ☐ `apt-get install system`
- ☐ `apt-get actualiza`

Correcto.

No es correcto, repasa los conocimientos del apartado.

Incorrecto, repasa los conocimientos del apartado.



No es la opción correcta, repasa los conocimientos del apartado.

Solución

1. Opción correcta
2. Incorrecto
3. Incorrecto
4. Incorrecto

Anexo.- Licencias de recursos.

Licencias de recursos utilizados en la Unidad de Trabajo.

Recurso (1)	Datos del recurso (1)	Recurso (2)	Datos del recurso (2)
	Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de la pagina web: http://windows.microsoft.com/es-ES/windows7/products/compare		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de la pagina web: http://www.microsoft.com/es-es/howtotell/Software.aspx#Packaging
	Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.
	Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.
	Autoría: Oracle VM VirtualBox Licencia: Copyright cita. Procedencia: Captura de pantalla de VirtualBox.		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.
	Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows , ejecutándose en maquina virtual de Oracle.
	Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.
	Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de GNU/GRUB.		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.
	Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.
	Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.
	Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.
	Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.		Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.

			
	<p>Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.</p>		<p>Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.</p>
	<p>Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.</p>		<p>Autoría: Microsoft. Licencia: Copyright cita. Procedencia: Captura de pantalla de Windows 7.</p>
	<p>Autoría: Kbolino Licencia: CC-BY-SA-2.5.</p> <p>Procedencia: Montaje sobre http://es.wikipedia.org/wiki/Archivo:GUID_Partition_Table_Scheme.svg</p>		<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>
	<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>		<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>
	<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>		<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>
	<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>		<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>
	<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>		<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>
	<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>		<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>
	<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>		<p>Autoría: Microsoft (Windows 7) y Oracle (VM VirtualBox). Licencia: Copyright 'Derecho de cita'. Procedencia: Capturas de pantalla de una máquina virtual Oracle VM VirtualBox, que ejecuta la instalación de Windows 7 de Microsoft.</p>

