



5. Instalación y explotación de un sistema Linux

Autor	ⓧ Xerach Casanova
Clase	Sistemas Informáticos
Fecha	@Jan 6, 2021 6:13 PM

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

1.1. Historia de GNU-Linux

1.2. Distribuciones de GNU-Linux

1.3. Versiones de Ubuntu y Fedora

Versiones de Ubuntu

Versiones según servicios que ofrece

Versiones de Fedora

1.4. Entorno gráfico X-Windows

2. Instalación de Ubuntu en una máquina virtual

PASO 1. Descarga de la iso de Ubuntu

PASO 2. Crear la máquina virtual y montar iso

PASO 3. Instalar Ubuntu en la nueva máquina

PASO 4. Instalar Guest Additions

3. Primeros pasos en Linux

3.1. Entorno gráfico y directorios de Linux

Interfaz gráfica

Diferencias entre Windows y Linux

Directorios más importantes en Linux

Observaciones

3.2. Configuración de Ubuntu

Creación de un nuevo usuario

4. Terminal de comandos de GNU-Linux

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

Sintaxis de comandos

Primeros comandos

Usuarios de Linux. Trabajar como administrador. Cambios de usuario

Cambiar a otro usuario. Comando su

Varios comandos sencillos

Inicio de sesión del usuario en Linux. Directorio /home

Significado de los parámetros del Prompt o Shell del sistema

4.2. Comandos de directorios

4.3. Comandos de ficheros

Editor de texto plano:

4.5. Instalación de aplicaciones

Distribuciones basadas en Debian (paquetes debian o .deb, gestores apt-get y aptitude y apt

Distribuciones basadas en Red Hat. Paquetes Red Hat o .rpm. Herramienta yum

Instalar sin gestores de instalación

Instalar paquetes de código fuente en cualquier Linux

Mapa conceptual

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

1.1. Historia de GNU-Linux

En los años 80 los estudiantes utilizaban el sistema operativo UNIX y programaban en C, realizando las prácticas en papel por no disponer de un sistema operativo similar para los PC.

Linus Torvalds, de la Universidad de Helsinki, diseña un S.O. en 1991 para PC desde cero, que funcionaba con los mismos comandos que UNIX. Para ello comenzó trabajando el código fuente de Minix (un pequeño UNIX para PC).

Una vez liberado el código, muchos programadores aportaron conocimientos para que se convirtiera en un S.O. estable, robusto y potente, hasta que llegó la versión 1.0 del Kernel.

Linux se une al proyecto GNU de Richard Stallman de software libre y nace GNU-Linux.

GNU-Linux utiliza en general los mismos comandos que UNIX, pero no son el mismo S.O. UNIX no es libre y GNU significa "no es UNIX. Hoy en día se encuentra en multitud de sistemas, de hecho, Android parte de un kernel Linux. Mac OS y Solaris están basados en UNIX.

1.2. Distribuciones de GNU-Linux

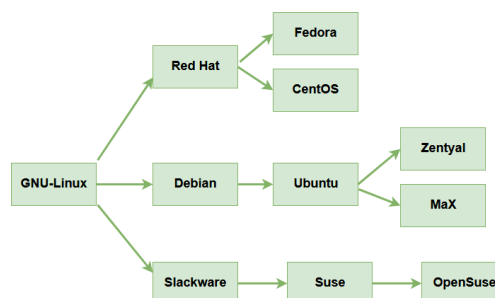
Una distribución de Linux consiste en 2 partes:

- Kernel (núcleo del S.O.).
- Software que acompaña a la distribución.

El S.O. lo forma el kernel, tiene todo lo necesario para que funcione el S.O., pero a este le acompaña software adicional que completa la distribución.

Cuando se instala una distribución GNU-Linux se obtienen la mayor parte de aplicaciones que utiliza un usuario habitual: suite ofimática:

LibreOffice/OpenOffice, lector PDF, navegador web (Mozilla), compresor, programa de vídeo, grabador de DVD(Brasero),...



Al instalar GNU-Linux también se suele instalar la mayoría del hardware y sus drivers. Cualquier empresa o particular puede crear su propio S.O. con su propia distribución, la cual debe ser libre.

Hay 3 distribuciones principales de las que suele derivar el resto: Red Hat, Debian y Slackware y en 2003 nacen algunas muy conocidas hoy día como Ubuntu o Fedora (basada en Red Hat). Fedora es libre totalmente mientras que Red Hat puede contener software gratuito pero no es libre. También, basada en Red Hat está CentOS, dirigida para servidores.

MaX y Zentyal son distribuciones españolas.

1.3. Versiones de Ubuntu y Fedora

Versiones de Ubuntu

Las versiones de Ubuntu se basan en 2 números que representan el año y el mes.

En Ubuntu se mantienen 2 versiones con distinto soporte:

- Versiones con soporte de 9 meses. Sale una cada medio año (abril y octubre). No son versiones estables.
- Versiones con soporte extendido LTS. Más estable y mantiene soporte de actualizaciones durante 5 años. Salen cada 2 años.

Lo habitual es instalar versiones LTS, más estables y con mayor periodo de actualizaciones.

Versiones según servicios que ofrece

las versiones LTS se puede descargar la versión Desktop y la versión Server, en la cual se instalan más servicios.

Los requisitos recomendados para Ubuntu 20.04 LTS son:

- 4 GB RAM.
- Procesador Dual Core de 2 Ghz o superior, solo 64 bits (**ubuntu 20.04** ya no ofrece instalación para equipos de 32 bits).
- 25 GB de espacio libre en disco duro.

Versiones de Fedora

También salen cada medio año. Son números correlativos.

1.4. Entorno gráfico X-Windows

La gran ventaja de los entornos gráficos en GNU-Linux son los entornos gráficos sencillos y amigables.

X-Windows es el sistema gráfico usado por sistemas UNIX. Desarrollado desde mediados de los 80 en el Instituto Tecnológico de Massachusetts (MIT). El grupo de desarrollo inicial ha dado lugar a diversos sucesores: Xconsortium (94-96), The Open Group (97-98), Xfree 86 (92-04), X.org (desde 2004).

X-Windows proporciona una interfaz gráfica (GUI) a Linux. Ofrece elementos de interacción como ventanas, cuadros de diálogo, botones y menús y es quien hace posible que Linux sea un competidor en el mercado de S.O. para PC.

Es una interfaz gráfica completa para Linux y por extensión, para UNIX. Está compuesto por: Servidor X, encargado de mostrar visualmente elementos en pantalla independiente del S.O. y el gestor de ventanas, encargado de la gestión de las ventanas mostradas para las aplicaciones, apariencia, creación, colocación en pantalla, etc. Así, es capaz de distribuir el procesamiento de aplicaciones siguiendo el paradigma cliente-servidor. El servidor provee servicios para acceder a teclado, ratón y pantalla, y clientes son las aplicaciones que toman estos recursos para interactuar con el usuario.

A esta forma de diseño se le llama "en dos partes" y origina diferentes implementaciones de gestores de ventanas. Servidor X es altamente portable y en Ubuntu permite utilizar tres entornos de escritorio GUI (Graphical User Interfaces): GNOME (orientado a la simplicidad, Ubuntu lo usa por defecto), KDE (mayor conjunto de aplicaciones y caracterización por defecto y Xfce (uso con requisitos de hardware bajos). Además se pueden usar otros como Fluxbox, Sugar y LXDE.

2. Instalación de Ubuntu en una máquina virtual

PASO 1. 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.

<https://ubuntu.com/download/alternative-downloads>

BitTorrent

BitTorrent is a peer-to-peer download network that sometimes enables higher download speeds and more reliable downloads of large files. You will need to install a BitTorrent client on your computer in order to enable this download method.

Ubuntu 18.10 	Ubuntu 18.04.1 LTS 	Ubuntu 16.04.5 LTS 	Ubuntu 14.04.5 LTS 
Ubuntu 18.10 Desktop (64-bit)	Ubuntu 18.04.1 Desktop (64-bit)	Ubuntu 16.04.5 Desktop (64-bit)	Ubuntu 14.04.5 Desktop (64-bit)
Ubuntu 18.10 Server (64-bit)	Ubuntu 18.04.1 Server (64-bit)	Ubuntu 16.04.5 Desktop (32-bit)	Ubuntu 14.04.5 Desktop (32-bit)
		Ubuntu 16.04.5 Server (64-bit)	Ubuntu 14.04.5 Server (64-bit)
		Ubuntu 16.04.5 Server (32-bit)	Ubuntu 14.04.5 Server (32-bit)

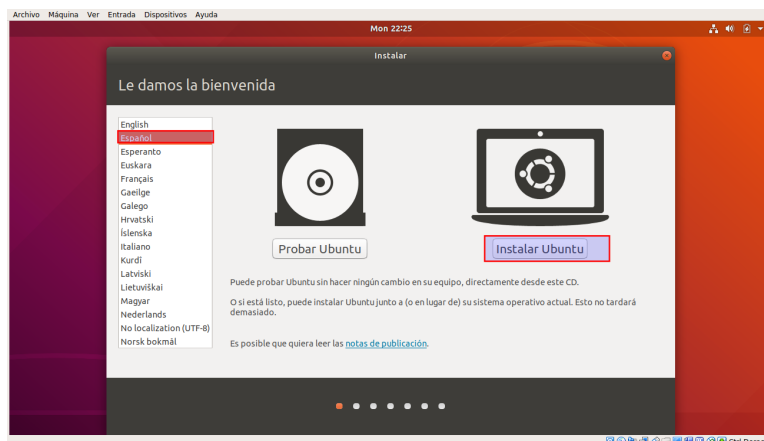
PASO 2. Crear la máquina virtual y montar iso

Estos pasos son iguales, que cuando se instaló la máquina con Windows 10. 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**.

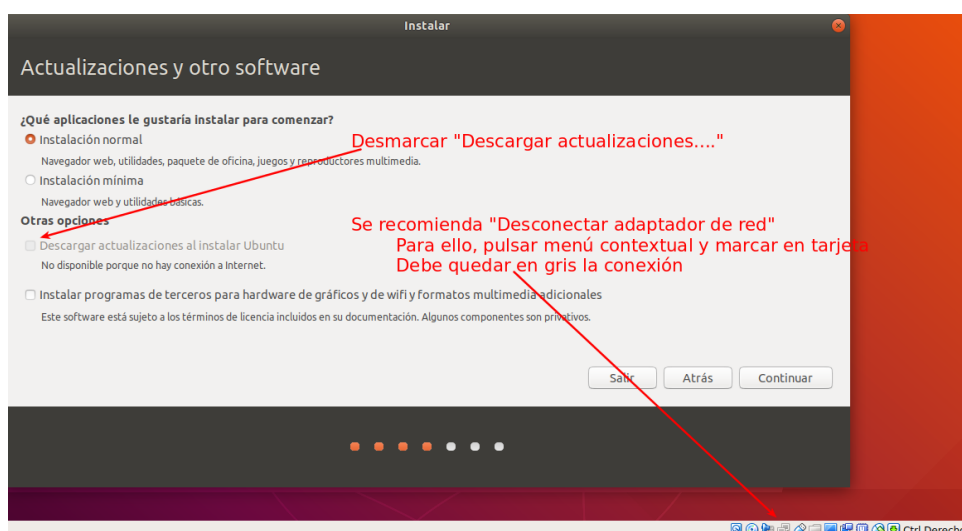


La siguiente pantalla, es para seleccionar idioma del teclado, ya debe aparecer Español. Pulsamos Continuar.

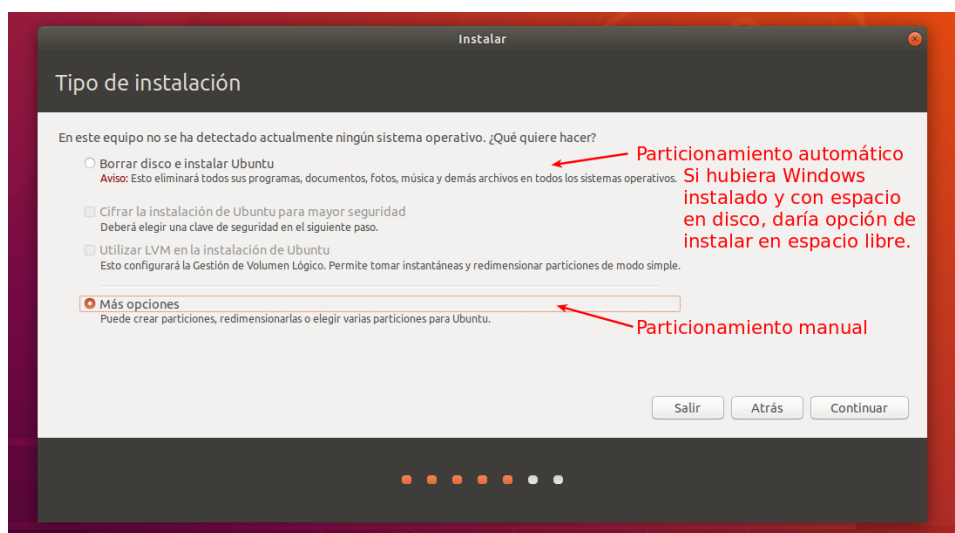
Para iniciar la instalación, el sistema muestra una pantalla preguntando si queremos una instalación normal o mínima. Además pregunta si queremos que descargue actualizaciones. En nuestro caso, realizamos

instalación normal y desmarcamos "Descargar actualizaciones..." para evitar que la instalación se alargue en el tiempo.

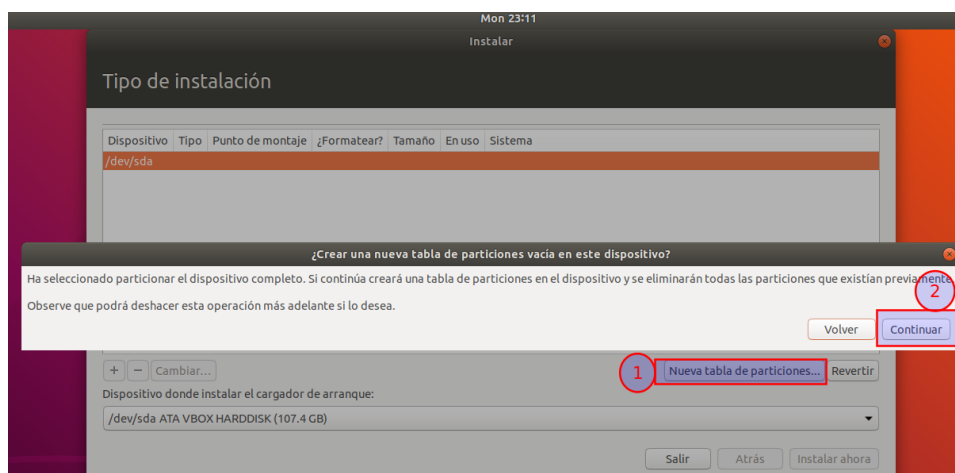
También se recomienda desconectar la tarjeta de red de la máquina virtual (para evitar que el sistema pierda tiempo, intentando conectar la red).



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.



En la siguiente pantalla, debemos pulsar “ **Crear nueva tabla de particiones** y



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

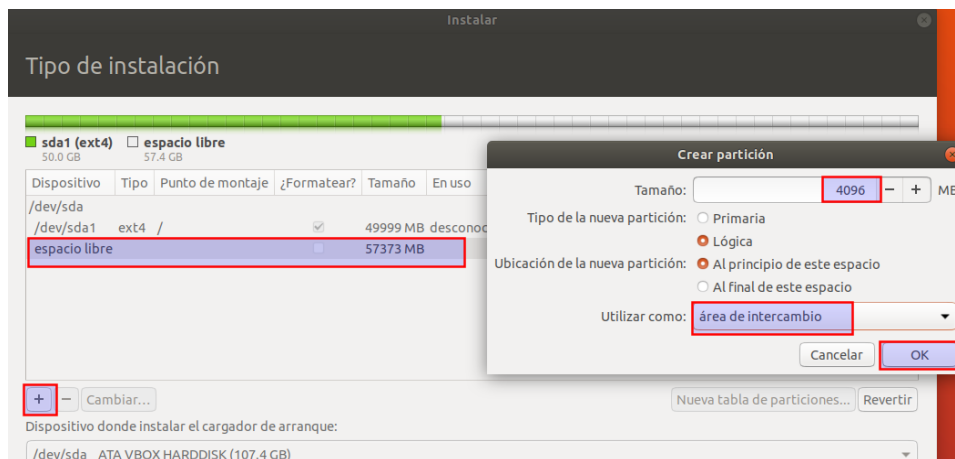
Para crear la **partición de sistema de 50GB** se seguirán los pasos siguientes:

- 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



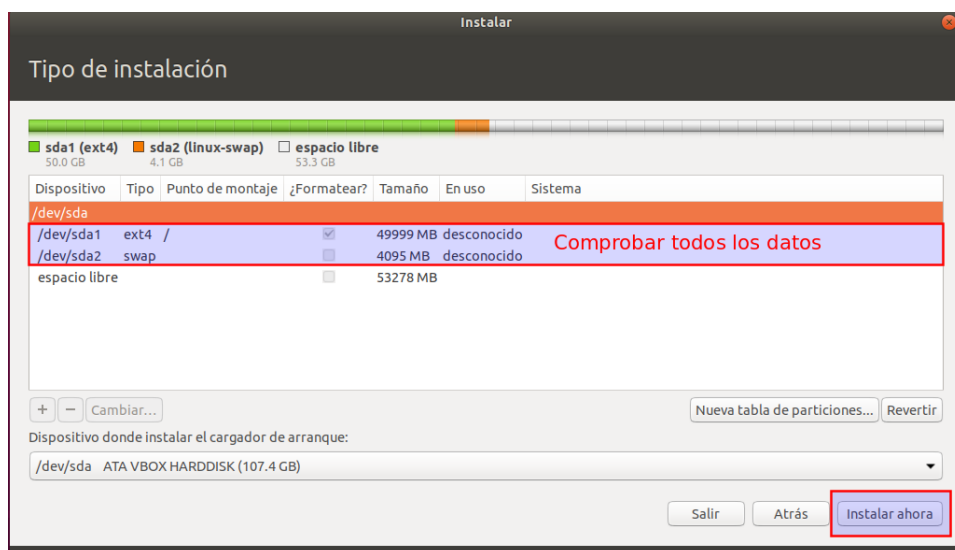
Creamos ahora la **partición swap** o área de intercambio. Como nuestra máquina virtual tiene poca memoria RAM, ponemos como tamaño de la partición el doble de memoria RAM.

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



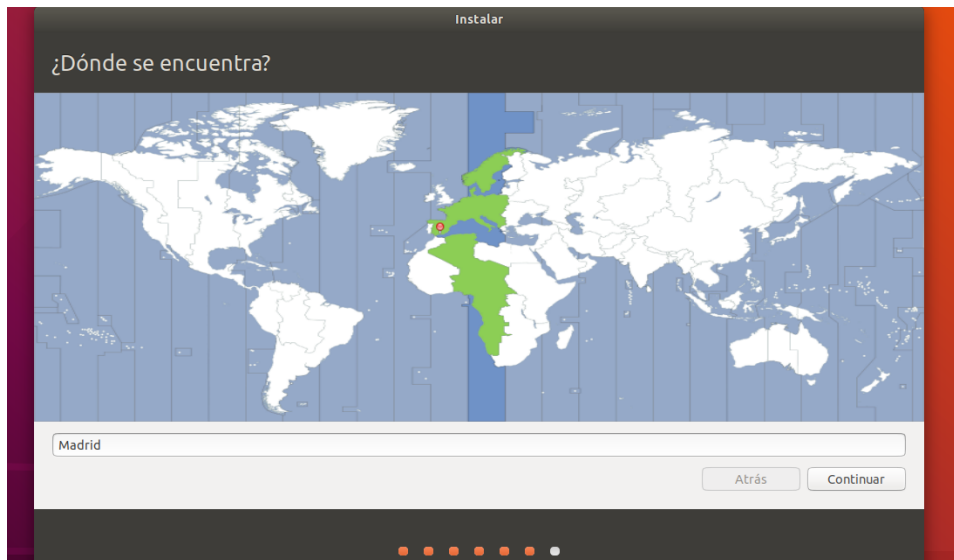
En la siguiente imagen, se ven las 2 particiones creadas. Son particiones primarias, reconocidas por Linux como sda1 y sda2.

Una vez creadas las dos particiones se pulsa **Instalar ahora**

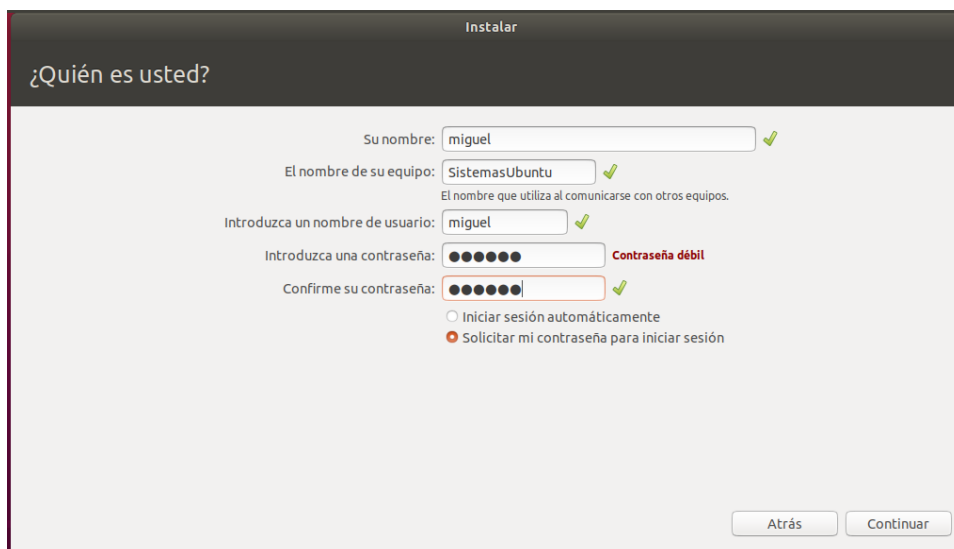


Al pulsar Instalar ahora, nos aparece una ventana informando que si pulsamos Continuar, se guardarán todos los cambios. Se pulsa Continuar.

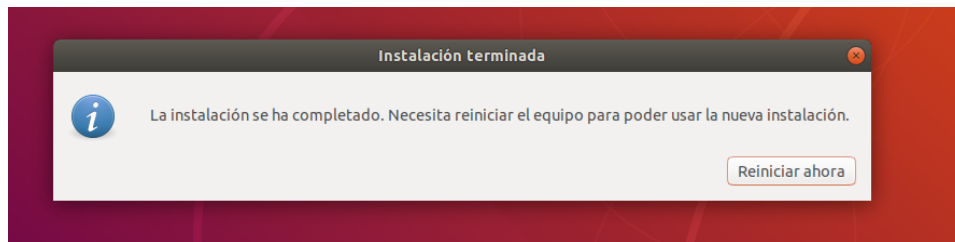
Posteriormente, nos pide seleccionar la zona horaria donde nos encontramos. Pulsar Continuar.



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.



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.

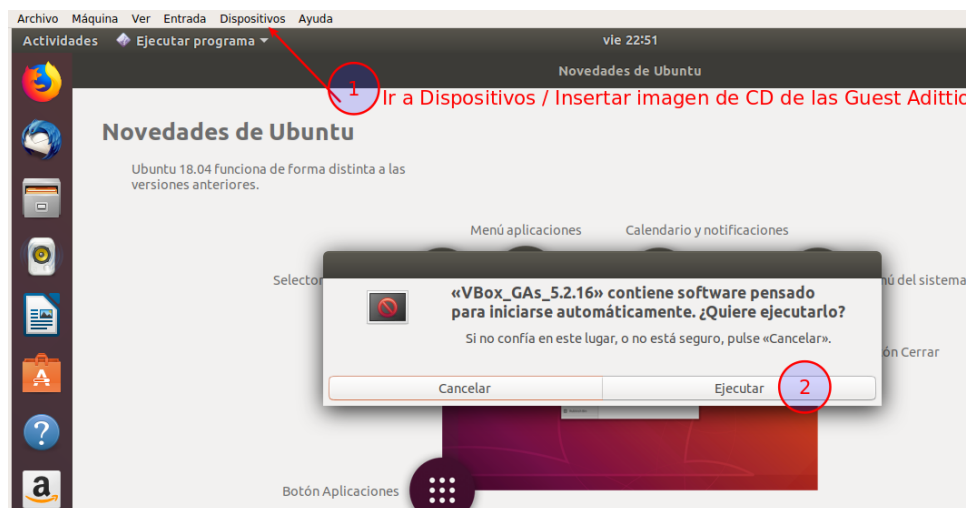


Cuando se está apagando, el sistema da un mensaje en el que hay que pulsar Intro. Este mensaje, sirve para decir que debemos extraer el CD. Esto significa, que al ser una máquina virtual, se desmonta la iso. Pero si estuviéramos instalando Ubuntu en una máquina real, al pulsar Intro se abriría la unidad 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.

PASO 4. Instalar Guest Additions

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



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

3. Primeros pasos en Linux

3.1. Entorno gráfico y directorios de Linux

Interfaz gráfica

Cuando se instala Ubuntu se obtienen un gran número de aplicaciones instaladas.

- El navegador o explorador de archivos se llama Nautilus.
- Para navegar entre distintas ventanas se utiliza Alt+Tab.
- El centro de software de Ubuntu sirve para instalar y desinstalar aplicaciones de forma gráfica.
- Cuando se instala una aplicación, Ubuntu busca el software en páginas web llamadas repositorios Ubuntu.

Diferencias entre Windows y 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 en Linux

- **/bin y /usr/bin:** contiene la mayoría de ficheros ejecutables y comandos. Se llama bin porque representa a los binarios puros. Se llama así a los ejecutables que con ceros y unos solo entienden las máquinas.
- **/etc:** contiene ficheros de configuración del sistema. Suelen ser archivos de texto plano editables.
- **/home:** contiene el directorio home de todos los usuarios salvo del usuario root.
- **/root:** contiene el directorio home del usuario root.
- **/usr:** aquí se instalan las aplicaciones de los usuarios.
- **/tmp y /usr/tmp:** almacena ficheros temporales
- **/boot:** ficheros necesarios para arranque del sistema.
- **/dev:** contiene los archivos de dispositivo (drivers de dispositivos): discos, usb, puertos, terminales... Ejemplo: /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 dispositivos de almacenamiento o unidades lógicas. Si se montan automáticamente se utiliza /media. Manualmente: /mnt.
- **/var:** se guardan archivos de información de sistemas, datos de aplicaciones... También se encuentran aquí los archivos de registro.

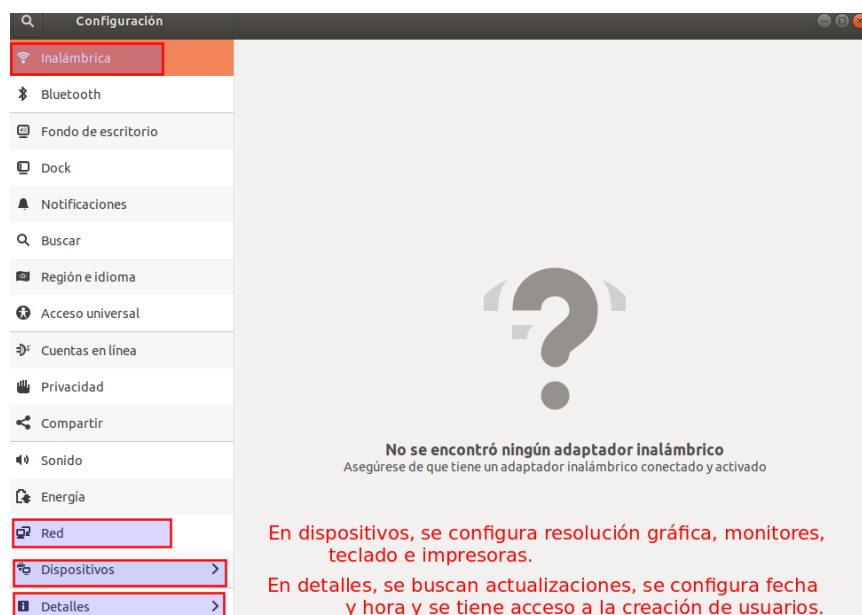
Observaciones

- En el explorador de archivos, "Otras ubicaciones" representa a /. Al abrir otras ubicaciones se puede explorar todo el árbol.
- Solo hay un único árbol raíz, un pendrive se verá en /media/usuario/label_dispositivo.
- Se pueden crear más particiones, utilizando esas particiones para un directorio concreto. Por ejemplo, cuando se instala un servidor suele crear estas 5 particiones: /, swap, /home, /boot, /tmp.

3.2. Configuración de Ubuntu

Toda la configuración se encuentra en la ventana "Configuración" (hacer click en Actividades y escribir "Configuración").

Los más importantes submenús de configuración son: Red, Dispositivos (resolución gráfica, monitor, teclados, impresoras y otros dispositivos y Detalles (creación de usuarios, actualización del sistema.



Al pulsar en detalles podemos ver:



Creación de un nuevo usuario

Hay que desbloquear el programa para poder crear y eliminar usuarios. Una vez desbloqueado pulsamos en "añadir usuario", rellenamos el nombre de usuario, contraseña y pulsamos en añadir. La contraseña debe tener cierta complejidad. Por defecto el usuario es estándar, no puede convertirse a superusuario o root.



4. Terminal de comandos de GNU-Linux

El intérprete de comandos o shell es la interfaz entre el usuario y el S.O., la shell recibe órdenes a través de la línea de comandos, las interpreta, ejecuta y muestra su resultado.

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

Se puede iniciar la aplicación terminal desde la interfaz gráfica. Para ello se pulsa en Actividad y se escribe la palabra terminal.

Además en la mayoría de versiones Linux, aparte de la sesión gráfica se inician varias terminales de texto (tty1, tty2..., tty6. Se muestran pulsando Ctrl+Alt+F1, F2... F6.

En Ubuntu tt1 y tt2 son sesiones gráficas y el resto son terminales de texto. En otros Linux, las terminales cambian.

Sintaxis de comandos

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

Por ejemplo: ls -l -a /home/alumno o ls -la /home/alumno

- ls es el comando que muestra lo que tiene un directorio (siempre en minúsculas).
- Opciones:
 - opción l sirve para que la info de cada archivo y directorio sea larga y extendida.
 - opción a sirve para mostrar directorios ocultos.

- Sirve tanto escribir cada opción con un guion delante, como escribir las opciones juntas precedidas de un guion.
- Parámetros:
 - Donde se va a realizar la acción, en este caso una ruta.

Primeros comandos

passwd usuario

Cambiar contraseña del usuario.

exit

para salir de la sesión. Si tenemos una terminal abierta en sesión gráfica, se cierra la terminal, si estamos en terminal de texto tt1 el sistema finaliza sesión y muestra login para iniciar sesión nuevamente.

man y help

Ayuda de comando, se solicita de dos maneras. Help ofrece ayuda resumida, man ofrece la ayuda completa del manual Linux. Para desplazarte se usan las teclas Repg y AvPg y para salir se pulsa la tecla q (quit).:

1. `passwd —help` (doble guion)
2. `man passwd`

Usuarios de Linux. Trabajar como administrador. Cambios de usuario

Trabajar como administrador

El usuario que se crea cuando instalas linux tiene poderes para convertirse en superusuario o root. Cualquier usuario que se cree posteriormente no tendrá poderes salvo que se configuren.

El primer nombre de usuario creado y root son dos usuarios distintos (en Windows no es así).

sudo comando_a_ejecutar

Sirve para ejecutar un solo comando como root (superusuario o administrador)

La terminal nos pide contraseña y ejecutará el comando con la identidad del root y sus permisos.

sudo su

Es importante ser root solo cuando es necesario, para evitar crear archivos con permisos de root y que luego el usuario no pueda manipularlo.

Cambiar a otro usuario. Comando su

su nombre_usuario

Se solicita password del usuario.

su root

No funciona por defecto en distribuciones Ubuntu, ya que root no tiene contraseña por defecto, para poder habilitarlo se le debe poner contraseña al root con `sudo passwd root`

Varios comandos sencillos

poweroff

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

reboot

Reiniciar el equipo.

who

Devuelve todos los usuarios conectados al equipo, en distintas terminales o en red.

echo

Devuelve mensaje en pantalla.

pwd

Devuelve el directorio actual.

clear

Limpiar la pantalla.

Inicio de sesión del usuario en Linux. Directorio /home

Al iniciar con un usuario en una consola Linux, se accede por defecto a su directorio \$HOME, a ese directorio home solo tiene acceso ese usuario y los administradores. En Ubuntu, por defecto, se permite lectura de los archivos de otros usuarios.

En la mayoría de las shell aparece el símbolo ~ cuando está en su home, en lugar de /home/directorio

Significado de los parámetros del Prompt o Shell del sistema

root@localhost:etc#

root: estamos conectados con el usuario root.

localhost: el nombre del ordenador.

etc: directorio actual, si aparece ~ estamos en el HOME de usuario.

#: indica que el usuario no es administrador, si es administrador se visualiza \$.

4.2. Comandos de directorios

- La utilización de rutas absolutas y relativas son iguales en Win. y Linux.
- Linux utiliza también * y ? con el mismo significado que Windows.
- En Linux un archivo o directorio oculto empieza por el carácter punto.
- En Linux un archivo ejecutable se distingue por sus permisos y no por su extensión.

pwd

Muestra el directorio actual con ruta absoluta.

cd

Cambiar de directorio. Ejemplo:

Ruta relativa:

```
usuario2@SistemasUbuntu:~$ cd /home/usuario1
```

Ruta absoluta:

```
usuario2@SistemasUbuntu:~$ cd ../usuario
```

mkdir

Crear directorios nuevos

rmdir

Borrar directorio, debe estar vacío, para eliminar el directorio con todo su árbol y ficheros se utiliza `rm -rf`

tree

Muestra info jerárquica de un directorio: ficheros y subdirectorios.

ls

Muestra los archivos de un directorio. Sintaxis: ls [laRtr] [ruta]

- **-l** muestra información detallada de cada archivo (propietario, fecha, permisos).
- **-a** sirve para incluir archivos ocultos en el listado, empiezan por punto.
- **-R** muestra información de subdirectorios (árbol entero).
- **-t** muestra archivos ordenados por fecha (t de time).
- **-r** muestra el orden inverso.

Información que devuelve -l. Ejemplo:

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

- **d**: Si el primer carácter es una d, es un directorio, si no es d, es un fichero. En el caso de fichero puede aparecer:
 - **Guion** (archivos normales o regulares)
 - **b**: archivo de bloque, se utiliza en discos duros donde las transferencias de info se realizan en bloque.
 - **c**: archivo de carácter, se usa en dispositivos de carácter a carácter, por ejemplo las terminales tty1, tty2...
- **rw-r-xr-x**: son los permisos de fichero. Si aparece la x que es ejecutable.
- **2**: significa que tiene dos enlaces duros, en Linux hay 2 tipos de enlaces: duros y blandos.
- **alumno**: nombre del usuario propietario del fichero.
- **alumno**: grupo propietario del fichero (todos los usuarios pertenecen a un grupo, cuando se crea un usuario se crea dentro de un grupo que lleva el nombre del usuario por defecto y seguidamente se introduce ese alumno en el grupo).
- **4096**: tamaño del fichero en bytes.
- **2012-04-15 23:43**: fecha y hora de la creación o modificación.
- **Descargas**: nombre del directorio o fichero.

4.3. Comandos de ficheros

Editor de texto plano:

En Ubuntu vienen instalados nano y vi para editores de texto plano en terminal (en entorno gráfico incorpora gedit).

Utilizando nano, podemos ejecutarlo escribiendo: **nano archivo.txt** si existe el archivo se muestra su contenido, si no existía se muestra vacío. En ambos casos podemos añadir y modificar. El menú inferior muestra las opciones que se pueden realizar con el archivo.



touch

Crear un fichero vacío y si ya existía se modifica su hora de modificación.

type

Muestra la ruta donde se encuentra el ejecutable de un comando. Ejemplo:

```
alumno@sistemasUbuntu:~$ type touch
```

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

cp

Sintaxis: cp [-R] origen destino

Copia archivos o directorios, se pueden copiar con otro nombre, la R copia toda la estructura de subdirectorios y ficheros.

Ejemplos:

- Estando en \$HOME, copiar "archivo.txt" en el \$HOME de miguel con el nombre copia.txt.
 - Ruta relativa:

```
alumno@sistemasUbuntu:~$ cp archivo.txt ../miguel/copia.txt
```

- Ruta absoluta:

```
alumno@sistemasUbuntu:~$ cp /home/alumno/archivo.txt /home/miguel/copia.txt
```

- Copiar todos los archivos .txt de home de usuario 1 al subdirectorío datos con extensión .dat

```
cp /home/usuario1/*.txt /home/usuario1/datos/*.dat
```

- Copiar la estructura del directorio home de usuario1 al home de usuario2 con ruta absoluta.

```
cp -R /home/usuario1 /home/usuario2
```

mv

Sintaxis: mv origen destino

Mover archivos o directorios, se puede cambiar el nombre a la vez, también sirve para renombrar ficheros o directorios.

rm

Sintaxis: rm [irf] [ruta]/fichero

Borrar ficheros

- -i pide confirmación antes de borrar cada archivo.
- -r borra todo lo que haya en directorios enteros.
- -f fuerza el borrado de archivos sin preguntar.

Direccionamientos >, >>, 2>, y 2>>

Los operadores de direccionamiento se utilizan igual que en windows.

- El operador > sobrescribe el archivo.
- El operador >> añade a lo que ya tuviera el archivo.
- 2> y 2>> redireccionan el buffer de errores.

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 22 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/carpeta
La ruta actual es ~/carpeta es decir, /home/alumno/carpeta
alumno@SistemasUbuntu:~/carpeta$ cd /home/miguel
Se cambia al directorio $HOME de miguel
alumno@SistemasUbuntu:/home/miguel$ rm -rf ../alumno/carpeta
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

Los administradores son los únicos que pueden instalar aplicaciones. Para ello se utiliza el gestor de paquetes.

Los gestores de paquetes buscan de forma automática en los repositorios de software, que son servidores web que los facilitan. Además, si alguna aplicación pide instalar otras previas, las busca y las instala.

Distribuciones basadas en Debian (paquetes debian o .deb, gestores apt-get y aptitude y apt

Ubuntu está basada en Debian y debemos tener presente la distribución con la que estamos trabajando cuando se instalan paquetes.

Anteriormente se usaba apt-get, el cual se había mejorado con aptitude, pero en las distribuciones más modernas se utiliza apt

Sintaxis de apt

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

Cambio de versión del Sistema Operativo:

Para cambiar a un S.O. más moderno:

`$ sudo apt update` (para actualizar repositorios)

`$ sudo apt upgrade` (para actualizar S.O. y aplicaciones).

`$ sudo update-manager -d`

Instalación gráfica:

En Ubuntu tenemos icono directo al centro de software de Ubuntu.

Se basa en apt y podemos instalarlas igual en ambos sitios.

Repositorios de software

Las direcciones de repositorios se encuentran configuradas en `/etc/apt/sources.list`

Algunas veces tenemos que añadir el repositorio antes de instalar alguna app.

Distribuciones basadas en Red Hat. Paquetes Red Hat o .rpm. Herramienta yum

En distribuciones Red Hat y derivadas se utiliza el gestor YUM, que funciona igual que apt

Sintaxis de yum

Instalar: `#yum install nombre_paquete`

Desinstalar `#yum remove nombre_paquete`

Buscar un paquete `#yum search nombre_paquete`

Actualización de todos los paquetes: `#yum update`

Instalar sin gestores de instalación

Se puede realizar instalación de paquetes .deb o .rpm sin utilizar gestores, pero suele ser más pesada. Al instalar un paquete busca las dependencias. Se debe primero instalar las dependencias y luego el paquete que queremos y eso suele ser tedioso.

comando dpkg

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

desinstalar paquete db: `#dpkg -r paquete.deb`

comando rpm

instalar paquete rpm `#rpm -ivh paquete.rpm`

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

Instalar paquetes de código fuente en cualquier Linux

Se puede bajar de internet el código fuente en lenguaje C e instalarlo, para ello se debe utilizar `tar -xvzf` para desempaquetar y descomprimir, después cambiamos a la carpeta descomprimida y ejecutamos:

`#./configure`

`#make`

`#make install`

Mapa conceptual

