

TEMA 9

INDICE

1.- Introducción.....	2
1.1.- Distribuciones.....	2
1.2.- Licencias de software.....	4
2.- Instalación.....	5
Pasos documentados para realizar la instalación de Ubuntu.....	5
INSTALACIÓN DE UBUNTU.....	5
3.- X-Windows.....	8
4.- Primeros pasos.....	9
4.1.- Intérprete de comandos.....	9
4.2.- Estructura de directorios.....	11
4.3.- Instalar y quitar componentes.....	12
4.3.1.- Synaptic.....	12
4.3.2.- apt-get.....	12
4.3.3.- Aptitude.....	13
4.3.4.- Instalación manual.....	13
4.4.- Webmin.....	14

Instalación y configuración (Linux I).

1.- Introducción.

Linux fue concebido por el finlandés Linus Torvalds, estudiante de la Universidad de Helsinki, quien comenzó trabajando sobre el código fuente de Minix (un pequeño UNIX desarrollado por Andy Tanenbaum) para lograr un Unix mínimo, capaz de ejecutar al menos un shell y un compilador. Primero fue la versión 0.02 ya que la 0.01 nunca llegó a ser compilada con éxito.

Linux fue liberado en Internet y la respuesta de los programadores y usuarios de UNIX 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 del Kernel, la versión 1.0. De allí en adelante, Linux fue evolucionando a un ritmo vertiginoso hasta convertirse en un fuerte rival de los sistemas operativos comerciales.

Desde su aparición los sistemas GNU/Linux han ido evolucionando y mejorando sus prestaciones drásticamente. Hoy en día puede encontrar los sistemas GNU/Linux en multitud de sistemas: desde grandes servidores hasta pequeños equipos domésticos como teléfonos móviles.

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.

1.1.- Distribuciones.

El kernel del sistema operativo Linux fue desarrollado por Linus Torvalds y licenciado bajo GPL. Pero un sistema GNU/Linux no es sólo su núcleo ya que existen hoy en día un gran número de aplicaciones desarrolladas también bajo licencias libres que permiten que los sistemas GNU/Linux tengan una gran versatilidad y funcionalidad.

La Fundación Linux realizó un interesante cálculo acerca de cuál sería la cifra que una empresa debería abonar para desarrollar desde cero la distribución Fedora 9. El trabajo concluyó que el coste total de desarrollo de Fedora 9 tiene un valor de 10.800 millones de dólares.

Existen muchas aplicaciones desarrolladas bajo licencia libre pero, sin duda alguna, las más importantes son:

OpenOffice	http://www.openoffice.org/
Apache	http://www.apache.org/
Firefox	http://www.mozilla.org/es-ES/firefox/new/

Estas son solamente unos ejemplos, pero existen aplicaciones libres para cualquier uso que imagines.

Puesto que el kernel y las aplicaciones son libres entonces puedes crear tu propio sistema operativo con los programas que más te gusten y crear tu propia distribución; lógicamente nuestra distribución también tendrá que ser libre. Una distribución es una agrupación de un conjunto de programas, imágenes, temas de escritorio, etcétera.

Si deseas hacer tu propia distribución GNU/Linux es recomendable que accedas a www.instalinux.com donde a través de un asistente podrás personalizar y crear tu propia distribución.

www.instalinux.com

Existen muchas iniciativas tanto empresariales como gubernamentales para crear su propia distribución. Un claro ejemplo lo puedes encontrar en las distribuciones **Guadalinex** (de la Junta de Andalucía), **gnuLinux** (de Extremadura), **trixbox** (distribución empresarial de telefonía IP), etcétera.

A partir de la libertad de los usuarios, empresas y organismos para personalizar su propia distribución han surgido una gran cantidad de distribuciones que nacen, evolucionan, derivan en otras distribuciones y como no, algunas mueren.

En la siguiente tabla puedes ver algunas de las distribuciones más utilizadas actualmente.

Distribuciones más utilizadas.	
Debian.	http://www.debian.org/
Gentoo Linux	http://www.gentoo.org/
OpenSuse	http://www.opensuse.org/
Ubuntu	http://www.ubuntu.com/
Puppy Linux	http://www.puppylinux.org/
Linux Mint	http://www.linuxmint.com/
CentOS	http://www.centos.org/
Kubuntu	http://www.kubuntu.org/
Fedora	http://fedoraproject.org/
Mandriva Linux	http://www.mandriva.com/
Slackware	http://www.slackware.com/
Sabayon	http://www.sabayon.org/
SLAX	http://www.slax.org/
PC LinuxOS	http://pclinuxos.com/
FreeBSD	http://www.freebsd.org/

En la web distrowatch.com puede ver y descargar cualquier distribución.

<http://distrowatch.com/>

En el siguiente enlace puede ver un mapa de distribuciones GNU/Linux que abarca su evolución, derivaciones y bifurcaciones en el tiempo partiendo de las distribuciones "matrices" como son Debian, Slackware y RedHat.

<http://futurist.se/gldt>

1.2.- Licencias de software.

Sin duda alguna no se puede hablar de los sistemas GNU/Linux sin mencionar las licencias de software libre, germen de todo el desarrollo de los sistemas GNU/Linux.

La Licencia Pública General de GNU, más conocida como GNU GPL, es una licencia creada por la Free Software Foundation en 1989. Su principal objetivo es garantizar la libertad de compartir y modificar el software. El término libre (free en inglés) se refiere a la libertad de poder modificar y distribuir el software, no a su precio.

Al desarrollar un programa, escribir un artículo o crear cualquier obra que se desea distribuir libremente, puedes optar por dos caminos diferentes: **dominio público** o **licencia libre**. Lógicamente, las dos opciones permiten liberar el software garantizando la autoría de la obra.

Si creas una aplicación para **dominio público**, el software estará disponible de forma gratuita para otras personas pero el código fuente no estará disponible. Por el contrario, si eliges una **licencia libre**, entonces el proceso es mucho más enriquecedor ya que pone disponible el código fuente por lo que permite que otros programadores puedan modificar, mejorar o adaptar nuestro software a sus necesidades.

Cuando se licencia un software bajo GPL se permiten los siguientes grados de libertades:

- ✓ **Libertad 0.** Ejecutar el programa sea cuál sea nuestro propósito.
- ✓ **Libertad 1.** Estudiar el funcionamiento del programa y adaptarlo a tus necesidades.
- ✓ **Libertad 2.** Redistribuir copias.
- ✓ **Libertad 3.** Mejorar el programa y luego distribuirlo.

Lógicamente, los términos de la licencia permiten la libertad de poder utilizar y adaptar el software a sus necesidades pero el resultado debe seguir siendo libre. Este punto es muy importante ya que permite que no se "rompa" la cadena de software libre.

Actualmente se encuentra en vigor GPL versión 3 que fue publicada en 2007 y define aspectos de una forma mucho más precisa sobre el uso del software licenciado GPL.

Si quieres saber más sobre el software libre te recomiendo que leas el libro "Software libre para una sociedad libre" del gurú del software libre Richard M. Stallman.

http://www.gnu.org/philosophy/fsfs/free_software.es.pdf

Las licencias Creative Commons están inspiradas en la licencia GPL, pero están destinadas a facilitar el uso y distribución de los contenidos garantizando la autoría de la obra.

Un enlace interesante es el artículo sobre el software libre de la wikipedia, te lo recomendamos para conocer más detalles sobre el software libre:

http://es.wikipedia.org/wiki/Software_libre

2.- Instalación.

Para aprender a instalar un sistema GNU/Linux, usaremos Ubuntu por su amplia difusión, facilidad a la hora de instalarlo en cualquier ordenador, y por sus posibilidades para instalar software adicional

El proceso de instalación del sistema GNU/Linux resulta bastante sencillo gracias al asistente que te guía durante todo el proceso de instalación. Antes de iniciar la instalación necesita tener en cuenta el uso que le vas a dar al sistema ya que de ello dependerá mucho el hardware del equipo. Como regla general necesitas un equipo con al menos 512 Megabytes de RAM y unos 5 Gigabytes de disco duro. Aunque todo depende del uso que quiera darle al sistema.

En Ubuntu existen tres versiones: **Desktop** (para equipos de escritorio), **Netbook** (para portátiles) y **Server** (para servidores). Además, para cada tipo puedes utilizar la versión de **32 bits** y de **64 bits**. Como el objetivo de la unidad es aprender a instalar un sistema operativo GNU/Linux, realizaremos la instalación de la versión **Desktop**.

En la página oficial de Ubuntu puedes descargar la versión Ubuntu Desktop: <http://www.ubuntu.com/>

A continuación vas a aprender a instalar Ubuntu Desktop en nuestro sistema.

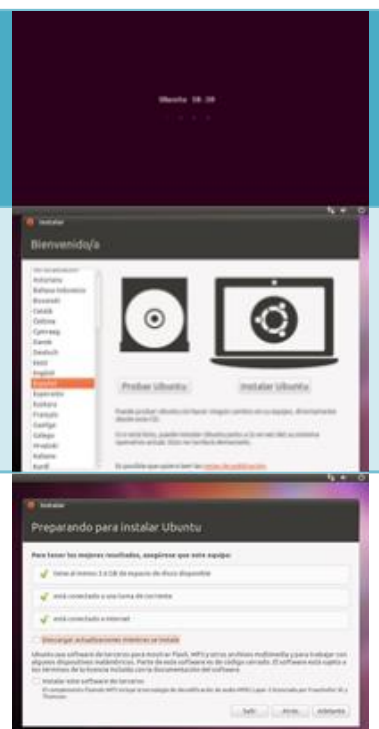
Pasos documentados para realizar la instalación de Ubuntu.

INSTALACIÓN DE UBUNTU

Para iniciar la instalación debes iniciar el equipo con el CD de la distribución. Si el equipo no muestra la pantalla de inicio puedes entrar en la BIOS del equipo y configurarla para que arranque el sistema directamente desde CD.

Una vez iniciado el sistema, aparece la pantalla de bienvenida donde debes seleccionar el idioma **Español** e indicar si quieres **Probar Ubuntu** o si deseas **Instalar Ubuntu**. En nuestro caso vamos a realizar la instalación de Ubuntu.

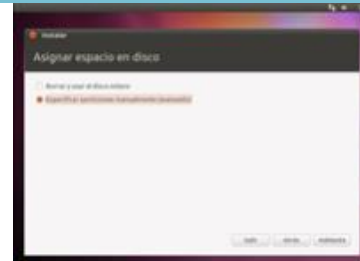
Para iniciar la instalación, el sistema muestra una pantalla con los requisitos que debe tener el sistema. Tal y como muestra la figura estos requisitos son: tener al menos 2 Gigabytes de espacio de disco duro, tener conexión a Internet y en el caso de tratarse de un portátil, estar conectado a una toma de corriente. Además, podemos indicarle al sistema que descargue automáticamente las actualizaciones del sistema mientras que se realiza la instalación. En nuestro caso, pulsamos **Adelante**.



El siguiente paso es realizar el particionamiento del sistema. Para ello puedes seleccionar la opción **Borrar y usar el disco entero** en el caso de querer hacer el particionamiento automático, o puedes seleccionar **Especificar particiones manualmente**. Para no perder la información del sistema vamos a seleccionar la opción **Especificar manualmente** y pulsar **Adelante**.

Para realizar el particionamiento del sistema debes tener en cuenta que hay que crear la partición donde se guardan los datos (**/**) y la partición swap. La partición **swap** o área de intercambio, debe tener al menos el doble del tamaño de la memoria RAM del sistema.

Para empezar el proceso de particionamiento pulsa en **Nueva tabla de particiones**.



Crea la partición del sistema de tipo **ext4** con al menos 2 Gigabytes con punto de montaje **/** y pulsa **Aceptar**.

Crea la **partición de swap** (área de intercambio) con el doble de tamaño que la memoria RAM del sistema.



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

Selecciona la **zona horaria** donde te encuentras y pulsa **Adelante**.



Indica la **distribución de tu teclado** y pulse **Adelante**.

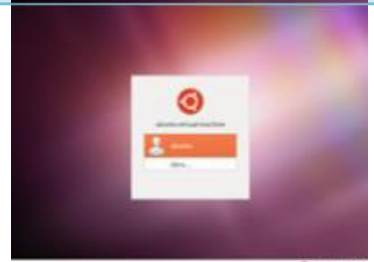
A continuación se va a crear una cuenta de usuario. Para ello, introduce tu **nombre**, el **nombre del equipo**, el **nombre de usuario** y tu **contraseña**. Y para acabar pulsa **Continuar**.



Para finalizar la instalación pulsa **Reiniciar Ahora**.



Una vez reiniciado el equipo, aparece la pantalla de inicio de sesión. Una vez introducido el nombre de usuario y contraseña ya puedes empezar a utilizar el sistema.



Para realizar la instalación de Ubuntu puedes utilizar una máquina virtual.

Si lo deseas, puedes grabar la imagen ISO en un USB y realizar la instalación del sistema más rápido. Para pasar una imagen ISO a un USB puedes utilizar Unetbootin.

<http://unetbootin.sourceforge.net/>

3.- X-Windows.

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**.

Actualmente **Unity**.

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

Entornos gráficos más utilizados.	
ENTORNOS GRÁFICOS	
KDE	http://www.kde.org/
GNOME	http://www.gnome.org/
Xfce	http://www.xfce.org/



En la dirección <http://elblogdeliher.com/que-entorno-de-escritorio-elegir-en-ubuntu/> podrás investigar entornos de escritorio

4.- Primeros pasos.

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.

4.1.- Intérprete de comandos.

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.

Para iniciar el intérprete de comandos tienes que ir al menú **Aplicaciones / Accesorios**, ejecutar la herramienta **Terminal** y aparecerá el terminal del sistema.

Una vez que accedes al sistema se muestra un prompt con el siguiente aspecto o parecido:

```
usuario@ubuntu-virtual-machine:~$
```

donde **usuario** es el nombre del usuario que está utilizando el sistema, **@ubuntu-virtual-machine** indica el nombre del equipo. A continuación, se muestra el directorio en el que se encuentra. En el caso de que se encuentre el carácter **~** es porque está en el directorio **home**. Por último, el símbolo **\$** o **#** indica si es un usuario normal (**\$**) o es el administrador del sistema (**#**).

El usuario **root** es el administrador del sistema y puede realizar cualquier tarea de administración. En algunas distribuciones puede acceder directamente al sistema como usuario **root**, pero otras distribuciones, como Ubuntu, te obligan a acceder al sistema con un usuario sin privilegios de administrador y luego cambiar de usuario.

Si deseas ejecutar una tarea de forma puntual como **root** puedes utilizar el comando **sudo** de la siguiente forma:

```
$ sudo <comando>
```

Si necesita ejecutar múltiples tareas puedes obtener un shell de **root** ejecutando **sudo bash** o **su**:

```
$ sudo bash
#
```

Además, si lo deseas, puedes activar la cuenta de **root** al establecer su contraseña:

```
$ sudo passwd root
```

Aunque a lo largo del curso aprenderá a utilizar el shell del sistema, a continuación puedes ver los comando más utilizados en los sistemas GNU/Linux.

A continuación mostramos información sobre los comandos más utilizados en GNU/Linux.

Manejo de archivos	
cd	cambia el directorio
compress	comprime archivos en formato .Z
cp	copia un archivo
chmod	cambia los permisos de un archivo o directorio
chown	cambia el propietario del archivo o directorio
df	muestra el espacio libre en disco
du	muestra el espacio en disco utilizado
fdformat	formatea un disquete
fdisk	particiona unidades
file	determina el tipo de archivo a través del análisis parcial de su contenido
find	encuentra un archivo
fsck	chequea el sistema de archivos
gzip	descomprime un archivo en formato GZip
ln	crea un enlace simbólico o físico depende la opción
ls	sirve para listar el contenido de un directorio
mkdir	crea un directorio
mkfs	crea un nuevo sistema de archivos
mkswap	crea un espacio de intercambio
more	exhibe el contenido de un archivo
mount	monta una unidad o partición en el sistema de archivos
mv	mueve un archivo y se utiliza para renombrar
pwd	devuelve la cadena correspondiente al directorio actual
rm	borra un archivo
rmdir	borra un directorio
swapon	activa el espacio de intercambio
swapoff	desactiva el espacio de intercambio
tar	empaqueta o desempaqueta un archivo en formato .tar
type	muestra la ubicación de un archivo señalando su "path"
umount	desmonta una unidad montada
Manejo de procesos	
free	muestra la memoria libre y utilizada
halt	apaga la maquina
kill	mata un proceso señalando el numero de este
ldd	muestra las librerías necesarias para ejecutar un proceso
ps	muestra todos los procesos q se están ejecutando en el sistema
pstree	lo mismo que ps pero en forma de árbol
reboot	reinicia el sistema

shutdown	cierra el sistema
top	monitorea procesos y el estado del sistema
uname	muestra información del sistema
Manejo de usuarios	
adduser	crea una cuenta de usuario
chsh	cambia la shell de un usuario
groups	muestra el listado de grupos de usuarios del sistema
id	muestra la información de usuario y grupo de un determinado usuario
logout	para salir del sistema y permitir el ingreso a otro usuario
passwd	cambia el password de un determinado usuario
su	le da privilegios de root a un usuario
talk	permite chatear con otros usuarios
users	lista los usuarios conectados al sistema
who	muestra información de los usuarios conectados al sistema
whoami	muestra información nuestra
Otros	
cal	muestra el calendario
date	muestra el día y la hora
info	muestra la ayuda de un comando
man	muestra las paginas del manual de un comando
startx	para iniciar XWindow
Comandos para red	
ping	el tan famoso ping que manda paquetes esperando una respuesta
ifconfig	para ver las placas de red
netstat	testeo de red
host "destino"	muestra ip de "destino"
nmap "ip de destino"	pequeño programa para analizar ip's o rangos de ellas

4.2.- Estructura de directorios.

GNU/Linux, al igual que UNIX, organiza la información del sistema en una estructura de árbol jerárquico de directorios compuesta de ficheros. Esta estructura se forma mediante un sistema de ficheros raíz (file system root) y un conjunto de sistemas de ficheros montables.

Un sistema de ficheros, o file system, es una estructura de directorios completa. Para poder utilizar un sistema de ficheros hay que montarlo; o sea, enlazarlo a la estructura de directorios ya existente. Los sistemas de ficheros se montan automáticamente cada vez que se inicia el sistema operativo. Cuando un usuario se conecta al sistema, se encuentra un único árbol de directorios formado por los distintos sistemas de ficheros que se encuentran montados en ese instante.

4.3.- Instalar y quitar componentes.

En GNU/Linux se puede realizar la instalación de una aplicación directamente a partir del código fuente o a través de la aplicación compilada (paquete). A las aplicaciones pre empaquetadas se le denomina paquete y contienen los binarios, los archivos complementarios y archivos de configuración para poder ejecutarse.

Para facilitar el proceso de instalación se utilizan gestores de paquetes que facilitan la administración de los paquetes. A continuación, vas a aprender a instalar aplicaciones de todas las formas posibles: mediante *x-Windows*, mediante `apt-get`, directamente instalando paquetes `deb` o a partir del código fuente. Las dos primeras formas son las más fáciles de utilizar y por lo tanto las recomendadas.

Versiones actuales dentro de *x-Windows* tenemos el **Centro de Gestión del Software**.

4.3.1.- Synaptic.

Synaptic es una herramienta de *x-Windows* que facilita las tareas de instalación y eliminación de software. Para utilizar la herramienta ejecuta *Gestor de paquetes Synaptic* que se encuentra en el submenú *Administración* dentro de *Sistema*.

Una vez iniciada la herramienta (véase la figura adjunta) el menú de la izquierda muestra las diferentes categorías de aplicaciones. Si pulsas en una categoría aparecen sus diferentes aplicaciones. Si una aplicación ya se encuentra instalada en el sistema su campo de selección se encuentra activo. Si deseas instalar o desinstalar una aplicación sólo debes seleccionar o deseleccionar la aplicación y pulsar el botón *Aplicar*.



Se pueden definir repositorios adicionales para aumentar la disponibilidad de software del sistema. Un repositorio es una ubicación de red que almacena paquetes de software junto a los metadatos que los describe. Los repositorios también se utilizarán con el comando `apt-get` que veremos más adelante.

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

<http://www.ubuntu-guia.com/2009/06/como-instalar-paquetes-y-programas-en.html>

4.3.2.- apt-get.

apt-get permite instalar o desinstalar por línea de comandos cualquier paquete. Para empezar, `apt-get` utiliza una serie de repositorios que se encuentran en el fichero `/etc/apt/sources.list`. Si lo deseas, puede modificar los repositorios del sistema y actualizar el sistema ejecutando:

```
# apt-get update
```

A continuación, se van a ver los procedimientos más utilizados

- ✓ **Actualizar el sistema.** Permite actualizar el sistema con todas las dependencias. Se utiliza:

```
# apt-get upgrade
```

- ✓ **Búsquedas.** Permite localizar un paquete o término en alguno de los repositorios. Se ejecuta:

```
# apt-cache search <nombre>
```

donde *nombre* indica el nombre del paquete que desea buscar.

- ✓ **Consulta de información.** Permite consultar información de un paquete.

```
# apt-cache show <paquete>
```

Por ejemplo si quiere información sobre el servidor web ejecute:

```
# apt-cache show apache2
```

- ✓ **Instalación de paquetes.** Permite realizar la instalación de paquetes con la resolución automática de dependencias.

```
# apt-get install <paquete>
```

Por ejemplo si desea instalar el servidor web ejecute:

```
# apt-get install apache2
```

- ✓ **Desinstalar un paquete.** Para desinstalar un paquete hay que ejecutar:

```
# apt-get remove <paquete>
```

Por ejemplo si desea desinstalar el servidor web ejecute:

```
# apt-get remove apache2
```



A la hora de buscar o instalar un programa puede utilizar el carácter ***** para indicar cualquier carácter. Por ejemplo si desea instalar cualquier aplicación que empiece por php entonces ejecuta `apt-get install php-*`.

4.3.3.- Aptitude.

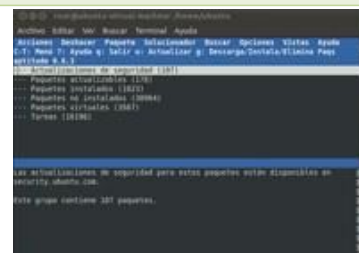
Aptitude es un gestor de paquetes por línea de comandos muy cómodo y sencillo de utilizar. Para poder utilizarlo antes necesitas instalarlo.

```
root@ubuntu ~ # apt-get install aptitude
```

Ejecutas en el terminal como *root* el comando:

```
root@ubuntu ~ # aptitude
```

Una vez ejecutada la herramienta puedes acceder a las diferentes categorías e instalar o desinstalar el software.



4.3.4.- Instalación manual.

Si lo deseas puedes realizar la instalación o desinstalación directa de un paquete. Para realizar la instalación debes descargar previamente el paquete y ejecutar:

```
# dpkg -i nombre_paquete
```

Si por el contrario deseas eliminar un paquete, primero debes conocer su nombre exacto. Para ello debes ejecutar:

```
# dpkg-query -s nombre
```

Una vez que conoce el nombre exacto se realiza la instalación ejecutando:

```
# dpkg -r nombre_completo
```

A veces se encuentran aplicaciones que no proporcionan paquetes de instalación, y hay que compilar a partir del código fuente. Para ello, lo primero que debe realizar es instalar las herramientas de compilación ejecutando:

```
# apt-get install build-essential
```

Además, puedes realizar la instalación directamente desde el código fuente, es posible que surja algún problema de dependencias. Si sucede esto, entonces debe resolver la dependencia y continuar con el proceso de instalación.

En general, los pasos a seguir para compilar una aplicación son los siguientes:

- ✓ Descarga el código fuente.
- ✓ Descomprime el código, generalmente está empaquetado con tar y comprimido con gzip (*.tar.gz o *.tgz) o bzip2 (*.tar.bz2).
- ✓ Accede a la carpeta creada al descomprimir el código.
- ✓ Ejecuta el script `./configure` que permite comprobar las características del sistema que afectan a la compilación y crear el archivo makefile.
- ✓ Compila el código ejecutando el comando `make`.
- ✓ Instala la aplicación en el sistema ejecutando `make install`. Si desea desinstalar la aplicación entonces ejecuta `make clean`.

4.4.- Webmin.

Webmin es una interfaz web que permite administrar el sistema de una forma cómoda y sencilla a través de cualquier equipo utilizando un navegador web. Por ejemplo, webmin permite administrar los usuarios del sistema, rendimiento del equipo, copias de seguridad, etc.

La página web de webmin es (en inglés):

<http://www.webmin.com/>

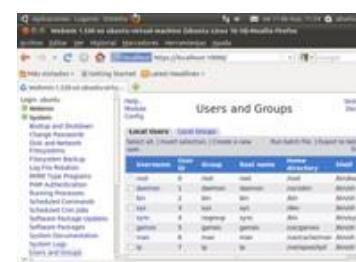
El proceso de instalación de webmin es muy sencillo, ya que una vez descargado el paquete de la web oficial, debes ejecutar el comando:

```
# dpkg -i webmin-1.530_all.deb
```

o directamente pulsar dos veces sobre el paquete para realizar la instalación de forma gráfica.

Una vez finalizado el proceso de instalación el sistema indica que hay que acceder a webmin a través de la dirección `https://localhost:10000`. Al acceder a webmin debes autenticarse en el sistema, por lo que debes introducir como nombre de usuario `root` y su contraseña.

Una vez dentro en la página principal el sistema muestra un resumen del sistema y desde el menú de la derecha puede acceder a las diferentes herramientas de administración del sistema. Por ejemplo, a continuación se muestra la herramienta de `Users and Groups` que permite administrar los usuarios y grupos del sistema.



Comandos más comunes DOS vs Linux

DOS	Comando Linux
cd directorio	cd directorio
dir, dir/w	ls, ls -l
chdir (directorio actual)	pwd
del (borra un archivo)	rm
deltree (borra un directorio y su contenido)	rm -r
copy	cp
xcopy (copia todo el contenido de un directorio)	cp -R
rename, move	mv
type (imprime el contenido de un archivo a la pantalla)	cat
help, [comando] /?	man
cls (limpia la pantalla)	clear
find (busca por una palabra(s) en un determinado archivo)	grep
edit nombre-de-archivo	gedit nombre-de-archivo
mem (muestra la memoria disponible)	free, top
scandisk	fsck
pkzip (crea un paquete de archivos)	tar, utilizado en conjunto con gzip para compresión
ipconfig (visualiza dirección IP y configuración de red)	ifconfig
route print (muestra tablas de ruteo)	route -n