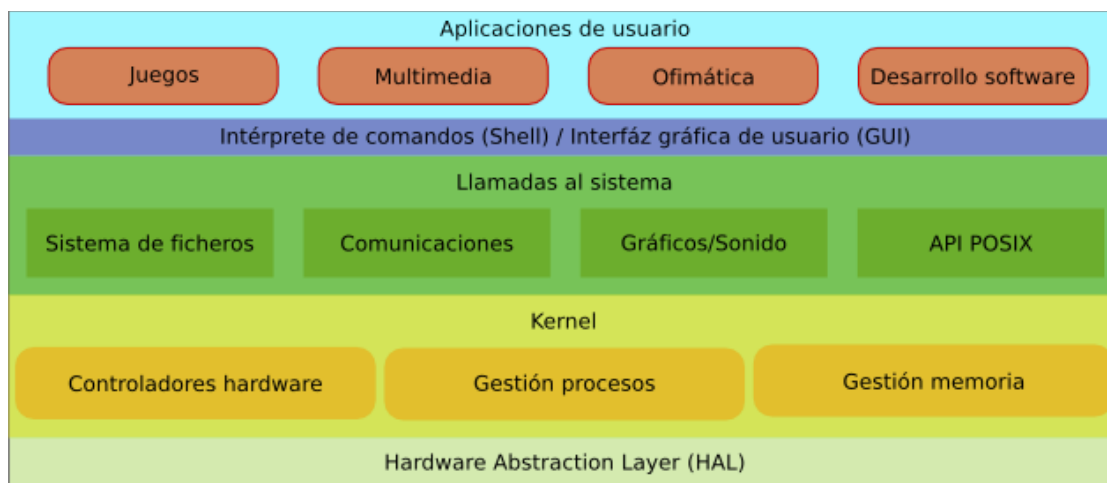


## Boletín de actividades

### Instalación y configuración (Linux I)

#### Introducción

Siendo estrictos, la palabra Linux hace referencia únicamente al núcleo (kernel) del sistema operativo. ¿Recuerdas la estructura de un sistema operativo?



El nombre Linux proviene de su creador, Linus Torvalds, cuando todavía siendo un estudiante de la Universidad de Helsinki empezó a trabajar en un sistema alternativo al comercial Unix. Su objetivo era conseguir un sistema mínimo capaz de ejecutar al menos un shell (intérprete de comandos) y un compilador.

El nombre más correcto para designar a estos sistemas operativos es GNU/Linux, ya que el resto del sistema operativo son herramientas del proyecto GNU. GNU es un proyecto desarrollado por la FSF (Free Software Foundation o Fundación para el software Libre), una organización creada por Richard Stallman y otros impulsores del software libre para difundir este movimiento. Este movimiento se identifica por la libre utilización, modificación y redistribución del software bajo los términos de una licencia que lo regule. Entre ellas, destaca la licencia GPL (General Public License o Licencia Pública General de GNU).

Algunos de los programas desarrollados por el proyecto GNU son: bash (intérprete de comandos), gnome (interfaz gráfica), gimp (edición fotográfica), emacs (editor de texto)....

Con el tiempo se han ido creado variantes del sistema que incorporan diferentes programas para alcanzar ámbitos de aplicación específicos o cubrir las necesidades de un determinado grupo de usuarios. A estas variantes se les denomina *distribuciones*.

Puedes echarle un vistazo al siguiente enlace para conocer las mejores distribuciones GNU/Linux: <http://www.linuxadictos.com/mejores-distribuciones-linux-2016.html>

Para cumplir los objetivos del módulo, nos centraremos en dos distribuciones **Ubuntu** y **Arch Linux**. La primera es más comercial y está enfocada a todo tipo de usuarios, de principiantes a profesionales. La segunda está totalmente guiada por una comunidad de usuarios y se considera una de las distribuciones que más ha mantenido el funcionamiento original de los sistemas GNU/Linux. Está pensada para usuarios avanzados convirtiéndose en una gran distribución para aprender cómo funcionan realmente los sistemas GNU/Linux.

Ten en cuenta que aunque instales una distribución específica, no hay ningún impedimento en modificar posteriormente el sistema para incorporar otras herramientas o programas que sean de nuestro interés.

Algunas de las principales diferencias entre ambas distribuciones son:

	Ubuntu	Arch Linux
<b>Entorno gráfico</b>	Unity	Ninguno (se recomienda xfce)
<b>Gestor de paquetes</b>	apt	pacman
<b>Mantenimiento del sistema</b>	Canonical y comunidad	Comunidad

### Actividad 1. Estructura de directorios

Antes de instalar, vamos a revisar la estructura de directorios que utilizan los sistemas GNU/Linux. Para gestionar adecuadamente un sistema de estas características, es fundamental que conozcas sus principales directorios y lo que almacenan.

Indica en qué directorio del sistema se guarda la información indicada:

Información	Directorio
Archivos que intervienen en el <b>arranque</b> del sistema. Por ejemplo: el gestor de arranque, ficheros de configuración del gestor de arranque....	
Archivos de <b>configuración</b> del sistema: Por ejemplo: configuración de la red, el nombre del equipo...	
Archivos personales de cada uno de los <b>usuarios</b> que inician sesión en el equipo. Por ejemplo: Escritorio del usuario, configuración personal...	

Recursos: <http://blog.desdelinux.net/estructura-de-directorios-en-linux/>

### Actividad 2. Instalación Ubuntu

Instala el sistema operativo Ubuntu en una máquina virtual.

Durante la instalación, elige un particionado manual del disco y ubica la carpeta /home en una partición independiente. Esto nos permitirá reinstalar el sistema sin perder nuestros datos personales.

Recursos: <http://somebooks.es/instalar-ubuntu-16-04-lts-xenial-xerus-desde-cero/>

### Actividad 3. Instalación Arch Linux

Instala el sistema operativo Arch Linux en una máquina virtual.

Recursos:

- [Wiki oficial de Arch Linux.](https://wiki.archlinux.org/)
- <http://blog.desdelinux.net/guia-de-instalacion-de-arch-linux-2014/>

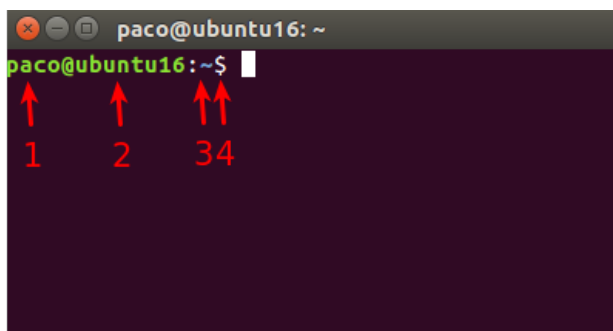
Instala también una interfaz gráfica para el sistema. Se recomienda Xfce.

Puedes ver una descripción de las interfaces gráficas más utilizadas en el siguiente enlace: <https://victorhckinthefreeworld.com/2016/01/26/los-mejores-entornos-de-escritorio-de-linux-para-el-2016/>

### Actividad 4. Intérprete de comandos.

La función del *intérprete de comandos* o *shell del sistema* es recibir las órdenes del usuario, interpretarlas, ejecutarlas y mostrar su resultado. Aprender a utilizarlo es fundamental para obtener el máximo rendimiento del sistema.

Indica a qué hace referencia cada parte del *prompt* del terminal:



### Actividad 5. Intérprete de comandos.

Hay ciertas órdenes que requieren privilegios de administrador para ser ejecutadas. Por ejemplo, la orden `apt-get upgrade` que sirve para actualizar el sistema.

Aún siendo `paco` un usuario con privilegios de administrador, fíjate lo que ocurre al ejecutar la instrucción:

```
paco@ubuntu16: ~$ apt-get upgrade
E: No se pudo abrir el fichero de bloqueo «/var/lib/dpkg/lock» - open (13: Permiso denegado)
E: No se pudo bloquear el directorio de administración (/var/lib/dpkg/), ¿está como superusuario?
paco@ubuntu16:~$
```

Entonces, ¿cómo se debe proceder para ejecutar este tipo de instrucciones?

**Actividad 6. Interprete de comandos.**

Empecemos a hacer operaciones con el sistema desde la línea de comandos.

*Nota: es muy recomendable que realices estas acciones en una máquina virtual con Ubuntu para no interferir con tu sistema original.*

Recursos: Documento PDF *Comandos GNU/Linux más utilizados* de la sección *Saber mas..* del apartado 4.1. *interprete de comandos*.

Ejercicios de manipulación de directorios y ficheros:

1. Crea una nueva carpeta *CarpetaPadre* en el escritorio de tu directorio personal. Dentro de ella crea otras dos llamadas: *CarpetaHija1* y *CarpetaHija2*. Dentro de estas carpetas crea otras dos llamadas: *CarpetaNieta1*, *CarpetaNieta2* y *CarpetaNieta3*, *CarpetaNieta4*.
2. Dentro de cada carpeta nieta crea dos archivos de texto, nómbralos como: *archivoTexto1*, *archivoTexto2*, *archivoTexto3*, *archivoTexto4*, *archivoTexto5*, *archivoTexto6*, *archivoTexto7*, *archivoTexto8*. Los archivos deben tener algún contenido.
3. Copia el *archivoTexto6* (en la *CarpetaNieta3*) a la *CarpetaNieta1* con el nombre *copia\_archivoTexto6*.
4. Copia el *archivoTexto5* (en la *CarpetaNieta3*) a la *CarpetaHija1* con el nombre *copia\_archivoTexto5*.
5. Copia la *carpetaNieta3* a la *CarpetaHija1* con el nombre *copia\_CarpetaNieta3*.
6. Mueve el *archivoTexto1* (en la *CarpetaNieta1*) a la *CarpetaHija2*.
7. Borra el archivo *copia\_archivoTexto5* (en la *CarpetaHija1*).
8. Borra la *CarpetaNieta4*.

Ejercicios de comandos generales comunes:

1. Apaga o reinicia el equipo.
2. Consulta la fecha y hora del sistema.
3. Limpia el terminal.
4. Consulta el manual del comando *ls* y averigua que opción permite ver los archivos ocultos de un directorio y qué opción permite ver una descripción detallada del contenido de una carpeta.
5. Muestra los procesos que se están ejecutando actualmente y su rendimiento.
6. Abre la calculadora desde la interfaz gráfica. Cierra la calculadora desde la línea de comandos. Acción típica cuando se bloquea una aplicación.
7. Cambia la contraseña a tu usuario.

**Actividad 7. Intérprete de comandos.**

Una vez revisados los principales comandos pon a prueba tus conocimientos con el *Crucigrama de conceptos GNU/Linux* de la sección *Saber más...* del apartado 4.1. *Intérprete de comandos*.

**Actividad 8. Instalación/desinstalación de aplicaciones.**

En los sistemas Linux hay dos conceptos importantes en la instalación o desinstalación de aplicaciones. Uno es el concepto de *paquete* y otro el de *dependencia*. El primero se refiere a una aplicación que ya está configurada para ser instalada. Un *paquete* contiene la aplicación y la configuración para instalarse en un sistema.

El segundo concepto hace referencia a que una aplicación puede requerir que ya se encuentren instaladas en el sistema otras aplicaciones o librerías. A estas otras aplicaciones o librerías necesarias para el funcionamiento de una determinada aplicación se les denomina *dependencias*.

Para gestionar los paquetes (instalarlos, desinstalarlos, resolver sus dependencias...) se utilizan aplicaciones denominadas gestores de paquetes. En Ubuntu, el gestor de paquetes principal es el conocido `apt-get`.

`apt-get` está configurado para utilizar unos repositorios. Un repositorio es una ubicación de red que almacena paquetes de software.

Realiza las siguientes tareas utilizando `apt-get`:

1. Instala la aplicación *shutter* para realizar capturas de pantalla.

2. Instala la aplicación *spotify* para escuchar música en streaming.

Recursos: <https://www.spotify.com/es/download/linux/>

3. Busca en los repositorios el nombre del paquete para instalar la aplicación Chromium. (versión libre de Google Chrome). Aségurate de que es el paquete que buscas consultando su información detallada.

4. Ahora vamos a desinstalar una aplicación. Haz las siguientes tareas:

- Instala la aplicación GIMP.
- Realiza algunas acciones con el programa y configura la aplicación a tu gusto.
- Comprueba que dentro de tu carpeta personal se ha creado un archivo/carpeta oculta para guardar tu configuración personalizada.
- Desinstala completamente la aplicación GIMP, incluyendo la configuración personalizada y las dependencias de la aplicación.

Recursos: <http://www.ubuntu-guia.com/2011/01/comando-apt-get-en-ubuntu.html>

5. Actualiza el sistema y todas las aplicaciones.

**Actividad 9. Instalación/Desinstalación de aplicaciones.**

Las aplicaciones *Synaptic* y *Aptitude* son dos gestores de paquetes que proporcionan una interfaz gráfica para el uso de `apt-get`. La primera utiliza el gestor de ventanas mientras que la segunda incorpora una interfaz amigable a través del terminal.

Ahora que sabes instalar aplicaciones, instala estos dos gestores de paquetes y prueba su funcionamiento.

Ten en cuenta que *aptitude* recuerda las dependencias de un paquete mientras que *synaptic* no lo hace. Esto provoca que las desinstalaciones con *aptitude* sean más limpias, desinstalando aquellos paquetes que se instalaron como dependencias siempre que no se requieran por otra aplicación.

**Actividad 10. Instalación/Desinstalación de aplicaciones.**

No todas las aplicaciones se encuentran en repositorios. En ocasiones, los desarrolladores proporcionan directamente el paquete con un formato adecuado para ser instalado manualmente en el sistema operativo.

En el caso de Ubuntu los paquetes de instalación tienen extensión `.deb` (dependiendo de la distribución de Linux la extensión puede variar, en este caso, `.deb` indica que es un paquete preparado para instalarse en una distribución basada en Debian como Ubuntu).

El principal gestor que se utiliza para realizar instalaciones manuales de paquetes es `dpkg`. En este sentido, otro gestor muy utilizado es `gdebi`. La principal diferencia entre `dpkg` y `gdebi` es que el primero no resuelve dependencias del paquete mientras que el segundo sí lo hace.

Tanto el editor de código *Atom* como *Visual Studio Code* se proporcionan como paquetes de instalación manual. También ambos tienen dependencias.

1. Instala *Atom* en el sistema utilizando `dpkg`. Ábrelo para comprobar que funciona.

Recursos: <http://www.ubuntu-guia.com/2011/01/comando-apt-get-en-ubuntu.html>

2. A continuación instala *Visual Studio Code* utilizando `gdebi`. Ábrelo para comprobar que funciona.
3. Desinstala *Atom* con `dpkg`.

**Actividad 11. Instalación/Desinstalación de aplicaciones.**

En otras ocasiones los desarrolladores no proporcionan un paquete sino un script que se encarga de instalar la aplicación. En estos casos, nuestra tarea consiste en descomprimir y desempaquetar la aplicación y a continuación identificar el script de instalación y ejecutarlo.

Instala la aplicación *Foxit Reader* como lector y editor de PDF siguiendo las instrucciones que aparecen en la propia web: <https://www.foxitsoftware.com/products/pdf-reader/>

**Actividad 12. Webmin**

*Webmin* es una aplicación web que no permite administrar un sistema Linux de forma remota.

1. Descarga *Webmin* de su página oficial e instálalo en el sistema.
2. Accede a la interfaz de administración remota desde la máquina anfitriona y realiza alguna tarea de administración sobre la máquina virtual.