Universidad de Alcalá

Departamento de Automática

INFORMÁTICA

Práctica 1.
Introducción al sistema operativo Linux.

Grado en Ingeniería en Electrónica y Automática Industrial
Curso 2024-2025
v2.8 (02.09.24)

INTRODUCCIÓN A LINUX.

Unix es un sistema operativo que ha ido siendo desarrollado desde los años 70. Inicialmente se creó en la empresa AT&T, pero, debido a su éxito, desde muy pronto aparecieron diversas variantes (Solaris, AIX, HP-UX). Algunas de ellas fueron desarrolladas por empresas de computación (IBM, Sun, HP), mientras que alguna otra la puso en marcha una universidad (BSD).

A día de hoy, tal vez la variante más extendida de Unix sea Linux, un proyecto de software de libre distribución iniciado por Linus Torvalds. La característica de libre distribución de Linux ha favorecido que multitud de programadores colaboren en el desarrollo del sistema, tanto en labores de creación de código como de depuración, documentación, etc.

Dado el gran interés que despierta este sistema, y debido al hecho de que el acceso al código fuente del mismo es posible, existen actualmente gran cantidad de grupos de trabajo que ofrecen una versión particular de Linux. Estas versiones se denominan coloquialmente distribuciones, y constan del llamado kernel de Linux, que es la parte que se encarga de hacer funcionar el computador y de gestionar los recursos físicos de los que dispone, y de una colección particularizada de aplicaciones para completar esta funcionalidad básica del kernel y permitir realizar actividades que van desde la generación de documentación (procesador de textos, hojas de cálculo, bases de datos) hasta el desarrollo de programas (compiladores y bibliotecas de código, depuradores), pasando por la gestión y reproducción de contenido multimedia (reproductores y editores de audio, video, fotografía), etc. Muchas de estas distribuciones son ofrecidas de forma gratuita por sus creadores, pero también hay algunas, con una orientación más profesional, que son de pago. La distribución que se ha elegido para ser utilizada en el laboratorio es Ubuntu, de la que se toma una actualización reciente. Se trata de una versión gratuita, fácil de obtener, y muy extendida, por lo que la búsqueda de documentación por parte del alumno ha de ser necesariamente fácil.

Manejo básico de Linux.

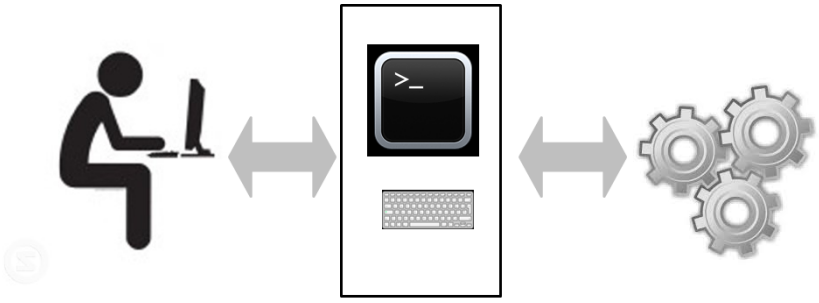
Como la mayoría de sistemas operativos de la actualidad, Linux dispone de un interfaz gráfico para prácticamente todas las tareas que permite realizar. Este entorno gráfico se centra en su escritorio que, cuestiones estéticas al margen, es muy similar al de cualquier otro sistema operativo. Existe un mecanismo para acceder a las aplicaciones, la barra de menú situada en la parte superior del escritorio, y desde ahí es posible lanzar aplicaciones como el administrador de archivos, clientes de HTTP (navegadores), las aplicaciones de configuración del sistema, el terminal, etc.

Estas aplicaciones permiten acceder a la gestión del equipo con un interfaz gráfico, lo que la hace muy cómoda, pero en esta práctica, con el objeto de permitir ver de una forma más directa lo que se está haciendo, se va a realizar la gestión básica del equipo por medio de comandos. Para ello, nos vamos a valer del interfaz de comandos más básico, el terminal.

El terminal.

El terminal es un interfaz básico de modo texto. Es bidireccional, y en el sentido salida vuelca la información a la pantalla. En el sentido entrada, recibe la información del teclado. El hecho mismo de lanzar un terminal lleva asociado el arranque de un intérprete de comandos, que es la aplicación con la que realmente se está llevando a cabo la comunicación. La siguiente figura ilustra este hecho.

De esta manera, cada comando que el usuario introduce en el teclado, que es la parte del terminal que se encarga de la entrada, se transmite al intérprete de comandos, que se encarga de hacerlo ejecutar (de hecho, lo que hace es crear un nuevo proceso que lo ejecute) y vuelca los resultados pertinentes en la ventana de texto, que es la parte del terminal que se encarga de la salida, para que el usuario los lea.



Usuario Terminal Interfaz de comandos

Comandos.

Para manejar el equipo por medio del terminal y el intérprete de comandos es preciso que el usuario conozca estos comandos. Existen diversas fuentes de documentación donde consultarlos, pero tal vez la más interesante sea una que todos los sistemas Linux suelen llevar incorporada, que es el manual. El manual es, a su vez, un comando, por lo que para invocarlo, es necesario abrir un terminal y teclearlo: Por ejemplo:

```
>man ls
```

(A continuación del comando se debe pulsar la tecla Intro). El comando `man` invoca la aplicación que muestra la página requerida del manual que se solicita, en este caso, la referente al comando `ls`.

A continuación se listan algunos de los comandos más útiles y una sencilla descripción de su funcionalidad.

Comando	**Funcionalidad**
`man`	muestra páginas del manual
`ls`	lista el contenido de un directorio
`cd`	cambia el directorio activo
`pwd`	muestra el directorio activo
`mkdir`	Crea un directorio
`cp`	copia archivos o directorios
`rm`	elimina archivos
`rmdir`	elimina directorios
`mv`	traslada archivos o directorios
`cat`	vuelca el contenido de un archivo
`chmod`	cambia los permisos de un archivo o directorio
`chown`	cambia el propietario de un archivo o directorio

Enseguida entramos a desarrollar con algo más de extensión estos comandos, pero antes es preciso entretenerse un momento con el sistema de archivos.

Sistema de archivos

Un sistema de archivos es una forma de organizar la información presente en los dispositivos de un sistema. Linux permite usar una gran variedad de sistemas de archivos, desde los suyos propios hasta sistemas de archivos de otros sistemas operativos (como Windows). Las características de más bajo nivel de un sistema de archivos están fuera del ámbito de interés de esta asignatura, así que no se tratarán aquí. En esta asignatura nos vamos a limitar a un tratamiento un poco superficial del mismo.

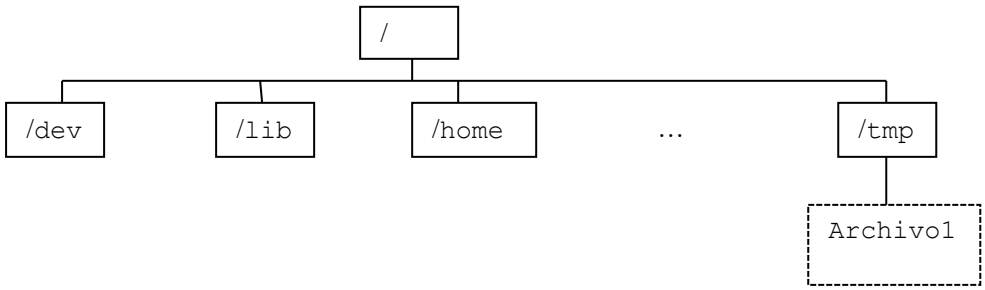
Un sistema de archivos se organiza en forma de árbol. Existe un punto, el directorio raíz, a partir del cual se realiza dicha organización. El directorio raíz contiene una serie de directorios y archivos. A su vez, cada uno de estos directorios puede contener otros directorios y algunos archivos, y así sucesivamente.

El sistema de archivos permite organizar el contenido de los dispositivos de almacenamiento del equipo de una forma sencilla y eficaz. Cada dispositivo (fundamentalmente discos) está organizado según un sistema de archivos, y Linux, al arrancar, crea un sistema de archivos básico adicional, a modo de estructura inicial, en el que irá insertando los sistemas de archivos de los dispositivos que vaya incorporando a medida que vaya progresando en su arranque. Una vez termina la secuencia de arranque, Linux tiene el contenido de todos los dispositivos organizado en un único sistema de archivos lógico (virtual), formado por todos los sistemas de archivos de todos los dispositivos que se ha encontrado en el arranque.

En cuanto a los dispositivos, un equipo suele tener al menos una unidad de disco duro, que puede estar dividida en particiones. Cualquier configuración es, en principio, posible, desde un único disco con una única partición hasta varios discos con varias particiones cada uno. Cada partición está organizada internamente como un sistema de archivos. De todas las particiones presentes en el equipo existe una, la principal, que es la que contiene la parte fundamental del sistema operativo. Esta partición, en concreto, su sistema de archivos, se "engancha a" o "cuelga de" (la expresión técnica es "se monta en") la partición raíz. Se dice que la raíz es el punto de montaje ("*mount point*") de la partición principal. Cada sistema de archivos de cada partición o disco que se añada al sistema de archivos lógico de Linux se monta en un punto de montaje, que no es más que un directorio presente en el sistema de archivos lógico de Linux.

Cuando se arranca un terminal, se define lo que se conoce como directorio activo, que es la carpeta a la que se aplican, por omisión, los comandos. En otras palabras, el directorio actual. Este directorio inicial suele ser la carpeta de usuario del usuario que ha entrado al sistema. Se suele denominar ruta a la situación de un directorio o un archivo con respecto a una posición de referencia. Si la posición de referencia es el directorio raíz del sistema de archivos, entonces la ruta se dice completa o absoluta.

Cuando un comando requiere que se le indique un archivo o un directorio sobre el que trabajar, éste se puede especificar bien con una ruta absoluta, bien con una ruta parcial. Si la ruta es parcial, se supone referida al directorio actual. En la figura siguiente, el archivo `Archivo1` está situado en el directorio `tmp`, que a su vez está situado en el directorio raíz. Si el directorio actual es `tmp`, para especificar el archivo `Archivo1` hay dos opciones: la primera, hacerlo de forma relativa al directorio actual. En ese caso, basta con indicar `Archivo1` al comando para que localice correctamente el archivo. La segunda opción es utilizar una ruta absoluta. En este caso, el archivo se especificará como `/tmp/Archivo1`. Por otra parte, para referirse al mismo archivo cuando el directorio actual es `home`, la ruta relativa al directorio actual sería `../tmp/Archivo1`, mientras que la absoluta no cambia respecto a la de antes: `/tmp/Archivo1`. En el caso de la ruta relativa, los dos puntos (`..`) hacen referencia al directorio del que cuelga el actual (en este caso haría referencia al raíz, que es del que cuelga `home`).



Comandos más habituales.

Los comandos más comunes para manejarse con el sistema de archivos de Linux están indicados en la tabla que se ha incluido más arriba. Son: `cd`, para cambiar de directorio activo, `ls`, para listar el contenido de un directorio, `pwd`, para mostrar la ruta completa del directorio actual, `cp` para copiar archivos o directorios de un lugar a otro del sistema de archivos, y `mv`, que hace algo parecido a `cp` pero eliminando el objeto de su ubicación inicial: `cp` duplica, y `mv` traslada. En ocasiones es preciso crear un

directorio. Para ello se usa el comando `mkdir`. Por último, a veces es preciso eliminar cosas. `rm` se utiliza para eliminar archivos, y `rmdir` para eliminar directorios (en el caso de éste último, para poder eliminar el directorio, éste debe estar vacío). Cada uno de estos comandos requiere unos parámetros de entrada, que son los objetos sobre los que va a trabajar, y unos de salida, que es donde va a dejar el resultado. En algunos casos no es necesario especificar un parámetro, ya que por omisión trabajan sobre el directorio actual. Por ejemplo, `ls` lista el contenido de un directorio. Si no se le indica nada más, el directorio sobre el que trabaja es el actual. Pero es posible indicarle que liste un directorio distinto del actual, simplemente añadiéndolo a continuación del nombre del comando.

Además de los comandos mencionados, y de `man`, que se explicó más arriba, en la tabla figura algún comando más. Por ejemplo, `cat` vuelca en pantalla el contenido de un archivo, y `chmod` y `chown` gestionan los permisos y la propiedad de un archivo o directorio. Un archivo o un directorio tiene tres tipos de permiso: para lectura, para escritura y para ejecución. En el caso de un archivo, esto significa respectivamente poder acceder al contenido, poder modificar el contenido y poder ejecutar el archivo. En el caso de un directorio, es ligeramente diferente: lectura significa ver el contenido de ese directorio, escritura significa modificar el contenido de ese directorio (añadir o eliminar archivos o directorios contenidos en él), y ejecución significa entrar en ese directorio (con un comando `cd`, por ejemplo). Cada elemento del sistema de archivos (archivo o directorio) tiene permisos asignados para tres tipos de usuario: el propietario del elemento, el grupo al que pertenece, y la totalidad de usuarios del sistema. Cada uno de estos tres grupos tiene permisos diferentes para permitir, por ejemplo, que un usuario pueda leer y modificar su propio archivo, pero los demás usuarios de su grupo sólo puedan leerlo, y un usuario genérico (uno que no sea ni el propietario del archivo ni otro usuario de su grupo) no pueda ni leerlo ni escribirlo. Esta información relativa a archivos y directorios se muestra cuando se utiliza el comando ls con la opción –l. El siguiente ejemplo ilustra todo esto:

```
bash-3.2$ ls -l
drwxr-xr-x  6 nacho  staff   204 Nov  3  2008 images
-rw-r--r--  1 nacho  staff  2628 Nov  3  2008 index.html
```

La invocación del comando `ls -l` da como resultado el listado de los dos elementos presentes en el directorio actual: el directorio `images` y el archivo `index.html`. Ambos son propiedad del usuario `nacho`, que pertenece al grupo `staff`. El directorio `images` (se sabe que es un directorio por la letra `d` que encabeza su línea) tiene permisos de lectura (`r`), escritura (`w`) y ejecución (`x`) para el usuario `nacho`. Los usuarios del grupo `staff` pueden ver su contenido (`r`) y entrar en él (`x`), pero no modificar su contenido. Por último, un usuario genérico del sistema puede también ver el contenido del directorio y entrar en él, pero no modificarlo. Los permisos figuran a continuación de la letra `d`, y son tres grupos de tres letras (o un guión, si el permiso no está concedido) en el orden `rwx`, es decir, lectura, escritura y ejecución.

El archivo `index.html` (se sabe que es un archivo porque la primera letra de su línea es un guión, es decir, no es un directorio) presenta permisos de lectura y escritura para `nacho`, pero no de ejecución (ya que no es un archivo ejecutable, como permite imaginar el nombre que se le ha asignado), y sólo de lectura tanto para los ususarios del grupo `staff` como para cualquier usuario genérico.

El comando `chmod` permite al propietario de un archivo o un directorio modificar los permisos que tiene asignados. De esta forma, un usuario puede en algún momento permitir la modificación de un archivo propio a otros usuarios, y cuando lo considere oportuno dejar de permitirlo simplemente retirando el permiso de modificación correspondiente. Por su parte, `chown` permite cambiar el propietario de un archivo. Sin embargo, por cuestiones de seguridad sólo el administrador del sistema (usuario `root`) puede ejecutar este comando.

EJERCICIOS PARA EL APARTADO 1.

A continuación se presenta una lista de tareas que el alumno debe realizar. **Como paso previo** a ello es preciso consultar las páginas del manual de los comandos descritos en la tabla incluida más arriba, con el objeto de adquirir un conocimiento más preciso tanto de su funcionamiento como de la sintaxis que se usa en su invocación. Una vez hecho esto, se puede proceder a realizar las tareas

Tareas:

- Crea un archivo de texto (por ejemplo, con la aplicación `gedit`). Guárdalo en tu carpeta de usuario.

- Abre un terminal. Sitúate en tu directorio de usuario (el terminal suele situarse ahí al ser iniciado). Comprueba que el archivo se encuentra ahí. Obtén su tamaño y permisos de acceso.

- Modifica estos permisos para que solamente pueda ser modificado por su propietario (tú). El resto de usuarios que tienen acceso a él tendrá únicamente permisos de lectura.

- Crea un subdirectorio. Un subdirectorio es simplemente un directorio que está situado dentro de otro. En este caso, lo vas a crear en tu carpeta de usuario. Copia en él el archivo de texto que has creado más arriba.

- Cambia al subdirectorio (haz que el directorio activo sea el subdirectorio). Comprueba que el archivo que has copiado antes está en él.

- Elimina el archivo.

- Cambia al directorio de usuario (un nivel por encima del subdirectorio que has creado antes). Elimina el subdirectorio. Sólo es posible eliminar subdirectorios vacíos, de modo que si no has eliminado el archivo que contenía, no podrás eliminar el subdirectorio.