# Apuntes.pdf

*Material Examen Módulo I*

**2º Sistemas Operativos**

**Grado en Ingeniería Informática**

**Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación**
**Universidad de Granada**

# KEEP CALM AND ESTUDIA UN POQUITO

WUOLAH + #QuédateEnCasa

# MÓDULO I: Administración de Linux (Teoría)

## Sesión 1: Herramientas de administración básicas

### 1.1 Súper-usuario

- Nombre: **root**
- Grupo: **root**
- Home: **/root**

### 1.2 Usuario

- **username**: nombre de usuario
- **UID**: identificador de usuario (número entero asignado por el sistema). El UID del root es el 0.
- **GID**: grupos a los que pertenece el usuario (especificado en **/etc/passwd** y **/etc/group**). El GID principal del súper-usuario es el 0.
- Si se quiere evitar que un usuario no pueda entrar al sistema se le puede asignar al campo en el que se indica el Shell los nombres de los archivos **/bin/false** o **/sbin/nologin**.

***Tabla 1**: Archivos que especifican los usuarios, grupos y contraseñas (passwords) del sistema.*

| | |
|---|---|
| **/etc/passwd** | Almacena información de las cuentas de usuarios. |
| **/etc/shadow** | Guarda los password encriptados e información de "envejecimiento" de las cuentas. |
| **/etc/group** | Definición de los grupos y usuarios miembros. |
| **/etc/skel** | Guarda unos archivos de configuración del Shell, los cuales se copian en el directorio HOME asignado cuando se crea una cuenta de usuario. |

***Tabla 2**: Archivos de configuración para el Shell Bash.*

| | |
|---|---|
| **.bash_profile** | Se ejecuta al hacer el login (conectarnos al sistema) y en él podremos indicar alias, variables, configuración del entorno, etc. que deseamos iniciar al principio de la sesión. |
| **.bashrc** | Su contenido se ejecuta cada vez que se ejecuta una Shell, tradicionalmente en este archivo se indican los programas o scripts a ejecutar. |

| | |
|---|---|
| **.bash_logout** | Se ejecuta al salir el usuario del siste,a y en él podremos indicar acciones, programas, scripts, etc. que deseemos ejecutar al salirnos de la sesión. |

***Tabla 3**: Valores para controlar el envejecimiento de contraseñas y cuotas. Uso da la orden **chage**.*

| | | |
|---|---|---|
| **changed** | **chage -d ult_dia usuario** | Fecha del último cambio de contraseña. |
| **minlife** | **chage -m min_dias usuario** | Número de días que han de pasar para poder cambiar la contraseña. |
| **maxlife** | **chage -M max_dias usuario** | Número de días máximos que puede estar con la misma contraseña son cambiarla. |
| **warn** | **chage -W warn_dias usuario** | Cuántos días antes de que la contraseña expire (**maxlife**) será informado sobre ello, indicándole que tiene que cambiarla. |
| **inactive** | **chage -I inac_dias usuario** | Número de días después de que la contraseña expire que la cuenta se deshabilitará de forma automática si no ha sido cambiada. |
| **expired** | **chage -E exp_dias usuario** | Fecha en la que la cuenta expira de forma automática. |

## 1.3 Grupos

- Conjunto de usuarios que comparten recursos o archivos del sistema.

- **Groupname**: nombre del grupo.
- **GID**: identificador del grupo.
- **/etc/group**: archivo de configuración. Cada línea de este archivo presenta el siguiente formato: **nombre:x:gid:lista de usuarios**.

## 1.4 Archivos

- Nomenclatura **absoluta**: empieza por **/**.
- Nomenclatura **relativa**: no empieza por **/**.

- **vmlinux* / vmlinuz***: archivo ejecutable que contiene el kernel de Linux.
- **/dev**: directorio que agrupa los archivos de dispositivo.
- Archivos **Socket/FIFO**: tipos de archivos para establecer comunicación entre proceso.
- Archivos de **directorio**: encargados de soportar la estructura jerárquica del sistema de archivo.

- Archivos de tipo **enlace**: archivos que referencian a otros archivos. Existen dos tipos diferentes, los **hard link** (enlaces duros) y **soft link** (enlace simbólico).
- **/etc**: directorio que recoge todos los archivos de configuración del sistema
- **/proc**: contiene archivos de texto que permiten acceder a información relativa a los sistemas de archivos.

***Tabla 4**: Nombres de directorio y tipo de información que almacenan en el estándar FHS.*

| | |
|---|---|
| **/bin** | Programas de utilidad fundamentales para ser utilizados por cualquier usuario del sistema. |
| **/sbin** | Programas de utilidad fundamentales para ser utilizados por el usuario **root**. |
| **/boot** | Archivos fundamentales para el programa **Boot Loader**. |
| **/dev** | Todos los archivos especiales de dispositivos. |
| **/etc** | Archivos de configuración del sistema. |
| **/home** | Los directorios de inicio de todos los usuarios que disfrutan de una cuenta en el sistema, excepto, el directorio de inicio del **root**: **/root**. |
| **/lib** | Bibliotecas sin las que no pueden funcionar los programas ubicados en /bin y /sbin. |
| **/media** | Este directorio actúa como punto de montaje para dispositivos extraíbles: DVD-ROM, dispositivos USB, etc. |
| **/mnt** | Este directorio actúa como punto de montaje para sistemas de archivos montados temporalmente. |
| **/opt** | Normalmente aquí se ubican los programas que no forman parte de la distribución instala en el sistema. |
| **/proc** | Sistema de archivos virtual que hace de interfaz con el núcleo y los procesos. |
| **/tmp** | Archivos temporales que normalmente no se mantienen una vez se apaga el sistema. |
| **/usr** | Archivos ejecutables, archivos de código fuente, bibliotecas, documentación y, en general, todos los programas y utilidades. |
| **/var** | Los archivos cuyo contenido se espera que cambie el funcionamiento normal del sistema. |

- **/etc/fstab** y **/etc/mtab**: archivos fundamentales para obtener información de los sistemas de archivo.

***Tabla 5**: Archivos con información de sistema de archivos que proporciona **/proc**.*

| | |
|---|---|
| **/proc/filesystems** | Enumera, uno por línea, todos los tipos de sistemas de archivos disponibles. |
| **/proc/mounts** | Sistemas de archivos montados actualmente, incluyendo los que se hayan montado manual o automáticamente tras el arranque del sistema. |

# Sesión 2: Herramientas de administración del SA

*Libro KERRISK pag 251 y Siguientes: conceptos de partición, sistema de archivos, i-nodo.*

*Libro KERRISK pag 339 y siguientes: Entradas de directorios, enlaces.*

## 2.1 Tipos de archivos

- Archivos **regulares**: archivos de programas y datos.
- **Directorios**: archivos que soportan estructuras jerárquicas de organización de la información en un SA. Consiste en una sucesión de entradas de número de i-nodo y nombre.
- **Enlaces simbólicos**: archivos que permiten referenciar a otros archivos desde distintas ubicaciones en el espacio de nombres (distintos directorios).
- **Archivos especiales de dispositivo**: archivos que representan dispositivos.
- **FIFO y Socket** (archivos para comunicaciones): permiten comunicar procesos.
- **I-nodo**: guarda todos los metadatos de un archivo.

## 2.2 Partición de dispositivos de almacenamiento secundario.

Para poder utilizar un dispositivo de almacenamiento secundario (*drive*) en un SO es necesario establecer secciones (**particiones**) dentro del dispositivo, que sean **identificables** y que permitan **alojar un SA concreto**. Cuando se crea una partición, debe asociársele una etiqueta que indique el tipo de SA que va a alojar una vez se formatee.

- **Ext2**: **0x83**
- **Swap: 0x82**

- **MBR** (Master Boot Record, sector de arranque maestro): almacena las cuatro particiones primarias.
- **Partición lógica**: subdivisión de una partición primaria de disco.

## 2.3 Asignación de un SA a una partición (formateo lógico)

- **ext2**: SA de alto rendimiento. Mejor rendimiento en términos de velocidad de transferencia de E/S y uso de CPU de entre todos los SA que soporta Linux.
- **ext3**: versión de **ext2** que incluye un "registro por diario", un mecanismo por el cual un sistema informático puede implementar transacciones. Evita la corrupción de las estructuras de datos que soportan la información del SA.
- **ext4**: mismas estructuras a las del **ext3** con las siguientes mejoras:

- **Extensiones**: permiten describir un conjunto de bloques de disco contiguos, mejorando de esta forma el rendimiento de E/S al trabajar con archivos de gran tamaño y reduciendo la fragmentación de disco.
- **Asignación retardada de espacio en disco** (*allocate-on-flush)*: permite postergar en el tiempo la asignación de bloques de disco hasta el momento real en el que se va a realizar la escritura.

## 2.4 Ajuste de parámetros configurables de un SA y comprobación de errores

Situaciones de inconsistencia de metadatos en el SA:

- Bloques que están asignados simultáneamente a varios archivos.
- Bloques marcados como libres pero que están asignados a un archivo determinado.
- Bloques marcados como asignados pero que en realidad están libres.
- Inconsistencia en el número de enlaces de un determinado archivo.
- I-nodos marcados como ocupados pero que realmente no están asociados a ningún archivo.

## 2.5 Montaje y desmontaje de SA

- **Montar un SA**: ponerlos a disposición de los usuarios haciendo que puedan ser accesibles dentro de la jerarquía de directorios.
- **Desmontar un SA**: el SA deja de estar disponible en el espacio de nombres.
  - **No es posible desmontar un SA si está siendo utilizado (busy).**
- **/etc/fstab**: **# <file system> <mount point> <type> <options> <dump> <pass>**
  - **<file system>**: número que identifica el archivo especial de bloques.
  - **<mount point>**: directorio que actúa como punto de montaje.
  - **<type>**: tipo de SA.
  - **<options>**: opciones que se utilizarán en el proceso de montaje. Se especifican como una lista separada por comas y sin espacios.
    - **rw**: lectura-escritura.
    - **ro**: solo lectura.
    - **suid/nosuid**: permitido/no permitido el acceso en modo SUID.

- **auto/noauto**: montar automáticamente/no montar automáticamente (ni ejecutando **mount -a**).
- **exec/noexec**: permitir/no permitir la ejecución de ficheros.
- **usrquota, grpquota**: cuotas de usuario y grupo.
- **defaults = rw,suid,dev,exec,auto,nouser,async**.
- **user, users, owner**: permite a los usuarios montar un SA.
- **u=500, gid=100**: propietario y grupo propietario de los archivos del SA.
- **umask**: máscara para aplicar los permisos a los archivos.

- **<dump>**: si su valor es distinto de 0, indica la frecuencia con la que se realizará una copia de seguridad del SA.
- **<pass>**: especifica el orden en el que la orden fsck realizará las comprobaciones sobre los SAs durante el arranque del sistema.

## 2.6 Administración de Software

La forma más sencilla de instalar y actualizar el software de un sistema es mediante el uso de los gestores de paquetes.

***Tabla 6**: Clasificación de los principales gestores de paquetes.*

| | **Formato .deb** (Debian, Ubuntu) | **Formato RPM** (Red Hat) |
|---|---|---|
| **Modo Línea de Órdenes** | dpkg<br>apt-get*<br>aptitude* (interfaz texto para apt) | rpm<br>TUM<br>apt-get<br>aptitude (interfaz de texto para apt) |
| **Modo Gráfico** | dselect<br>Synaptic*<br>Adept (basado en apt-get)<br>Kpackage (parte de kdeadmin) | Synaptic<br>Pup, pirut y yumex (basados en YUM)<br>Adept (basado en apt-get)<br>Gpk-application (parte de gnome-packagekit)<br>Kpackage (parte de kdeadmin) |

## 2.7 Administración de cuotas

Las cuotas de un disco permiten limitar el número de recursos de un SA que va a poder utilizar un usuario. Es decir, los bloques de disco y los i-nodos. Para ello, hace falta tener instalado el paquete **quota**.

- **Límite hard**: el usuario no puede sobrepasarlo. Si llegase el caso en el cual lo sobrepase, el sistema no le permitirá usar más bloques, por lo que no podrá ampliar el tamaño de sus archivos ya creados, ni crear nuevos archivos (usar más i-nodos).
- **Límite soft**: siempre debe configurarse como un número inferior al límite hard y se puede sobrepasar durante cierto tiempo, pero sin llegar a superar al límite hard. Transcurrido el tiempo que estipule el administrador para poder estar por encima del límite soft, el sistema de cuotas actúa como si se hubiese superado el límite hard. Este tiempo durante el cual se puede superar el límite soft se conoce con el nombre de periodo de gracia.

# Sesión 3: Monitorización del Sistema

## 3.1 Control y gestión de la CPU

Se le asigna la CPU al proceso con prioridad más baja. Los procesos bloqueados no se contemplan.

## 3.2 Control y gestión de dispositivos de E/S

- **Metadatos de un archivo**: información de los archivos.
- **I-nodo**: almacenan los metadatos de los archivos.

El objetivo de los enlaces a archivos es disponer de más de un nombre para los archivos en nuestro espacio de nombres de archivo soportado por la estructura jerárquica de directorios.

- Enlaces **simbólicos**: referencia al nombre del archivo.
- Enlaces **duros**: referencia a los metadatos del archivo.

Todos los archivos tienen 2 enlaces duros creados por el SO de forma automática.

# Sesión 4: automatización de tareas

## 4.1 Los procesos demonio

Características de un proceso demonio:

- Se ejecuta en background y no está asociado a un terminal o proceso login.
- Muchos se inician durante el arranque del sistema y continúan ejecutándose mientras el sistema esté encendido. Otros solo se ponen en marcha cuando son necesarios y se detendrán cuando dejen de serlo.
- Si termina por algún imprevisto, un mecanismo detecta la terminación y lo rearranca.

- En muchos casos, está a la espera de un evento. Si es un servido, espera a que llegue una petición de realizar un determinado servicio. Por ello, habrá un mecanismo que permita la comunicación entre el demonio servidor y los procesos cliente.
- En otros casos, el demonio tiene una labor que hacer de forma periódica, ya sea cada cierto tiempo o cuando se cumpla una condición.
- Suelen lanzar otros procesos para que realicen el trabajo.
- Son programas, no parte del kernel.
- Se ejecutan con privilegio de superusuario (UID=0) y tienen por padre al proceso **init** (PID=1).

## 4.2 Ejecutar tareas a una determinada hora

Con el demonio **atd** podemos provocar la ejecución de una orden en un momento de tiempo especificado.

- Al lanzar la ejecución asíncrona de una tarea con la orden **at** NO estamos creando un proceso hijo de nuestra Shell, este nuevo proceso NO tendrá como entrada estándar, salida estándar y salida de error estándar los asociados a nuestra consola terminal.
- Los archivos de configuración de **atd** son **/etc/at.deny** y **/etc/at.allow**, que determinan que usuarios pueden usar la orden at. El archivo **at.allow**, si existe, contiene una lista con el nombre de todos los usuarios habilitados para ello (uno por línea).

## 4.3 Ejecuciones periódicas

El demonio **cron** es el responsable de ejecutar órdenes con una periocidad determinada. La especificación de las tareas que se desea que se ejecute se hace construyendo un archivo (archivo **crontab**) que deberá tener un formato determinado.

Formato de los archivos crontab:

- Especificando **órdenes**: cada línea del archivo (excepto los comentarios, que empiezan por #) puede contener estos campos (que representan una orden):

  **Minuto hora día-del-mes mes día-de-la-semana orden**

  - Los campos: minuto, hora, día-del-mes, mes y día-de-la-semana se encargan de indicar la periocidad con la que se desea ejecutar **orden**.
  - Cada uno de los campos de determinación del tiempo pueden contener:
    - **Un asterisco:** indica cualquier valor posible.

        - **Un número entero**: activa ese valor determinado.
        - **Dos enteros separados por un guion**: indica un rango de valores.
        - **Una serie de enteros o rangos separados por coma**: activa cualquier valor de los que aparezca en la lista.

- Especificando **variables de entorno**: una línea de un archivo crontab puede ser una línea de asignación de valores a variables de entorno, con la forma:

  **<nombre>=<valor>**

- Variables de entorno automáticas:
    - **SHELL**: se establece a **/bin/sh**.
    - **LOGNAME y HOME**: se timan del archivo **/etc/passwd**.

Los archivos de configuración de cron son **/etc/con.deny** y **/etc/cron.allow,** equivalentes a los de at.