

Materia examen

21/02/2017

Andrei García

*2017*

Contenido

[1. Opciones de conexión de red (Máquina Virtual) 4](#_Toc474803987)

[2. Sistemas de codificación \* 4](#_Toc474803988)

[Sistemas de numeración 4](#_Toc474803989)

[Medidas de la información 4](#_Toc474803990)

[Conversión de sistemas 5](#_Toc474803991)

[Binario a decimal 5](#_Toc474803992)

[Decimal a binario 5](#_Toc474803993)

[Binario a hexadecimal 5](#_Toc474803994)

[Hexadecimal a binario 5](#_Toc474803995)

[Sistemas de codificación 5](#_Toc474803996)

[3. Estructura de un sistema informático 6](#_Toc474803997)

[Componente humano 6](#_Toc474803998)

[Componente informático 6](#_Toc474803999)

[Parte física 6](#_Toc474804000)

[Parte lógica 6](#_Toc474804001)

[4. Funciones de un sistema operativo 7](#_Toc474804002)

[5. Tipos de sistemas operativos 7](#_Toc474804003)

[ Número de usuarios 7](#_Toc474804004)

[ Número de programas o tareas. 7](#_Toc474804005)

[ Número de procesadores. 8](#_Toc474804006)

[ Tiempo de respuesta. 8](#_Toc474804007)

[6. Gestor de arranque 8](#_Toc474804008)

[7. Estructura de un disco duro \* 8](#_Toc474804009)

[Estructura física 8](#_Toc474804010)

[MBR 9](#_Toc474804011)

[Particiones MBR 9](#_Toc474804012)

[Particiones GUID 10](#_Toc474804013)

[8. Secuencia de arranque 11](#_Toc474804014)

[9. Niveles de ejecución 11](#_Toc474804015)

[10. Administración de usuarios locales 12](#_Toc474804016)

[11. Diseño de cuentas de usuario \* 12](#_Toc474804017)

[12. Usuarios predeterminados 13](#_Toc474804018)

[Ubuntu 13](#_Toc474804019)

[Windows server 2008 13](#_Toc474804020)

[13. Seguridad de cuentas de usuario \* 13](#_Toc474804021)

[14. Seguridad de contraseñas \* 14](#_Toc474804022)

[15. Perfiles locales 14](#_Toc474804023)

[16. Gestión centralizada de usuarios 15](#_Toc474804024)

[17. Administración de grupos locales 15](#_Toc474804025)

[18. Grupos predeterminados 15](#_Toc474804026)

[19. Derechos y permisos \* 16](#_Toc474804027)

[ Tipos de usuarios: 16](#_Toc474804028)

[ Tipos de permisos: 16](#_Toc474804029)

[ Ficheros de configuración 16](#_Toc474804030)

[Comandos útiles para el manejo de usuarios: 17](#_Toc474804031)

[Formas de permisos 18](#_Toc474804032)

[Simplificada 18](#_Toc474804033)

[Numérica 18](#_Toc474804034)

[Ampliación de derechos 18](#_Toc474804035)

[20. Listas de control de acceso 19](#_Toc474804036)

# Opciones de conexión de red (Máquina Virtual)

* **No conectado**: Se considera que el sistema invitado no dispone de conexión de red.
* **NAT**: VirtualBox se comporta como si fuera un dispositivo capaz de hacer de servidor de DNS, de DHCP y NAT, generando una IP de una clase distinta a la del host. Las máquinas virtuales con este tipo de red no serán visibles desde otros equipos.
* **Red NAT**: Similar a NAT, aunque en este tipo de red las máquinas virtuales podrán verse entre sí.
* **Adaptador puente**: La tarjeta de red del sistema invitado es la misma que la del anfitrión pero con una IP diferente, dentro del rango de la red del anfitrión.
* **Red interna**: Permite crear redes internas para las diferentes máquinas virtuales que no utilizarán la tarjeta de red del anfitrión.
* **Adaptador sólo-anfitrión**: se crea una tarjeta virtual en el host que permitirá la comunicación entre las máquinas virtuales y el. No es necesario que el anfitrión disponga de tarjeta de red física.
* **Controlador genérico**: Permite configuraciones complejas como la comunicación entre máquinas virtuales en distintos anfitriones.

|  |  |  |
| --- | --- | --- |
| 0 | 0 | 0 |
| 1 | 1 | 1 |
| 2 | 10 | 2 |
| 3 | 11 | 3 |
| 4 | 100 | 4 |
| 5 | 101 | 5 |
| 6 | 110 | 6 |
| 7 | 111 | 7 |
| 8 | 1000 | 8 |
| 9 | 1001 | 9 |
| 10 | 1010 | A |
| 11 | 1010 | B |
| 12 | 1100 | C |
| 13 | 1101 | D |
| 14 | 1110 | E |
| 15 | 1111 | F |

# Sistemas de codificación \*

Todas las memorias son biestables: Sólo pueden presentar dos estados (0 o 1), siendo éste el sistema de codificación básico.

## Sistemas de numeración

Un sistema de numeración es un conjunto de símbolos y reglas que se utilizan para representar cantidades. La característica fundamental de un sistema de numeración es su base, es decir, el número de símbolos diferentes que utiliza.

El **sistema de numeración decimal**, tiene 10 símbolos diferentes, del 0 al 9, y al ser tambiñen un sistema de numeración posicional, el valor de cada símbolo queda determinado por su valor absoluto y su posición dentro del conjunto.

En el **sistema de numeración binario** (en base 2) solamente pueden utilizarse dos símbolos diferentes: el 0 y el 1.

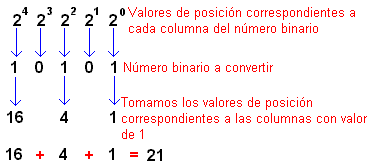
**DECIMAL BINARIO HEXADECIMAL**

## Medidas de la información

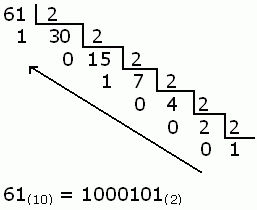
* Cuarteto = 4 bits.
* Byte u octeto = 8 bits.
* Kilobyte = 1 Kb. = 210 bytes = 1024 bytes.
* Megabyte = 1 Mb. = 1024 Kb.
* Gigabyte = 1 Gb. = 1024 Mb.
* Terabyte = 1 Tb. = 1024 Gb.

## Conversión de sistemas

### Binario a decimal



### Decimal a binario



### Binario a hexadecimal

Se agrupan en cuartetos binarios el numero a transcribir. Se transforma cada cuarteto a decimal y ese decimal a hexadecimal.

### Hexadecimal a binario

Se transforma cada caracter hexadecimal en el caracter decimal. Se transforma ese caracter a un cuarteto binario y se agrupan en octetos.

## Sistemas de codificación

Para codificar la información en formato binario se utilizan diversos métodos, que dependen del tipo de dato a codificar:

Para las instrucciones se utilizan códigos preestablecidos, de tal forma que a cada una de ellas le corresponde un número determinado de bytes y una secuencia concreta de ceros y unos.

Para codificar datos alfanuméricos el sistema más difundido es el ASCII que consiste en utilizar un byte para codificar cada carácter. Así, el número máximo de caracteres a codificar es de 256, que es el número máximo de combinaciones diferentes de ceros y unos que se pueden formar con 8 bits.

Para codificar datos numéricos se utilizan varios sistemas. Entre los más difundidos seencuentran:

• **Binario puro**: se utilizan 32 bits de los cuales el de más a la izquierda seutiliza para representar el signo, 0 si es positivo y 1 si es negativo, y los 31 restantes para representar el valor del número.

• **Decimal desempaquetado**: se utilizan tantos bytes como dígitos tiene el número decimal a codificar. Cada byte tiene los cuatro bits de la izquierda con unos (1111), se denominan bits de zona, y los cuatro de la derecha corresponden a la codificación en binario del dígito decimal correspondiente. El byte de más a la derecha, en vez de tener cuatro unos en los bits de zona, tiene “1100” si el número es positivo y “1101” si el número es negativo.

• **Decimal empaquetado**: cada dígito decimal se codifica en un cuarteto y en el cuarteto de más a la derecha se codifica el signo, con las mismas combinaciones que para el caso anterior. Se utiliza un número entero de bytes aunque cuando el número de dígitos decimales es par el cuarteto de más a la izquierda lleve el valor “0000”.

• Existen **más métodos** que permiten, por ejemplo, la codificación de cantidades en notación científica o coma **flotante** utilizando 4 bytes (simple precisión) o **double** 8 bytes (doble precisión).

# Estructura de un sistema informático

Un Sistema Informático es el conjunto de elementos que permiten realizar un tratamiento automático de la información. El tratamiento de la información consiste en la realización de una secuencia de operaciones con datos, efectuando funciones de entrada, control, procesamiento, almacenamiento y salida de los mismos. Se compone de la parte humana y la parte máquina:

## Componente humano

Se compone de las personas que tienen interacción con el sistema. Por ejemplo, analista, programador, diseñador, usuario…

## Componente informático

### Parte física

Denominado **hardware**. Está formado por los ordenadores y los periféricos. Su función es proporcionar la capacidad de procesar eficientemente la información, facilitar la entrada y salida de datos y posibilitar la comunicación con otros sistemas.

### Parte lógica

Se compone de todo el software utilizado por el sistema (desde la BIOS a Microsoft Office) La función del sistema operativo es la comunicación entre el usuario y los programas con el hardware del ordenador. Ejemplos de sistemas operativos: Ubuntu, Windows.

**Hay dos tipos de interfaces**:

* **De linea de comandos**: permiten introducir instrucciones para realizar las tareas oportunas. Como ejemplos tenemos los diferentes shell de Linux o el cmd.exe de Windows.
* **Gráficos**: permiten operar sobre las aplicaciones y componentes del sistema operativo mediante elementos visuales, siendo más simple para el usuario. Algunos ejemplos pueden ser los entornos de escritorio son gnome o Windows Aero.

# Funciones de un sistema operativo

De forma más detallada se puede decir que las funciones que cumple un sistema operativo son las siguientes:

* **Administración del procesador**: distribuyendo los tiempos que el procesador o los procesadores dedican a cada programa en ejecución.
* **Gestión de la memoria RAM**: asignando a cada usuario y a cada programa la memoria suficiente para que realicen su trabajo. Si la memoria RAM es insuficiente, hacen uso y gestionan la memoria virtual. La memoria virtual es un espacio de disco duro que se utiliza como ampliación de la memoria RAM. El uso de estas memorias hace posible la ejecución de algunos programas aunque sea de forma mucho más lenta que cuando se ejecutan en memoria RAM.
* **Gestión de entrada/salida**: permitiendo a los programas el acceso a los dispositivos de entrada/salida de información, a través de los drivers.
* **Gestión de ejecución de aplicaciones**: asignando a cada aplicación los recursos de memoria, tiempos de procesador, acceso a dispositivos, etc. necesarios para que la aplicación se ejecute.
* **Administración de autorizaciones**: encargándose de que cada programa sea ejecutado, cada dispositivo utilizado, cada fichero accedido, ... de acuerdo con las autorizaciones del usuario.
* **Gestión de archivos**: permitiendo la lectura/escritura en el sistema de ficheros acorde a las autorizaciones de acceso de los usuarios y aplicaciones.
* **Informar acerca de su propio funcionamiento**: registrando las anomalías, avisos e indicadores que permitan diagnosticar el funcionamiento correcto o anómalo del mismo. Ubuntu almacena en la carpeta /var/log ficheros que registran diferentes eventos sucedidos durante el arranque del sistema o ejecución de alguna aplicación.

# Tipos de sistemas operativos

Las principales características para categorizar un sistema operativo son:

## Número de usuarios

* + **Monousuario**: El ordenador con todos sus recursos solamente puede ser utilizado por un usuario. Si otro usuario quiere realizar cualquier operación tiene que esperar a que el anterior termine.
  + **Multiusuario**: Varios usuarios pueden utilizar a la vez el sistema operativo y los recursos de un ordenador.

## Número de programas o tareas.

* + **Monotarea**: Solamente puede ejecutarse una tarea a la vez y las demás deberán esperar a que termine la que se encuentra en ejecución. Cada tarea dispone de todos los recursos del sistema para su desarrollo.
  + **Multitarea**: Pueden ejecutarse varias tareas a la vez. La CPU se encarga de repartir los tiempos de utilización del procesador entre los programas en ejecución.

## Número de procesadores.

* + **Monoproceso**: El sistema operativo solamente puede gestionar un procesador. Evidentemente, este tipo de sistemas operativos se instalarán en equipos que únicamente dispongan de un procesador.
  + **Multiproceso**: El sistema operativo puede gestionar varios procesadores. Asignándole las tareas al siguiente si el anterior no puede responder. En todo momento se utilizan todos los procesadores para todos los programas si es necesario.

## Tiempo de respuesta.

# Gestor de arranque

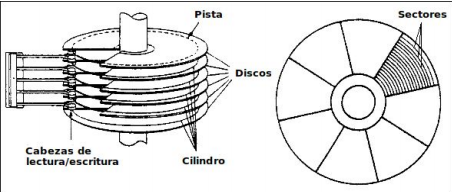
La función fundamental de los programas gestores de arranque es facilitar la elección del sistema operativo de inicio del sistema, en aquellos equipos en los que se han instalado varios sistemas operativos o diferentes versiones de núcleo para un mismo sistema operativo. De lo contrario, el gestor de arranque cargará el único instalado.

El gestor por defecto de GNU/Linux es GRUB.

# Estructura de un disco duro \*

## Estructura física

En cuanto a la estructura física, se puede suponer que un **disco duro está formado por una serie de discos con un eje de giro común**. Cada **disco tiene dos caras** (**heads**), **cada cara está compuesta por una serie de pistas concéntricas** (**tracks**) y **cada pista está dividida en sectores**. Las caras están numeradas desde la superior (cero) hasta la inferior, las pistas desde la más interna (la cero) hasta la más externa y los sectores desde una marca en adelante empezando desde el número 1. Un cilindro lo forman las pistas de las dos caras de todos los discos que tienen el mismo número de pista, es decir, el cilindro cero está compuesto por todas las pistas número cero.



Algunos datos interesantes relacionados con la estructura del disco son:

* Un sector físico es la cantidad mínima de información que se lee o se graba en una operación de lectura/escritura del dispositivo. Un disco duro no puede leer ni grabar una parte de un sector.
* Una partición debe ocupar un número entero de cilindros.
* La capacidad de los sectores difiere de unos discos a otros.
* Para identificar un sector pueden utilizarse varias notaciones:
* **CHS** (Cylinder-Head-Sector): consiste en especificar el nº de cilindro, de cabeza y de sector teniendo en cuenta que siempre se empieza a numerar por el cero.
  + **LBA** (Logical Block Address): consiste en especificar el nº absoluto de sector, siendo el primero el nº cero.

**Fórmula para hallar número total de sectores: Sectores \* cabezas \* cilindros.**

Así, si un disco tiene 25 cilindros, 10 cabezas y 64 sectores por pista, tiene un total de 16.000 sectores (64 \* 10 \* 25), que pueden numerarse del 0 al 15.999.

Cada cilindro tendrá 640 sectores y el sector nº 641 será el primer sector de la pista, de la primera cabeza, del segundo cilindro (cilindro = 1, cabeza = 0, sector = 1).

## MBR

El **Master Boot Record** (**MBR**) comprende los primeros 512 bytes de un dispositivo de almacenamiento. El **MBR** no es una partición; está reservada al cargador de arranque del sistema operativo y a la tabla de particiones del dispositivo de almacenamiento.

## Particiones MBR

Existen tres tipos de particiones:

* **Particiones primarias** (de arranque): Se caracterizan porque el sistema puede arrancar desde ellas sin necesidad de ningún programa gestor de arranque. Un disco duro puede tener como máximo 4 particiones primarias. El sector de arranque de estas particiones ejecuta los primeros 512 bytes cuya función es cargar en memoria y ejecutar el primer fichero correspondiente al sistema operativo instalado en dicha partición. Si la partición es de datos el sector de arranque estará vacío. El área de datos donde el sistema operativo guarda los ficheros atendiendo a la organización del sistema de archivos.
* **Particiones extendidas** (de almacenamiento): Contiene una tabla similar al MBR e información relativa al sector de carga, así como el sector de arranque de la partición y el área de datos. Tienen como características fundamentales:
  + Para arrancar desde ellas se necesita la ayuda de un programa gestor de arranque.
  + En el caso de que exista será única.
  + Destinado a contener unidades lógicas, cuyo funcionamiento es similar al de las particiones.
* **Particiones lógicas**:
  + Sector de arranque del disco: Situado en la primera pista (pista cero) del disco. Sus funciones fundamentales son:
    - Identificar si el disco ha sido particionado, operación que se realiza comprobando si el valor hexadecimal de los bytes 511 y 512 del MBR es 55 AA (particionado) o no.
    - Contener la tabla de asignación de particiones: son los bytes 447 a 510, ambos incluidos. Estos 64 bytes están divididos en 4 elementos (entradas) de 16 bytes cada uno. Cada elemento contiene información de una partición. Por esta razón un disco no puede contener más de cuatro particiones.
    - Cargar en memoria el pequeño programa o conjunto de instrucciones que deben ejecutarse a continuación.

## Particiones GUID

Son las usadas por GPT, EFI y UEFI como mejora a las particiones MBR tradicionales.

La EFI comunica el arranque con el MBR y con el nuevo GPT (GUID Partition Table o tabla de particiones GUID) que presenta, entre otras, las siguientes ventajas:

• Soporta un número ilimitado de particiones (por defecto 128) frente a las 4 del MBR.

• Cada partición puede tener una capacidad de 9,4 ZB frente a los 2,2 TB del MBR.

• Permite seleccionar y cargar sistemas operativos sin programas gestores de arranque. Requisitos para usar particiones GUID:

• Disponer de sistema EFI (o UEFI) en vez de BIOS.

• El sistema operativo debe ser de 64 bits y soportar GUID.

• El disco debe estar particionado con GUID (GPT, EFI o UEFI).

Con este sistema de particionamiento el disco se organiza en tres zonas:

• El MBR para compatibilidad, la cabecera de GPT y la tabla de particionamiento primaria (GPT Primaria).

• El área de datos de las particiones.

• La tabla de particionamiento secundaria (GPT Secundaria) y su cabecera. El contenido es una copia de la primaria en orden inverso.

El MBR de protección y compatibilidad o LBA 0. Ocupa el primer sector del disco que puede ser de 512 bytes (están empezando a utilizarse de 4096 bytes) y su función es evitar que utilidades de disco basadas en MBR no reconozcan o dañen discos con particiones GPT.

La cabecera de la tabla GPT o LBA 1 define los bloques de disco que pueden ser utilizados por el usuario y el número (128 por defecto) y tamaño (128 bytes por defecto) de las entradas de partición que conforman la GPT. También contiene el GUID del disco, su propio tamaño (el de la cabecera) y localización (siempre LBA 1) y la localización de la cabecera y GPT secundaria (siempre en los últimos sectores del disco).

La GPT primaria se compone de entradas o bloques de información que describen una a una todas las particiones. Es típica una distribución de 128 entradas de 128 bytes cada entrada para sectores de 512 bytes, lo que permite 4 entradas por sector; pero estos datos, como puede verse, son configurables en la cabecera de la tabla.

# Secuencia de arranque

1. Una vez que el usuario enciende el ordenador, la BIOS y la inician la pantalla y el teclado y comprueban la memoria RAM. A continuación se lee la hora, la fecha y los datos de los periféricos más importantes de los valores que se encuentran en la CMOS. Entre los datos leídos se encuentra el orden de arranque de los periféricos. La última operación que realiza la BIOS o la EFI es la carga en memoria del MBR o los módulos de arranque de la partición de arranque de EFI, respectivamente.
2. Se ejecutan las instrucciones del MBR. Luego, la única función del MBR es cargar el programa o instrucciones que deben realizarse a continuación y cederle el control. La EFI realiza las funciones de la BIOS y del MBR juntos. Si se trata del MBR estándar, cargará en memoria el contenido del sector de arranque de la partición activa, que será el cargador del gestor de arranque.
3. Se selecciona el sistema operativo a iniciar en el gestor de arranque si procede, sino, se cargará el por defecto.
4. Se cargan los ficheros correspondientes al núcleo del sistema operativo y continua la carga de los demás procesos del sistema operativo hasta quedar listo para el inicio de sesión por parte del usuario.
5. Por lo general, el sistema operativo solicita un nombre de usuario y contraseña. Si ambos son correctos permite la entrada y si alguno de los dos no es válido se solicita la identificación completa.

# Niveles de ejecución

|  |  |
| --- | --- |
| Niveles de ejecución | Funcionalidad |
| 0 | Parada del sistema (system halt). |
| S | Modo monousuario desde el prompt del arranque. |
| 1 | Modo monousuario. |
| 2 | Modo multiusuario local, sin red remota. |
| 3 | Modo multiusuario con red. |
| 4 | No usado. |
| 5 | Modo multiusuario con red y con entorno gráfico. |
| 6 | Reiniciar el sistema |

Un nivel de ejecución o runlevel consiste en la ejecución de unos determinados procesos o demonios que proporcionan unos servicios concretos. En la instalación de un servidor es habitual diseñar una configuración para que en determinados momentos se puedan ofrecer determinados servicios y en otros no. Para permitir este tipo de funcionamiento, el sistema operativo nos proporciona diferentes niveles de ejecución que podremos adaptar a nuestras necesidades. Si bien se puede configurar el número de niveles de ejecución que se quiera y la funcionalidad de cada uno de ellos, generalmente los sistemas Linux nos proporcionan 7 niveles diferentes (numerados del 0 al 6) para permitir que el ordenador trabaje con diferentes configuraciones de arranque (diferentes servicios). La siguiente tabla muestra los diferentes niveles de ejecución y su funcionalidad.

# Administración de usuarios locales

La característica fundamental de los sistemas operativos multiusuario es que distintos usuarios pueden trabajar de forma independiente en el mismo sistema. Cada usuario tiene una cuenta de usuario que consiste en un nombre de usuario y una contraseña personal para entrar al sistema. Además, de forma generalizada para la práctica totalidad de los sistemas operativos, cada usuario puede tener asignado:

* Un directorio personal de trabajo para almacenar sus archivos.
* Una configuración de su entorno de trabajo.
* Derechos o privilegios que lo autorizan a realizar ciertas operaciones en el equipo.
* Permisos (reglas asociadas a un elemento u objeto, generalmente ficheros o carpetas) que regulan si tienen acceso a un objeto y de qué manera.

Cada sistema operativo dispone de una serie de derechos asignables a usuarios o grupos. Los permisos son propiedades de los objetos que pueden empezar a asignarse cuando se crea el objeto. Un aspecto fundamental en la administración de cuentas de usuario y de grupo, que simplifica enormemente la organización y el trabajo del administrador, es la posibilidad de centralizar las cuentas en determinados equipos.

# Diseño de cuentas de usuario \*

1. **Una convención de nombres**: Nombres de usuario únicos e intuitivos respecto al equipo, teniendo en cuenta la longitud mínima y máxima posible del nombre admitida por el sistema operativo. Así podrá incluir caracteres que hagan referencia a su puesto de trabajo, a su temporalidad, etc.
2. **Los requisitos de las contraseñas**: Crear contraseñas robustas y poco intuibles por terceros. Algunas recomendaciones para la instalación de varios equipos son:
   * Asignar siempre una contraseña al usuario Administrador/root.
   * Cambiar el nombre de la cuenta Administrador/root.
   * Determinar si los usuarios pueden o no cambiar su contraseña.
   * Determinar si una cuenta de usuario debe caducar o no, dependiendo de la temporalidad del usuario.
   * Informar a los usuarios sobre el correcto uso de las contraseñas a los usuarios.
3. **El horario de uso del ordenador y de conexión a la red**: Como método de protección y seguridad de los datos, estableciendo el horario de inicio de sesión y de uso cuando los usuarios requieran el acceso dentro de unos períodos de tiempo específicos.
4. **Ubicación de las carpetas de trabajo**: Al contener toda la información privada de un usuario, se podría estudiar la posibilidad de cambiar su localización
   * Realización de copias de seguridad y restauración: Se simplifica si las carpetas personales se encuentran centralizadas en un servidor.
   * Espacio ocupado: Limitar o no el espacio en función del contenido de cada usuario.
   * Uso de dispositivos removibles.
   * Rendimiento de la red: Si se trata de un flujo constante de grandes ficheros, no conviene barajar la posibilidad de centralizar.

# Usuarios predeterminados

## Ubuntu

**Root**: Usuario administrador total, el cual posee privilegios totales sobre el sistema. Para llamar a este usuario se ejecuta la instrucción *sudo*, aunque no se puede acceder sesión de manera directa con este usuario (sólo con la mencionada instrucción *sudo*).

**Usuario inicial**: Su cuenta se configura en la instalación del sistema operativo y posee privilegios de administrador sobre el sistema. Su directorio de trabajo se sitúa en */home/[usuario]* y puede utilizar la instrucción *sudo*.

**Usuario invitado**: Cuenta de acceso temporal y memoria volátil, la cual pierde sus datos al reiniciar el equipo. Su directorio de trabajo se sitúa en */tmp* y no puede ejecutar la instrucción *sudo*.

## Windows server 2008

**Administrador**: Esta cuenta tiene control total del equipo y puede asignar derechos de usuario y permisos de control de acceso a los usuarios según se necesite. Puede estar habilitada o deshabilitada y siempre es miembro del grupo Administradores. No se puede eliminar ni quitar del grupo Administradores pero se puede cambiar el nombre (recomendable) y deshabilitar.

Aunque la cuenta Administrador esté deshabilitada, si se inicia el equipo en modo seguro podrá utilizarse.

**Invitado**: Utilizada por los usuarios que no tienen cuenta real en el equipo. No requiere contraseña y se encuentra deshabilitada de forma. Se le pueden asignar derechos y permisos como a cualquier otra cuenta y pertenece al grupo predeterminado Invitados, lo que le permite iniciar sesión en un equipo.

# Seguridad de cuentas de usuario \*

Para incrementar la seguridad del equipo y evitar al máximo los momentos de vulnerabilidad se recomienda:

1. No iniciar sesión con una cuenta con privilegios de administrador. Se aconseja iniciar la sesión como un usuario normal y hacer uso, siempre que se pueda, de instrucciones que permitan, al menos temporalmente, realizar tareas administrativas.
2. Incluir las menos cuentas de usuario posibles en los grupos con privilegios de administradores.
3. Mantener deshabilitada la cuenta asignada para los usuarios Invitados o Anónimos puesto que suelen ser cuentas que no requieren contraseña y cualquier usuario puede utilizarlas, con el consiguiente riesgo para la seguridad.
4. Mantener deshabilitada la cuenta del usuario Administrador, por ser la cuenta que todos los intrusos desean desproteger.
5. Incluir en los grupos con privilegios de administradores a los usuarios que gocen de la confianza adecuada para garantizar la seguridad necesaria.

# Seguridad de contraseñas \*

Consejos para el buen uso de contraseñas:

* Número mínimo de caracteres (recomendado 8).
* Que no contenga palabras reconocidas por terceros.
* Que no contenga palabras completas que puedan estar incluidas en cualquier tipo de diccionario.
* Que sea significativamente diferente a las usadas con anterioridad.
* Que contenga símbolos de los diferentes tipos: letras mayúsculas, letras minúsculas, números y caracteres especiales permitidos (<>&%$()=?¿ …).

Para recordar contraseñas que cumplan con las condiciones de seguridad pueden emplearse métodos como:

* Usar las iniciales de las palabras de una frase fácil de recordar, por ejemplo: “Yo nací el 25 de marzo de 1963” podría convertirse a “Yne25-M/1963”.
* Trasladar las letras de una palabra a números, caracteres especiales y errores ortográficos, por ejemplo, Santiago puede convertirse en 5@Mt1@j0.

Si aún así, se cree conveniente anotar la contraseña para que no se olvide, debe hacerse en lugar seguro y sin indicar que es una contraseña.

# Perfiles locales

El perfil de un usuario está compuesto por los ficheros que contienen la información de configuración para ese usuario específico. Esa información puede ser relativa a la configuración del escritorio, las conexiones persistentes a carpetas e impresoras de red y la configuración de aplicaciones.

Generalmente el perfil inicial de un usuario se genera en el momento de la creación del usuario o la primera vez que el usuario inicia una sesión en el equipo.

En Ubuntu, estos perfiles están ubicados en /home/[usuario], y en Windows wn C:/Users/[usuario]

* Descargas
* Documentos
* Escritorio
* Imágenes
* Música
* Plantillas
* Público
* Vídeos

En Ubuntu…

**/etc/skel**: el contenido de este directorio se le copia a todos los usuarios que se creen.

**Ficheros ocultos**: Comienzan por punto.

**Nautilus**: Entorno de ventanas nativo de Ubuntu.

# Gestión centralizada de usuarios

La centralización de usuarios en uno o varios equipos de la red puede facilitar de forma muy significativa el trabajo de administración y control de los equipos y de la red en su conjunto. Cuando los permisos y privilegios de los usuarios están perfectamente organizados, gestionados y controlados los problemas de la red se reducen. Ligada a la gestión centralizada de usuarios se encuentra la gestión centralizada de carpetas. Aunque son dos aspectos perfectamente delimitados, muy frecuentemente el servidor que contiene y autentica a los usuarios también contiene algunas de sus carpetas de trabajo.

Dependiendo de la funciones que se quieran centralizar pueden distinguirse tres categorías:

* **Compartir recursos**: Los dos protocolos más utilizados son NFS (UNIX) y SMB/CIFS (Windows). Las funciones principales son: centralizar los directorios de inicio de sesión de los usuarios, la realización de copias de seguridad, el espacio requerido, los dispositivos removibles disponibles, la carga de la red, etc.
* **Autenticación y permisos de acceso**: Permite establecer y comprobar la identidad de un usuario o equipo de la red y determinar los permisos de acceso a recursos. Dos de los servicios de autenticación más utilizados son NIS y Kerberos.
* **Servicios de directorio**: Centralizan toda la información susceptible de compartir a través de la red, incluyendo las dos funciones anteriores. Los dos servicios más utilizados son LDAP en sistemas Linux y Active Directory de Microsoft.

# Administración de grupos locales

La función de los grupos es simplificar la administración de usuarios. Los dos casos fundamentales para los que se generaliza el uso de grupos son:

* **Administración de recursos**: se crean grupos según los diferentes permisos que se puedan asignar a los diferentes recursos. Posteriormente, si se desea, se podrá incluir a los usuarios en uno de estos grupos.
* **Administración de usuarios**: se crean los grupos según las características homogéneas de los usuarios.

El objetivo principal consiste en limitar el acceso a elementos de forma ordenada y simple.

# Grupos predeterminados

Además de los grupos creados por el usuario existen otros grupos especiales que sirven fundamentalmente para:

* Establecer los privilegios y controlar los accesos a los dispositivos mediante los permisos adecuados a ficheros de dispositivos situados en /dev. A este tipo pertenecen los siguientes grupos predeterminados:
  + **plugdev**: permite acceder automáticamente a dispositivos de almacenamiento externo.
  + **admin**: permite administrar el sistema.
  + **video**: permite capturar vídeo de TV o de webcams, y usar aceleración 3D.
  + **sambashare**: permite compartir archivos con la red local.
  + **dip**: permite conectar a internet con un modem.
  + **netdev**: permite conectarse a redes ethernet e inalámbricas.
  + **lpadmin**: permite configurar impresoras.
  + **fax**: permite enviar y recibir faxes.
  + **adm**: permite monitorizar los registros del sistema.
  + **fuse**: permite montar sistemas de archivos es espacio de usuario.
  + **audio**: permite usar dispositivos de sonido.
  + **dialout**: permite usar modems.
  + **cdrom**: permite usar unidades de CD-ROM.
  + **tape**: permite usar unidades de cinta.
  + **vboxusers**: permite usar VirtualBox.

Para dar acceso a un usuario a uno de estos servicios, basta con añadirlo al grupo adecuado. Permitir el uso de algunas aplicaciones instaladas. Al instalar algunas aplicaciones, automáticamente se crea un grupo de tal forma que aquellos usuarios que quieran tener acceso a la aplicación deben ser incluidos en dicho grupo. Los grupos se encuentran en **/etc/group,** y para conocer los grupos a los que pertenece un usuario puede utilizarse la instrucción: **groups**.

# Derechos y permisos \*

## Tipos de usuarios:

* + **Owner**: Es el usuario que crea el fichero o directorio.
  + **Group**: Son los demás usuarios que pertenecen al mismo grupo que el propietario.
  + **Other**: Los excluyentes a los grupos anteriores.

## Tipos de permisos:

Su significado es diferente para los ficheros que para los directorios.

* + **Read**: Permite ver leer el contenido del fichero o del directorio.
  + **Write**: Permite modificar el fichero. Aplicado a directorios permite crear nuevos ficheros o borrar los existentes.
  + **Execution**: Permite ejecutar el fichero (si es ejecutable) o situarse en el directorio (hacer que sea el directorio activo).

## Ficheros de configuración

Los usuarios se guardan en el fichero **/etc/passwd**, a razón de un usuario por línea, con la información:

cuenta:contraseña:UID:GID:nombre\_completo:directorio:intérprete

El significado de cada campo es el siguiente:

* **Cuenta**: La cuenta de usuario.
* **Contraseña**: Aparece una”x” puesto que las contraseñas, cifradas, se encuentran en el fichero /etc/shadow.
* **UID**: Número de identificación asignado al usuario.
* **GID**: Número de identificación del grupo principal al que pertenece el usuario.
* **Nombre\_completo**: Nombre completo del usuario.
* **Directorio**: Directorio de trabajo del usuario.
* **Intérprete**: Nombre del programa intérprete de comandos que utiliza el usuario.

Se tomarán valores por defecto del fichero **/etc/adduser.conf** a la hora de crear un nuevo usuario.

Los datos correspondientes a los grupos se encuentran en el fichero **/etc/group**. Cada línea de este fichero contiene:

nombre\_grupo:clave:GID:otros\_usuarios

El significado de cada campo es el siguiente:

* Nombre\_grupo: Nombre identificativo del grupo.
* Clave: Contraseña del grupo.
* GID: Número único e identificador del grupo.
* Otros\_usuarios: Se utiliza cuando un grupo contiene más de un usuario y se especifican separados por comas.

El directorio **/etc/skel** contiene las carpetas y ficheros que se incluirán de forma automática en la carpeta de trabajo al crear un usuario.

Al añadir un usuario, por defecto se crea un grupo con el mismo nombre y los mismos privilegios que el usuario

## Comandos útiles para el manejo de usuarios:

* **groups**: Muestra los grupos a los que pertenece el usuario activo.
* **chown**: Permite cambiar el propietario y el grupo de los ficheros y directorios.

chown usprueba03:gr-prueba34 fich

* **chgrp**: Permite cambiar el grupo de los ficheros y directorios.

chgrp gr-prueba5 fich

* **chmod**: Permite cambiar los permisos de los ficheros y directorios.

chmod g+w fich

chmod 764 fich

## Formas de permisos

### Simplificada

* El primer carácter identifica a quien afecta el cambio, pudiendo ser: u (propietario), g (grupo), o (otros) y a (todos). Puede utilizarse una combinación de ellos.
* El segundo carácter expresa si se añade el permiso (+), se quita (-) o se establece (=).
* El tercer carácter identifica el permiso, que será lectura (r), escritura (w) o ejecución (x). Puede utilizarse una combinación de varios caracteres

Ejemplo:

g+w

### Numérica

Cada grupo de tres caracteres tendrá el carácter correspondiente al permiso para expresar la posesión del mismo, o un guión que expresa la ausencia, por ejemplo “rwx rw- r--”. Si se cambian las letras por números 1 y los guiones por 0 se obtiene una notación en binario de la forma: 111 110 100. Pasando cada grupo de tres dígitos a decimal se obtiene “7 6 4”. Algunos ejemplos más serían:

rw- rw- r-- → 664

A los ficheros y directorios que crea un usuario se les asigna unos permisos por defecto que pueden ser modificados mediante la instrucción “**umask**”. Esta máscara es restada por defecto a todo elemento creado en el sistema. Su instrucción asigna una máscara de manera temporal. Si se desea que esta máscara sea permanente, se ha de modificar en **/home/[*usuario*]/bashrc**. Para cambiárselos a todos los usuarios se debería editar el fichero **/etc/profile**.En umask se utiliza una expresión numérica que cumple la relación:

**Permisos\_reales** = Permisos\_base - Expresión\_umask

## Ampliación de derechos

* **El bit setuid**: Existen situaciones en las que los derechos básicos son demasiado restrictivos. En estos casos, existen en Linux opciones adicionales que permiten cambiar temporalmente el usuario y grupo de identidad actual para realizar una determinada acción. Setuid (Set User ID) es un atributo especial de archivo que el sistema regula, con el fin de que se ejecuten los programas seleccionados con una determinada identificación de usuario previamente indicada.
* **El bit setgid**: El atributo setuid es válido para usuarios, pero existe un atributo equivalente para grupos: setgid. Cuando se asigne a un programa, este se ejecutará con la identificación de grupo definida, independientemente del usuario que lo haya arrancado. Por eso todos los archivos y subdirectorios que se crean dentro de este directorio pertenecen al mismo grupo que el directorio. Veamos el siguiente ejemplo:
* **El bit sticky**: Aquí se debe diferenciar si se trata de un programa ejecutable o de un directorio. En el caso de un programa, un archivo marcado con este bit se carga a la memoria RAM para no tener que obtenerlo del disco duro cada vez que se utiliza. Hoy en día, este atributo se usa en raras ocasiones porque los discos duros modernos cumplen con creces los requisitos de velocidad. Ahora bien, si se asigna este atributo a un directorio, este impide que los usuarios borren los archivos. (En directorios con sticky bit, los usuarios sólo puede eliminar archivos que posean ellos mismos). Los ejemplos típicos son los directorios “/tmp” y “/var/tmp”:

drwxrwxrwt 15 root root 4096 2010-03-11 11:40 tmp

# Listas de control de acceso

Una **lista de control de acceso** o ACL (del inglés, access **control** list) es un concepto de seguridad informática usado para fomentar la separación de privilegios. Es una forma de determinar los permisos de **acceso** apropiados a un determinado objeto, dependiendo de ciertos aspectos del proceso que hace el pedido. De manera tradicional, para cada objeto en Linux se definen tres grupos de permisos. Estos grupos reflejan los permisos de lectura (r), escritura (w), y ejecución (x) para las tres clases de usuarios: propietario del archivo (owner), grupo (group) y el resto (other).

# Consideraciones previas a la instalación del sistema operativo

Antes de proceder a la instalación de un sistema operativo deben tenerse en cuenta:

* **Los requisitos hardware mínimos exigidos por el sistema operativo**: Generalmente, todos los sistemas operativos, en todas sus versiones, ofrecen información acerca de las características hardware mínimas que debe cumplir el equipo en el que se va a instalar. Estas características mínimas suelen hacer referencia a los elementos:
  + Tipo y velocidad del procesador.
  + Capacidad de la memoria RAM.
  + Capacidad del disco duro. Número y tipo de particiones.
  + Características de la tarjeta de vídeo.
  + Dispositivos necesarios (lector de DVD, tarjeta de red, etc.).
* **Características del equipo**: Además de saber si el equipo cumple con los requisitos de hardware exigidos para el funcionamiento del sistema operativo, hay que conocer:
  + Los discos , particiones, tipos y sistemas operativos instalados previamente.
  + Las particiones que conlleva la instalación del nuevo S.O.
  + El diseño final de particiones, con capacidades y tipos, que se pretende conseguir.
* **Licencias**: Es necesario conocer el tipo de licencia requerido para poder utilizar el S.O. conforme a la legislación vigente.
* **Gestores de arranque**: Se ha de tener en cuenta la compatibilidad del gestor de arranque respecto al sistema operativo a instalar.
* **Copias de seguridad**: Es imprescindible realizar copia de seguridad de los datos de un disco antes de proceder a instalar en él un S.O. La instalación de sistemas operativos es la operación que con más frecuencia produce desastres en la información que previamente contenía el disco. Un reparticionamiento erróneo o un problema de compatibilidad puede producir la pérdida de la información de una partición.
* **Aplicaciones a instalar**: El tipo de aplicaciones a instalar y sus requerimientos de capacidad determinarán el S.O. y las capacidades de las particiones a utilizar.
* **Particiones diferentes para el sistema y para los datos**: Se aconseja tener en particiones diferentes el S.O. y los datos de los usuarios. Esta disposición facilita la realización de copias de seguridad y, además, en caso de tener que reinstalar el sistema operativo, no es necesario tener que volcar todos los datos a un dispositivo externo y reubicarlos otra vez después de la instalación. El inconveniente es tener que calcular el espacio que se destinará a cada una de las dos particiones.
* **Métodos de instalación soportados**: Además de la instalación típica en la que el usuario administrador realiza la instalación desde los DVD's del sistema, existen circunstancias que pueden hacer aconsejable el uso de técnicas de instalación más sofisticadas. Un caso típico es la instalación del S.O. en un gran número de equipos.

Algunas de las técnicas más utilizadas son:

* + **Utilización de ficheros con la información requerida por el sistema durante la instalación**. Este tipo de instalaciones se llaman desatendidas porque no requieren de la presencia física de un usuario durante la instalación. La información requerida durante la instalación es tomada de un fichero preparado con antelación. El usuario únicamente inicia el proceso de instalación, que suele realizarse mediante una instrucción que contiene una opción para especificar el fichero que contiene las respuestas.
  + **Instalación desde un servidor**. El software del sistema operativo se encuentra en un equipo de la red y el usuario solamente necesitará disponer de un sistema de inicio que establezca la comunicación con el servidor de instalaciones. Evita que los usuarios dañen el soporte donde se encuentra el software original.
* **Duplicación de discos**. Como en el caso anterior se necesita disponer de un sistema de inicio del equipo, que se comunicará con un servidor que contendrá una copia de una instalación realizada anteriormente. Esta técnica es muy eficaz cuando los equipos donde se va a realizar la instalación tienen las mismas características de hardware. La diferencia con el método anterior es que aquí no se produce una instalación real, no se descomprime software ni se realizan comprobaciones de dispositivos, etc, solamente se copian ficheros.
* **Versiones Live**: Cuando lo que se necesita es realizar pruebas con el S.O. para comprobar sus características, o se quiere conservar su integridad sin modificaciones, puede hacerse uso de una versión Live en DVD o lápiz USB, que permite el inicio del sistema y su funcionamiento sin realizar instalaciones en el disco duro.
* **Máquina virtual**: para casos similares al anterior (realización de pruebas) también puede considerarse la posibilidad de manejar el S.O. en una máquina virtual. Con este método el S.O. se instala en un fichero que puede ser borrado y copiado en cualquier momento con mayor facilidad que tener que trabajar con particiones. Un ejemplo se utiliza en esta documentación como laboratorio de pruebas con varios sistemas operativos.