

Identificación del Software instalado

- ▶ Para identificar el software instalado en nuestro equipo, los S.O. disponen de herramientas gráficas y comandos de consola.
- ▶ En nuestro caso vamos a ver las herramientas que disponen el S.O. Windows y Ubuntu.

- S.O. Windows: abrimos una de las siguientes consolas en modo administrador.

- Consola CMD: utilizamos wmic. Para salir del entorno pulsa ctrl+C. Si deseamos obtener un archivo con los resultados, ejecuta la siguiente sintaxis:

```
wmic /output:D:\InstalledSoftwareList.txt product get name,version
```

- Consola PowerShell en modo administrador. Para obtener un archivo de salida utiliza la sintaxis:

```
Get-WmiObject -Class Win32_Product | Select-Object -Property Name >  
D:\InstalledSoftwareList.txt
```

- S.O. Ubuntu: abrimos un terminal y escribimos la siguiente sintaxis:

```
dpkg --get-selections
```

Si además queremos mandar el resultado a un archivo de salida:

```
dpkg --get-selections > fichero
```

Identificación del software - Windows

- ▶ La herramienta gráfica más eficaz para visualizar los programas instalados en el sistema es “Programas y características”, la cual es dentro de la categoría “Programas” dentro del panel del control.
- ▶ Con la consola CMD utilizamos el comando wmic.

Identificación del software - ubuntu

- ▶ Abrimos una consola y ejecutamos el comando `dpkg --get-selections`

Administración de Discos

- ▶ La administración de discos nos permiten realizar operaciones sobre los dispositivos de almacenamiento por bloques (discos duros, pendrives, memorias flash, etc...) instalados en el sistema.
- ▶ Las operaciones básicas son: crear, formatear, eliminar y redimensionar particiones. También, dependiendo del software utilizado y del S.O. podemos crear particiones básicas y dinámicas, además de establecer un sistema RAID.
- ▶ Estas operaciones se pueden realizar en entorno gráfico o mediante comandos. En éste último caso tendríamos:
 - S.O. Windows: comando diskpart para la consola CMD.
 - S.O. Ubuntu: comando fdisk.

Administración de Discos - Ubuntu

- ▶ Las versiones actuales de Ubuntu traen la aplicación “Discos” que nos permiten gestionar los dispositivos de almacenamiento por bloques de forma gráfica.
- ▶ Hay otras herramientas, como por ejemplo “gparted” que podemos instalar, ya que no viene instalada por defecto. Ésta última herramienta, es utilizada en diversas distribuciones de Linux.
- ▶ Se puede instalar desde el centro de software o mediante la sintaxis:

```
# apt-get install gparted
```

Administración de Discos - Ubuntu

- ▶ Buscamos la aplicación discos en el categoría “Buscar en el equipo”.
- ▶ La interfaz obtenida nos mostrará los dispositivos montados en el sistema. Por defecto vendrá seleccionado el disco duro donde está instalado el sistema operativo.

Administración de Discos - Ubuntu

- ▶ Buscamos la aplicación discos en el categoría “Buscar en el equipo”.
- ▶ La interfaz obtenida nos mostrará los dispositivos montados en el sistema. Por defecto vendrá seleccionado el disco duro donde está instalado el sistema operativo.

Particiones

- ▶ Una partición es una división lógica en un dispositivo de almacenamiento de tipo bloque (disco duro, pendrive o memoria flash) que funciona como una unidad física separada. Cada partición tiene asociado su propio sistema de archivos.
- ▶ Hay dos tipos de esquemas de particionado:

- MBR (Master Boot Record): conocido por registro de arranque maestro, está ubicado en el primer sector del dispositivo de almacenamiento. Contiene el cargador del sistema operativo y la tabla de particiones.

Utiliza un modo de direccionamiento basado en cilindro-cabezal-sector (CHS), el cual limita el número de particiones primarias a 4 o 3 primarias y una extendida.

- GPT (Guid Partition Table): forma parte del estándar UEFI (Unified Extensible Firmware Interface) que tiende a reemplazar a la tradicional BIOS. GPT usa moderno modo de direccionamiento lógico (LBA, logical block addressing) que sustituye al tradicional MBR, de forma que numera los bloques según un índice (el primer bloque el LBA0, el segundo LBA1, etc...)

Este modo nos ofrece una serie de ventajas frente al MBR como: interfaz gráfica más moderna, sistema de inicio seguro, mayor velocidad de arranque, soporte para discos mayores de 2 TB y hasta 128 particiones.

Particionado MBR

El modelo de particionado apoyado en el MBR tradicional utiliza los siguientes tipos de particiones:

- ▶ **Primaria:** es reconocida por la BIOS para iniciar el sistema operativo, por ello dispone de un sector de arranque (Boot Sector). Puede haber 4 como máximo por disco duro. Una de ellas se denomina activa, pues será la que se cargue por defecto. Se le asigna un sistema de archivos y una letra de unidad.
- ▶ **Extendida:** es una partición primaria especial, que sirven de contenedor a las particiones lógicas. Se utilizan para dividir el disco en más de cuatro particiones. No se le pueden asignar un sistema de archivos ni una letra de unidad.
- ▶ **Lógica:** es una división de una partición extendida. Se le asigna un sistema de archivos y una letra de unidad. En su origen, no se podía instalar sistemas operativos en ella, aunque los actuales S.O. ofrecen esta posibilidad. Teóricamente, puede haber hasta 32 particiones lógicas, aunque esta cantidad puede estar restringida por el sistema operativo.

Sistemas RAID

- ▶ Un sistema RAID (Redundant Array of Independent Disk) consiste en interconectar varios discos duros o particiones de éstos por medio de controladoras, software o combinación de ambas, con la finalidad de garantizar la integridad de los datos.
- ▶ Los tipos de sistemas RAID más usuales son:
 - RAID 0
 - RAID 1
 - RAID 5

Sistemas RAID

- ▶ RAID 0: se consideran todos los discos del sistema como uno solo, de forma que la información se reparten entre ellos y no se establece ningún tipo de redundancia.
 - Ventajas: proporciona alto rendimiento, tiempos de acceso bajos, posibilidad de acceso en paralelo, y se emplea todo el disco.
 - Inconvenientes: no es un verdadero sistema RAID ya no garantiza la integridad de los datos. Un fallo en un disco duro supone la pérdida de la información almacenada en él.

Sistemas RAID

- ▶ RAID 5: es un sistema que integra códigos de error mediante paridad, generados a partir de los datos almacenados en el disco. Por ello, estos códigos no se guardan en el mismo disco que los datos asociados.
 - Ventajas: alto rendimiento en aplicaciones que requieran gran velocidad. No utiliza ningún disco para almacenar solo códigos de paridad.
 - Inconvenientes: bajo rendimiento en escritura, y se requieren un mínimo de tres discos.

Sistemas RAID

- ▶ Un sistema RAID (Redundant Array of Independent Disk) consiste en interconectar varios discos duros o particiones de éstos por medio de controladoras, software o combinación de ambas, con la finalidad de garantizar la integridad de los datos.
- ▶ Los tipos de sistemas RAID más usuales son:
 - RAID 0
 - RAID 1
 - RAID 5

Discos Básicos y Dinámicos

- ▶ Pueden coexistir en los sistemas Windows Server 2003, 2008, 2012 y 2016, y en estaciones de trabajo con Windows XP o posteriores.
- ▶ Disco Básico: es un disco físico con sus correspondientes particiones primarias, extendida y lógicas. Dichas particiones se conocen como volúmenes básicos. Las versiones actuales de Windows Server no soportan los volúmenes básicos multidiscos, es decir un volumen básico que ocupe dos o más particiones de diferentes discos.
- ▶ Disco Dinámico: es un disco físico que contiene uno o varios volúmenes dinámicos creados por el sistema operativo. Un volumen dinámico está constituido por una parte de uno o varios discos físicos, funcionando como una unidad independiente. Los volúmenes dinámicos no pueden ser particionados de igual forma que los volúmenes básicos, sólo pueden contener volúmenes distribuidos, reflejados, seccionados (RAID 0) y RAID 5.

Tipos de Conjuntos de Volúmenes de Disco

► Conjunto de Volúmenes.

- Es la unión de varias áreas de espacio disponible que pueden estar en uno o varios discos duros.
- Sólo se puede aplicar sobre discos dinámicos.
- Utiliza el sistema de archivos NTFS en Windows y tendrá asignada una letra de unidad para su acceso.
- Es posible ampliar su tamaño sin provocar pérdida de la información, pero no reducirlo, por lo tanto sólo se podría eliminar.
- Se corresponde con los volúmenes distribuidos.

Tipos de Conjuntos de Volúmenes de Disco

► Conjunto de Espejos

- En conjunto de espejos se utilizan dos particiones de dos discos distintos que son configurados para que sean idénticas.
- Sólo se puede aplicar sobre discos dinámicos y utilizan el sistema de archivo NTFS en Windows.
- La partición espejo no aparece en la Administración de discos, sólo refleja los datos de la otra partición, y entrará en funcionamiento cuando falle la primera.
- Se corresponde con RAID 1. Se corresponde con volumen reflejado.

Tipos de Conjuntos de Volúmenes de Disco

- ▶ Conjunto de Bandas.
 - Un conjunto de bandas es la unión de dos o más áreas de espacio disponibles situadas en dos o más discos duros, que a su vez están divididas en bandas.
 - Sólo se puede aplicar sobre discos dinámicos, de forma que en cada disco se creará una partición del mismo tamaño, a la que aplicamos el sistema de archivos NTFS si utilizamos Windows.
 - El volumen resultante tendrá asignado una letra de unidad.
 - Hay dos tipos:
 - Con paridad.
 - Sin paridad.

Tipos de Conjuntos de Volúmenes de Disco

- Conjunto de Bandas sin Paridad.

Un conjunto de bandas sin paridad dividirá cada uno de los discos en partes pequeñas denominadas bandas.

Por ejemplo: si hay 4 discos duros y cada uno tiene 10 bandas, obtendremos 10 filas de 4 bandas cada una. Un archivo se guardará distribuyendo su contenido en la primera banda disponible de cada disco duro, es decir ocupando la primera fila de bandas disponible.

Esta técnica permite un acceso más rápido a la información, pero provoca una pérdida de la información ante un fallo de un disco duro. Se corresponde con un RAID 0, por lo cual su seguridad es baja, y su equivalente en los discos dinámicos es volumen seccionado.

Tipos de Conjuntos de Volúmenes de Disco

- Conjunto de Bandas con Paridad.

Esta técnica utiliza una banda para guardar la información de paridad correspondientes a otras bandas.

Ejemplo: 5 discos con 10 bandas cada uno. Obtendremos 10 filas de 5 bandas, siendo una de ellas la de paridad. Si se pierda la información de una banda, se puede recuperar utilizando la banda de paridad, aunque esto provocará una disminución de la velocidad de almacenamiento.

Otro inconveniente es la disminución del espacio para guardar la información en un porcentaje igual al número de discos que forman parte del conjunto de bandas.

Si hay 5 discos se perderá un 20%, si son 4 un 25 %

Otro factor a tener en cuenta es la cantidad de memoria RAM extra necesaria, un 25% del total. Esta técnica se corresponde con un RAID 5 y su equivalente en los discos dinámicos es volumen RAID 5.

Copias de seguridad

- ▶ Las copias de seguridad es un mecanismo que nos protege de las pérdidas de datos ante un fallo del dispositivo de almacenamiento. Su importancia es fundamental, sobre todo si trabajamos en un entorno empresarial.
- ▶ Existen diversos programas para realizar copias de seguridad, además de los propios que traen incorporados los sistemas operativos.
- ▶ Se recomienda, guardar las copias de seguridad en dispositivos de almacenamiento externo al equipo para evitar su pérdida en caso de mal funcionamiento del equipo.
- ▶ A continuación, se muestran las herramientas propias de los sistemas Windows y Ubuntu para realizar dichas copias.

Tareas automáticas

- ▶ Una tarea automática o programada es una acción que el S.O. tiene que realizar cada cierto tiempo de forma desatendida por el usuario. Por ejemplo: diariamente, semanalmente, mensualmente, una sólo vez, al iniciar el equipo, al iniciar la sesión, cuando se registre un evento específico, etc...
 - En los sistemas Windows, para ejecutar el Programador de tareas abrimos el panel de control, sistema y mantenimiento, programador de tareas.
 - En los sistemas Ubuntu, tenemos el comando cron que ejecuta las tareas recogidas en el archivo crontab.

Planificación en comandos - Ubuntu

- ▶ Cuando queremos programar una tarea en Ubuntu a través de la línea de comandos, podemos utilizar el fichero “crontab”, que está vinculado con el proceso demonio “cron” y cuyo servicio asociado se denomina “crond”.
- ▶ Cron es un proceso que se encarga de revisar y de ejecutar de forma periódica las tareas recogidas en el archivo “crontab”. Las tareas se introducen por líneas, cada una de las cuales recoge una tarea a ejecutar y sus correspondientes parámetros de configuración.
- ▶ Cada usuario tiene su propio archivo “crontab”, por ejemplo el correspondiente al usuario root se ubica en /etc con el nombre “crontab”. Para editarlo, podemos usar:

nano /etc/crontab

- ▶ Este archivo es muy útil para automatizar diversas tareas como la actualización del sistema o las copias de seguridad.

Planificación en comandos - Ubuntu

- Administración del archivo “crontab”.

Para crear el archivo “crontab” de un usuario concreto o editar el existente se utiliza la siguiente sintaxis:

crontab -e

La primera vez, el sistema nos dará elegir el editor de texto plano a usar. Se recomienda utilizar nano.

Los archivos “crontab” de los usuarios se guardan en el directorio /var/spool/cron/crontab con el nombre del usuario propietario.

Ambos archivos disponen de los mismos parámetros, excepto el archivo asociado al usuario root, que dispone de un campo más para especificar el usuario.

Planificación en comandos - Ubuntu

Como podemos apreciar, “crontab” no es solo el nombre de un archivo, sino también un comando que nos permite operar sobre el archivo. Por ejemplo:

- *crontab -l* nos muestra el contenido del archivo crontab del usuario activo.
- *sudo crontab -l* nos muestra el contenido del archivo crontab del usuario root.
- *crontab -u usuario archivo* sustituye el archivo crontab actual del usuario por el archivo
- *crontab -r* nos permite borrar el archivo crontab asociado al usuario.
- Para administrar el crontab de otro usuario: *crontab -u usuario*
 - # *crontab -l -u root*
 - # *crontab -e usuario*
 - # *crontab -d -u usuario*

Planificación en comandos - Ubuntu

- ▶ Cuando editamos un archivo “crontab”, nos podemos encontrar: variables, parámetros y comentarios.
- ▶ Los comentarios son las líneas que comienzan por “#”.
- ▶ Las variables que podemos encontrar son:
 - **SHELL** : se corresponde con el “Shell” bajo el cual se ejecuta el cron. En el caso de no especificar ninguno, se tomará por defecto el asociado al usuario dentro del archivo */etc/passwd*.
 - **PATH** : contiene las rutas a los directorios donde el proceso cron buscará el comando a ejecutar Este path es distinto al path global del sistema o del usuario.
 - **MAIL TO** : se especifica el e-mail de un usuario, al cual se le envía la salida del comando ejecutado, si la hay. Es un método eficaz para controlar si se ha ejecutado las tareas programadas. Si no especificamos nada, se enviará intentará enviar el correo al usuario propietario del comando.
 - **HOME** : hace referencia al directorio raíz o principal del comando cron. En caso de no especificar nada, se tomará como raíz la asociada al usuario en el archivo */etc/passwd*.

Planificación en comandos - Ubuntu

- ▶ Los parámetros se utilizan para programar las tareas y son los siguientes:
 - **m** corresponde al minuto en que se va a ejecutar el programa o script, el valor varía entre 0 a 59
 - **h** se corresponde con la hora exacta, se maneja el formato de 24 horas, los valores van de 0 a 23, siendo 0 las 12:00 de la medianoche.
 - **dom** hace referencia al día del mes, por ejemplo se puede especificar 15 si se quiere ejecutar cada día 15
 - **mon** hace referencia al mes, puede tomar valores numéricos, entre 1 y 12, o las tres primeras siglas del nombre del mes en inglés.
 - **dow** significa el día de la semana, puede ser numérico (0 a 7, donde 0 y 7 son domingo) o las 3 primeras letras del día en inglés: mon, tue, wed, thu, fri, sat, sun.
 - **user** define el usuario que va a ejecutar el comando, puede ser root, u otro usuario diferente siempre y cuando tenga permisos de ejecución del script.
 - **command** refiere al comando o a la ruta absoluta del programa o script que se va a ejecutar, por ejemplo: `sh /home/usuario/scripts/actualizar.sh`, si acaso llama a un script este debe ser ejecutable.

Planificación en comandos - Ubuntu

► Otros ejemplos de rangos especiales:

- @reboot Ejecuta una vez, al inicio
- @yearly ejecuta sólo una vez al año: 0 0 1 1 *
- @annually igual que @yearly
- @monthly ejecuta una vez al mes, el día primero: 0 0 1 * *
- @weekly Semanal el primer minuto de la primer hora de la semana. 0 0 * * 0
- @daily diario, a las 12:00A.M. 0 0 * * *
- @midnight igual que @daily
- @hourly al primer minuto de cada hora: 0 * * * *

Planificación en entorno gráfico - Linux

Sistema Ubuntu.

- ▶ Los S.O. Linux disponen de la aplicación “gnome-schedule” (tareas programadas), que realiza las mismas funciones que el proceso cron sobre el archivo crontab, pero de forma gráfica.
- ▶ Esta herramienta ha sido retirada de los repositorios de las actuales versiones de Ubuntu, por lo cual para utilizarla hay que usar una versión más antigua, por ejemplo Ubuntu 14.04.
- ▶ Para acceder a esta aplicación, abrimos el centro de software de Ubuntu y en búsqueda escribimos “gnome schedule” o “tareas programadas”.

Planificación en entorno gráfico - Ubuntu

- En el campo comando, se ha introducido una sintaxis que reinicia el sistema. Se ha introducido la ruta completa donde está ubicado el comando, pues se va a ejecutar con los privilegios de root.
- Con respecto al tipo de salida que queremos, tenemos las siguientes opciones: comportamiento predeterminado, suprimir la salida, aplicación X (se utiliza con las aplicaciones gráficas) o “aplicación X: suprimir la salida”. En nuestro caso se ha prescindido de la salida.
- En lo referente a la fecha y hora, se ha configurado el reinicio todos los domingos a las 12:00 am. Los días de la semana se numeran desde el 0 al 7, donde ambos corresponden al domingo.