

Instalación y administración de Ubuntu Server



Curso 2023/24

Ubuntu Server

Instalación

Planificación

La versión de Ubuntu que utilizaremos será **Ubuntu Server LTS**. Para realizar la instalación, haremos uso del software VMware, versión Player.

La idea es descargar el **.iso**, crear una MV en VMware e instalar nuestro Ubuntu Server.

Se recomienda, al menos, dejar 30 GB de espacio libre. En cuanto a la RAM, con 2 GB de memoria RAM debería ser suficiente.

Cabe destacar que las versiones server **no suelen traer un entorno gráfico**. Esto es así porque lo normal es acceder **remotamente** a los servidores a través de la consola y realizar la gestión del servidor desde ahí. Aún así, veremos cómo instalar el entorno GNOME.

Instalación I

Lo primero que debemos hacer es seleccionar la opción que viene por defecto para comenzar con la instalación. A continuación, debemos elegir el idioma.

```
[ Deutsch  
[ English  
[ English (UK)  
[ Español
```

Elegimos que, en efecto, es Ubuntu Server lo que queremos instalar.

```
Choose type of install
```

```
Choose the base for the installation.
```

```
(X) Ubuntu Server
```

```
The default install contains a curated
```

Instalación II

Si estamos conectados a Internet, al menos debería poder detectar una interfaz. En principio se tomará la dirección IP por DHCP. Como ya hemos nombrado en muchas ocasiones, en realidad un servidor **debe tener** una IP fija.

Network connections

Configure at least one interface this server can access for updates.

NAME	TYPE	NOTES
[ens33	eth	-
DHCPv4	192.168.253.130/24	
00:0c:29:3b:73:9c / Intel Corporation / 82545EM		

Instalación III

Durante el siguiente paso se ofrece la posibilidad de configurar un proxy mediante el formulario que aparece. ¿Por qué querríamos poder configurar un proxy aquí?

```
Configure proxy
If this system requires a proxy to connect
Proxy address: 
                If you need to use a HTTP
                leave this blank.
```

El **mirror** lo dejamos tal y como está.

```
Configure Ubuntu archive mirror
If you use an alternative mirror for Ubuntu, enter its details here.
Mirror address: 
                You may provide an archive mirror that will be used
```

Instalación IV

Le vamos a decir que queremos usar **todo el disco** para realizar la instalación. Y no, no vamos a encriptar nada; ni disco ni ninguna partición.

```
Guided storage configuration
Configure a guided storage layout, or create a custom one:
(X) Use an entire disk
    [ /dev/sda local disk 50.000G ▼ ]
[X] Set up this disk as an LVM group
    [ ] Encrypt the LVM group with LUKS
```

En esta pantalla tenemos que desplazarnos hasta abajo y pulsar **Done**.

```
[ Done ]
[ Back ]
```

Instalación V

En la configuración de almacenamiento le diremos que lo queremos todo tal y como viene por defecto. Nos dirá que dicha acción **destruirá** todo el contenido. Pues OK.

```
Storage configuration

FILE SYSTEM SUMMARY

MOUNT POINT      SIZE      TYPE      DEVICE TYPE
[ /              23.996G   new ext4   new LVM Logical Volume ► ]
[ /boot          2.000G   new ext4   new partition of local disk ► ]

Confirm destructive action

Selecting Continue below will begin the installation process and
result in the loss of data on the disks selected to be formatted.

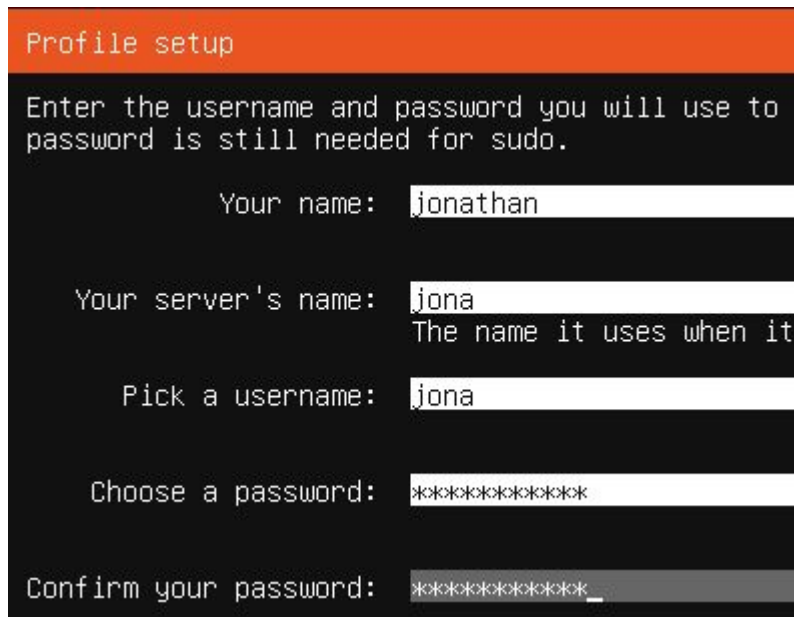
You will not be able to return to this or a previous screen once the
installation has started.

Are you sure you want to continue?

[ No ]
[ Continue ]
```


Instalación VI

A continuación, debemos crear nuestro login. Aquí tenéis el mío.



The image shows a terminal window titled "Profile setup" with an orange header. The text inside the terminal is as follows:

```
Profile setup
Enter the username and password you will use to
password is still needed for sudo.

Your name: jonathan

Your server's name: jona
The name it uses when it

Pick a username: jona

Choose a password: 12345678901234567890

Confirm your password: 12345678901234567890_
```

Instalación VII

Dejamos las opción de cifrado SSH tal y como vienen por defecto.

```
SSH Setup
You can choose to install the OpenSSH server package
[ ] Install OpenSSH server
```

Y le decimos que **no queremos** añadir ninguna característica más a nuestro servidor.

```
Featured Server Snaps
These are popular snaps in server environments
publisher and versions available.
[ ] microk8s          canonical✓
[ ] nextcloud         nextcloud✓
[ ] wekan             xet7
[ ] kata-containers   katacontainers✓
```

Instalación VIII

Y listo, **comenzará** la instalación.

```
configuring installed system
  running 'mount --bind /cdrom /target/cdrom'
  running 'curtin curthooks'
    curtin command curthooks
      configuring apt configuring apt
      installing missing packages
      configuring iscsi service
      configuring raid (mdadm) service
      installing kernel |
```

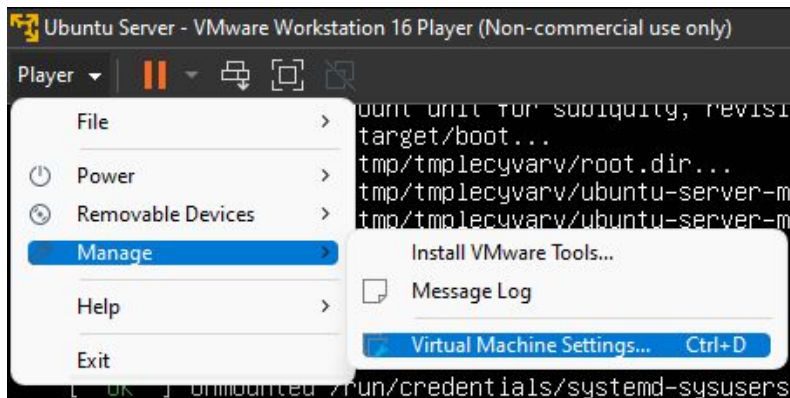
En algún punto de la misma nos dirá que ya podemos reiniciar el sistema.

```
[ View full log ]
[ Reboot Now    ]
```

Instalación IX

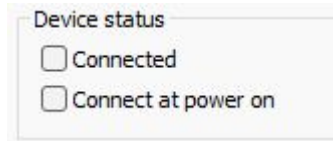
Cuando se reinicie el sistema debemos quitar la imagen .iso. Recordemos que esto es como si dejamos el CD metido en la CPU. Y, de hecho, nos dará un aviso.

```
Please remove the installation medium, then press ENTER:  
  Unmounting /cdrom...  
[FAILED] Failed unmounting /cdrom.
```



Instalación X

Seleccionamos CD-ROM y debemos quitar los check que indican que nuestro CD está conectado.



Después de esto, cuando arranque nuevamente el servidor, debemos esperar unos minutos. A continuación, pulsamos ENTER y debería aparecer el login. Aquí tenemos que introducir el usuario y password que configuramos anteriormente.

```
ck datasource  
[ 22.587694] hub 2-2:1.0: hub_ext_port_status failed (err = -110)  
jona login:
```

Instalación XI

Por último, cuando aparezca el **prompt**, ya estaremos listos para poder comenzar a gestionar y configurar nuestro Ubuntu Server.

```
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
jona@jona:~$
```

Instalación del entorno gráfico

Los escritorios o entornos gráficos son **un conjunto de programas** que ofrecen ventanas, iconos y otras aplicaciones que facilitan el uso del ordenador. Los entornos más populares en GNU/Linux son GNOME, KDE, Unity y Cinnamon.

Para hacer la experiencia de administración un poco más agradable, **se recomienda** instalar el entorno gráfico en Ubuntu Server (aunque **no es**, ni mucho menos, necesario).

Tarea: instala un entorno gráfico en Ubuntu Server y anota los pasos que has seguido.

Actualizar el sistema desde el repositorio

A la hora de llevar a cabo la instalación de un SO, siempre resulta conveniente comenzar con una **instalación mínima** y, poco a poco, ir añadiendo los paquetes necesarios.

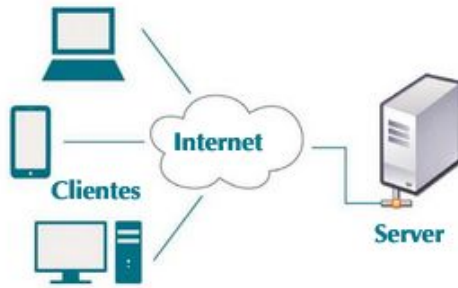
Ubuntu dispone de **varios servidores de repositorios**, que son contenedores de paquetes que se encuentran accesibles en un servidor para que los sistemas que accedan a él puedan descargar los **recursos necesarios**.

Apt es una **herramienta** disponible en distribuciones como Debian y Ubuntu, entre otras, que permite **gestionar la descarga y actualización de software**.

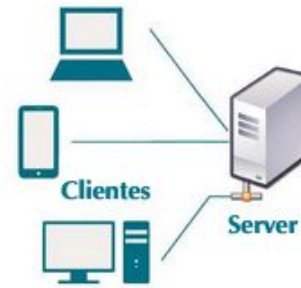
Actualizar el sistema desde el repositorio

Estos repositorios, en ocasiones, se encuentran configurados en la **red local** de manera que, así, los diferentes equipos clientes **no acceden a la red externa** para actualizarse, sino que **acceden a dicho servidor** de repositorio situado en la red local.

Esto **evita que haya mucho tráfico** en la red hacia Internet y optimiza los tiempos de actualización.



Servidor de repo en Internet



Servidor de repo en red local

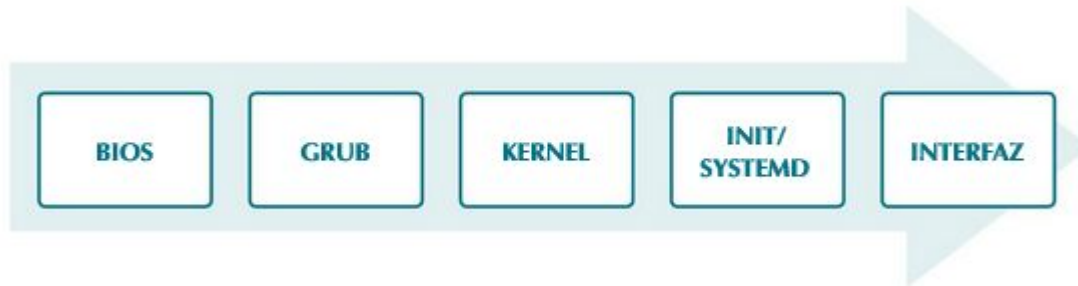
Arranque del SO y servicios

Durante el proceso de arranque de un sistema, Linux lleva a cabo varias etapas:

- ❑ En primer lugar, toma el **control la BIOS**, que realiza la comprobación de los dispositivos disponibles y carga en memoria el **cargador de arranque**, al que termina cediendo el control.
- ❑ El gestor de arranque GRUB (GRand Unifier Bootloader) muestra un **menú de inicio**, ofreciendo qué partición queremos como activa y cargando el kernel en la memoria. El fichero de configuración de GRUB se encuentra en `/boot/grub/grub.cfg`.

Arranque del SO y servicios

- ❑ El kernel **transfiere el control** del proceso de arranque al programa `/sbin/init` (padre de todos los procesos), ahora conocido como **systemd** (system daemon).
- ❑ Este **supervisa y gestiona las tareas y servicios de inicio** cuando el sistema arranca y los detiene cuando el sistema se apaga.
- ❑ Por último, muestra al usuario la **pantalla de inicio**.



Proceso de arranque

Administración de archivos y directorios

La tarea de administrar archivos y directorios es una de las **más delicadas** durante el mantenimiento del sistema, ya que se han de modificar ficheros de configuración, administrar usuarios y grupos, gestionar recursos, etc.

Cabe destacar que, en los sistemas Linux, el **elemento básico es el archivo**. De hecho, los directorios son archivos que contienen archivos y los dispositivos físicos se tratan como archivos, eso sí, especiales. Estas son algunas de las operaciones básicas.

	Directorio	Fichero
Crear	mkdir directorio	vi fichero.ext
Consultar	ls directorio	more fichero.ext
Borrar	rm -r directorio	rm fichero.ext
Renombrar	mv origen destino	
Copiar	cp origen destino	

Administración de archivos y directorios

A grandes rasgos, es posible resumir las características especiales de los archivos y directorios.

- ❑ No pueden contener **caracteres especiales** en su nombre. Estos son: ! # & ' \" \$ > < @ { } * ? Tab Spacio + -
- ❑ Un archivo suele tener una extensión compuesta por **tres caracteres** que ofrecen información acerca de su contenido.
- ❑ Linux es **case sensitive**, por lo que distingue minúsculas y mayúsculas. El fichero `fichero.txt` no es el mismo que `FICHERO.txt`.

Administración de archivos y directorios

Otro concepto muy importante es el de ruta, que es el camino que ha de seguirse hasta alcanzar un objetivo.

- ❑ Relativas: depende de la posición actual. Por ejemplo, `ls ../dir2/dir1/carlos`
- ❑ Absolutas: camino que depende de la raíz. Por ejemplo, `/home/carlos/dir1`.

Los directorios especiales en Linux son los siguientes:

- ❑ `.` es el directorio actual.
- ❑ `..` hace referencia al directorio padre.
- ❑ `/` es el directorio raíz de la estructura.

Administración de archivos y directorios

Los permisos y sus efectos sobre archivos y directorios se adjunta en la siguiente tabla. Es **importante** aprenderse de memoria los efectos de cada permiso.

Permiso	Directorio	Archivo
r	Permite conocer qué archivos o directorios contiene.	Permite visualizar.
w	Permite crear archivos o directorios en él.	Permite modificar o escribir en él.
x	Permite situarse en él.	Permite ejecutar un archivo.

Si en lugar del permiso `r`, `w` o `x`, el archivo o directorio representa un `-`, indica que dicho permiso no existe.

Administración de archivos y directorios

Como ya sabemos, esta asignación de permisos se realiza con el comando `chmod`. Con este comando, existen dos nomenclaturas de permisos: **letras** y **números**.

usuario	grupo	otros
rwX	rwX	rwX
u	g	o
a		

Nomenclatura de letras al asignar permisos

Administración de archivos y directorios

Octal	Binario	Lectura (r)	Escritura (w)	Ejecución (x)
0	000	✗	✗	✗
1	001	✗	✗	✓
2	010	✗	✓	✗
3	011	✗	✓	✓
4	100	✓	✗	✗
5	101	✓	✗	✓
6	110	✓	✓	✗
7	111	✓	✓	✓

Nomenclatura de números al asignar permisos

Administración de archivos y directorios

Otro de los aspectos realmente útiles de Linux es el **redireccionamiento**, el cual nos permite **redireccionar entrada y salida de información** desde o hacia un fichero o destino determinado.

Cada proceso tiene asociado 3 ficheros para entrada salida (E/S):

- ❑ 0: entrada estándar (teclado).
- ❑ 1: salida estándar (pantalla).
- ❑ 2: salida de errores (pantalla).

Administración de archivos y directorios

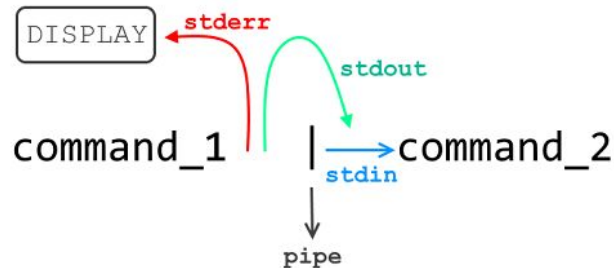
Para **utilizar** o **modificar** tanto la salida como la entrada estándar, así como la salida de errores, se pueden utilizar los siguientes símbolos.

Carácter	Función
(pipeline)	La salida estándar de un comando se convierte en la entrada estándar del siguiente comando.
<	Redirecciona la entrada estándar (teclado) sustituyéndola por el fichero que se indique.
>	Redirecciona la salida estándar (pantalla) de un proceso al fichero indicado, eliminando la información que el fichero contenía y creándolo si el fichero no existía.
>>	Redirecciona la salida estándar (pantalla) de un proceso al fichero indicado, no eliminando el contenido del fichero, sino añadiendo la información al final.
2> o 2>>	Redirecciona la salida de errores al archivo indicado.
&> o &>>	Redirecciona la salida de errores y la estándar al fichero indicado.

Administración de archivos y directorios

Vamos por partes. Empezando por las tuberías, es común que estas vayan asociadas al comando `grep`, el cual se utiliza para **encontrar cadenas** de texto en la terminal.

Con una tubería puedes hacer que la salida estándar o `stdout` de un comando **pueda pasar directamente** a la entrada estándar de otro. En vez de usar parámetros introducidos por el teclado, con una pipe se le entrega la información generada por el comando previo mediante esta canalización representada con el símbolo `|`.



Administración de archivos y directorios

Lo mejor es verlo con ejemplos. Imagina que quieres listar el contenido de un directorio, pero que solo te interesa ver los nombres que coincidan con la palabra `doc`. Entonces, podría usar una pipe para **canalizar la salida** de `ls` y llevarla a la entrada del filtro `grep` para decirle que solo muestre los que **coincidan con ese patrón**.

```
ls -l | grep doc
```

Así, en vez de mostrarte todos los nombres, solo te muestra los que **realmente te interesan**.

Administración de archivos y directorios

Igual puedes hacer con el contenido de un archivo. Imagina que solo quieres ver la información de los procesos con nombre **firefox** y no todos:

```
ps aux | grep firefox
```

En vez de mostrar toda la salida del programa `ps` en pantalla (stdout), lo que hace es **canalizarlo** hacia la entrada del filtro `grep` y solo muestra en la salida lo que se corresponde con el patrón **firefox** en este caso.

Tarea: ¿para qué sirve el comando `ps aux`?

Administración de archivos y directorios

Si quieres, puedes usar **varias tuberías** para llevar la salida de un comando a la entrada de otro segundo comando, y la salida de ese segundo hacia la entrada de un tercero y así sucesivamente.

```
cat libro | grep love | more
```

El comando `more` te permite mostrar el resultado de la ejecución de un comando en la terminal de a una página a la vez. Esto es especialmente útil cuando se ejecuta un comando que causa un **gran desplazamiento** (como el comando `cat` o `ls`).

Administración de archivos y directorios

Por otro lado tenemos las redirecciones. Ya hemos visto `>` y `>>`, por lo que vamos a pasarlas por alto. La redirección `<` puede que no sea especialmente útil, pero puedes ver su comportamiento creando un fichero que contenga *"Esto es una prueba"* y ejecutando el comando:

```
cat < nombrefichero.txt
```

Observa cómo el comando **cat** toma como entrada el contenido del fichero `nombrefichero.txt` y lo muestra por la **salida estándar** (pantalla).

Administración de archivos y directorios

Observando nuevamente la tabla, las últimas cuatro opciones son sencillas de probar.

Tarea: prueba a redireccionar la salida de error. Para hacer esto, debes ejecutar un comando que tú sepas que dará error (puedes hacer la prueba usando en primer lugar la salida estándar y luego redireccionarla hacia un fichero). Por lo tanto, prueba los siguientes comandos: `2>`, `2>>`, `&>` y `&>>`.

Conocimiento del hardware

Una vez identificado el hardware del equipo, el siguiente paso es manejar herramientas que ayuden a **controlar el rendimiento** de ese hardware.

El rendimiento de un equipo está **íntimamente ligado al funcionamiento del hardware** (CPU, RAM, disco duro, etc.). Por ello, es **altamente recomendable** manejar herramientas que controlen, supervisen o monitoricen los recursos hardware con aras de detectar problemas que puedan afectar al rendimiento global del sistema.

En el mercado, existen **muchas herramientas** con las que se podrá monitorizar un sistema GNU/Linux. Algunas de ellas son **gráficas** y presentan diagramas, líneas, sectores, etc. Y otras, en las que nos centraremos, son por **línea de comandos**.

Conocimiento del hardware

Monitorear la CPU es labor es muy importante porque si llega a un porcentaje muy elevado de utilización, las aplicaciones se detendrán o funcionarán lentamente.

Uno de los comandos más usados es `top`, el cual mostrará de manera dinámica y en tiempo real un resumen de la utilización del sistema.

```
top - 12:29:11 up 2:30, 1 user, load average: 0.18, 0.04, 0.01
Tasks: 299 total, 1 running, 298 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.3 us, 0.0 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 1941.6 total, 86.8 free, 928.4 used, 926.5 buff/cache
MiB Swap: 2048.0 total, 2047.0 free, 1.0 used, 803.3 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2134	jona	20	0	560544	53876	41588	S	0.3	2.7	0:06.04	gnome-terminal-
3691	root	20	0	0	0	0	I	0.3	0.0	0:00.02	kworker/u256:2-events_unbound
3692	root	20	0	0	0	0	I	0.3	0.0	0:00.22	kworker/1:1-events
1	root	20	0	167084	11964	8076	S	0.0	0.6	0:05.42	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.03	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp

Conocimiento del hardware

En azul, se aprecia que, en ese justo momento, el 0.3% de la CPU está siendo utilizada por procesos del usuario y el 0.0% por procesos del sistema, por lo que está inactiva al 99.7% (es posible que con un decimal no sea capaz de calcular la utilización de los procesos del sistema).

Además, se observan tres procesos que están consumiendo el 0.3% (la gestión de procesos es muy compleja y probablemente estén compartiendo la utilización de la CPU).

Tarea: investiga y averigua qué indica cada una de las columnas del comando `top`.

Conocimiento del hardware

Otro comando muy útil (y especialmente más agradable) es `htop`. Podría decirse que este comando tan parecido a `top` permite **matar procesos** fácilmente y visualizar la información (tanto CPU como RAM) de manera mucho más **gráfica**.

0[6.7%]										Tasks: 118, 259 thr; 1 running	
1[7.3%]										Load average: 0.07 0.02 0.02	
Mem[950M/1.90G]										Uptime: 02:52:16	
Swp[12.8M/2.00G]											
PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
1692	jona	20	0	4022M	243M	105M	S	5.4	12.5	1:23.06	/usr/bin/gnome-shell
2134	jona	20	0	548M	54308	41876	S	2.7	2.7	0:15.65	/usr/libexec/gnome-terminal-server
5958	jona	20	0	8696	4668	3496	R	2.7	0.2	0:00.19	htop
5923	jona	20	0	2726M	59408	44492	S	2.0	3.0	0:00.82	gjs /usr/share/gnome-shell/extensions/ding@rastersoft
755	systemd-o	20	0	14824	5260	4464	S	0.7	0.3	0:22.28	/lib/systemd/systemd-oomd
1716	jona	20	0	305M	7168	6108	S	0.7	0.4	0:00.05	/usr/libexec/gvfs-afc-volume-monitor

Conocimiento del hardware

La *memoria física* son los módulos de memoria RAM que **están conectados en la placa** base del equipo, es **más rápida** y su velocidad de acceso es la misma **independientemente de la posición** de memoria a la que se esté accediendo.

Otro concepto es el de *memoria virtual*, que **no existe realmente como componente**, sino que utiliza otro medio de almacenamiento (generalmente el disco duro).

Una vez que la memoria física se llena, el SO deberá **intercambiar procesos** desde la memoria física a la memoria virtual con el consiguiente descenso de rendimiento del equipo, que podría llegar a colapsar.

Conocimiento del hardware

El estado de estas dos memorias se puede consultar con el comando `free`. También podemos usar `free -h` para el tema de las unidades (GB y MB).

	total	used	free	shared	buff/cache	available
Mem:	1988240	950652	88840	38360	948748	822544
Swap:	2097148	1036	2096112			

La primera línea se corresponde con los valores asociados a la memoria física o memoria RAM, y la segunda línea se corresponde con valores asociados a la memoria virtual (también llamada swap).

Observamos que, de los 1.9 GB que hay de RAM, **tan sólo** tenemos 88 MB libres. ¿Cómo puede ser esto posible? ¿No es **demasiado uso** para tener únicamente la terminal?

Conocimiento del hardware

La explicación es que el SO de Linux tiende a usar el **máximo** posible de RAM debido a:

- ❑ **Acelerar lecturas en disco:** Linux usa parte de la memoria RAM que no se está usando para almacenar la información de los archivos y programas que se usan más frecuentemente para que, cuando se vuelva a acceder a ellos, se haga mucho más rápido.
- ❑ **Acelerar la asignación de memoria RAM a los procesos:** cuando se lanza un programa, este se convierte en un proceso, por lo que el kernel reserva espacio para él. Lo que ocurre es que esta cantidad es superior a la que se va a usar. Así, en el momento en el que el proceso necesite más RAM, el kernel se la puede asignar de inmediato puesto que ya la tenía reservada cuando se lanzó el programa.

Conocimiento del hardware

Después de la memoria total, usada y libre, nos encontramos con otras **tres columnas**:

- ❑ **shared**: muestra la cantidad de memoria RAM que está siendo compartida y usada por más de un proceso. De esta forma, los procesos se pueden comunicar entre ellos y evita copiar datos redundantes en la memoria.
- ❑ **buff/cache**: es la cantidad de memoria RAM que Linux se reserva para los dos objetivos nombrados anteriormente.
- ❑ **available**: es una estimación de la memoria RAM disponible para ejecutar nuevos procesos sin necesidad de recurrir a la memoria virtual.

Conocimiento del hardware

Por último tenemos la monitorización del disco. Esta es **realmente importante** por las implicaciones que puede tener.

Por ejemplo, si la partición `/home` está ocupada al 100%, los usuarios **no podrán almacenar ningún fichero** nuevo en sus carpetas de trabajo. Pero si la partición que está llena es la raíz `/`, los problemas pueden ser aún mayores, como por ejemplo que **no arranque el entorno gráfico** o caídos y **fallos críticos** en el sistema.

```
jona@jona:~$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
tmpfs	195M	1.7M	193M	1%	/run
/dev/mapper/ubuntu--vg-ubuntu--lv	24G	9.8G	13G	44%	/
tmpfs	971M	12K	971M	1%	/dev/shm
tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
/dev/sda2	2.0G	253M	1.6G	14%	/boot
tmpfs	195M	96K	195M	1%	/run/user/1000

Instalación y administración de Ubuntu Server



Curso 2023/24

Profesor
Jonathan Carrero