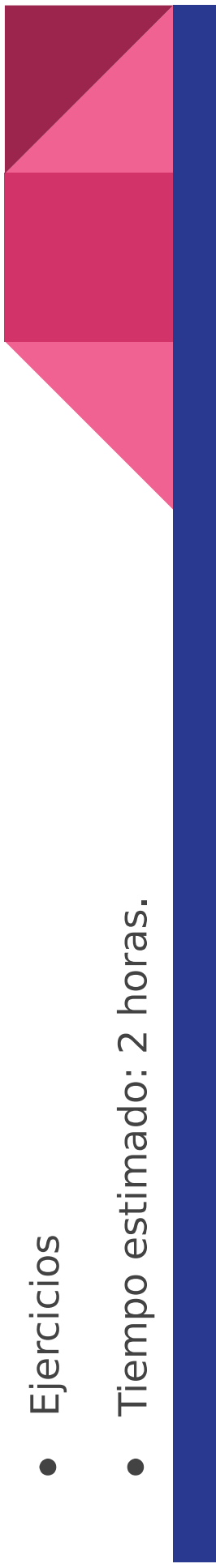




Monitorización de rendimiento y logs

Monitorización de rendimiento y logs. Objetivos

- Monitorización de rendimiento y logs (monitorización de E/S, memoria, c parámetros de red, revisión automática de logs).
- Usar herramientas predefinidas para ello.
- Aprender a hacer scripts con bash, sed, grep, cut, tr, join awk y similares el análisis de logs.
- Permisos de ejecución, uso del shebang.
- Ejercicios
- Tiempo estimado: 2 horas.



Monitorización de rendimiento y logs. Man.

- Consulta el manual de un comando concreto
 - `man comando`
- Manual en español:
 - `man -L es awk`
- Buscar en las descripciones un comando concreto:
 - `man -k regexp`
 - Por ejemplo: `man -k hardware` (comandos relacionados con el hardware)
 - `apropos "list directory"` (alternativa)



Monitorización de rendimiento y logs. Grep.

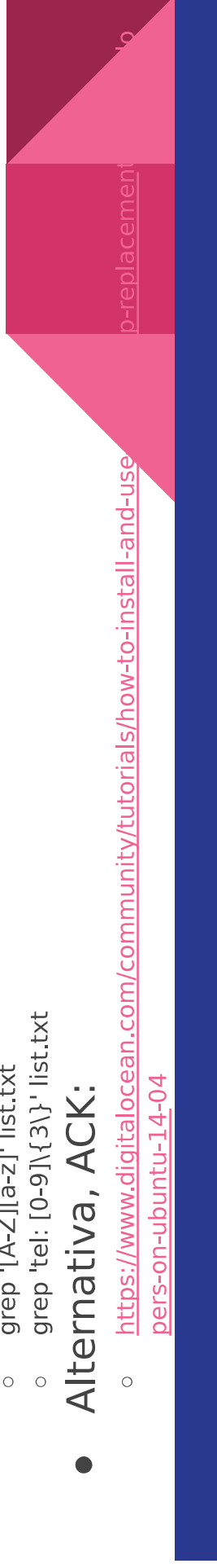
- Busca cadenas, y muestra las líneas donde se encuentra:
 - `grep [options] regex [files]`
 - <http://regexr.com/>
 - <https://linuxzone.es/2012/12/03/tip-colorea-la-salida-del-comando-grep/>
- Líneas que contienen la cadena "inode lis " en el fichero list.txt:
 - `grep "inode lis " list.txt`
- Buscar "inode list" en todos los ficheros del directorio actual:
 - `grep "inode list" *`
- Buscar "inode list" recursivamente a partir del directorio actual:
 - `grep -R "inode list"`

● Usando expresiones regulares:

- `grep 's\{3,4\}' list.txt`
- `grep '[A-Z][a-z]' list.txt`
- `grep 'tel: [0-9]\{3\}' list.txt`

● Alternativa, ACK:

- <https://www.digitalocean.com/community/tutorials/how-to-install-and-use-pers-on-ubuntu-14-04>

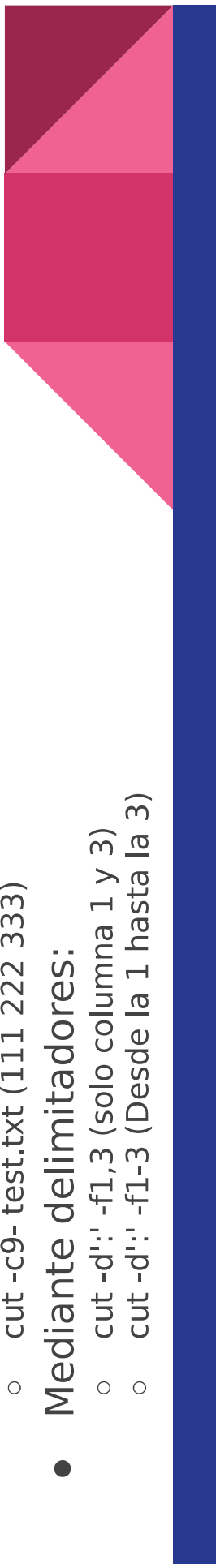


Monitorización de rendimiento y logs. Sed.

- Sustituye texto en líneas.
 - `sed [options] 'command1' [files]`
`sed [options] -e 'command1' [-e command2 ...] [files]`
`sed [options] -f script [files]`
- Eliminar líneas que tienen "O" al principio:
 - `sed '/^O/d' list.txt`
- Sustituir letras individuales C,R,O por c,r,o:
 - `sed 'y/CRO/cro/' list.txt`
- Eliminar líneas vacías:
 - `sed '/^$/d' list.txt`
- Sustituir cadena Oop por Wee, solo la primera ocurrencia en línea:
 - `sed 's/Oop/Wee/' lsst.txt`
- Eliminar cadena ss, todas las ocurrencias de la línea:
 - `sed 's/ss//g' list.txt`

Monitorización de rendimiento y logs. Cut.

- Muestra o elimina secciones (campos/columnas concretas), mediante índices o usando separadores.
 - `cut OPTION... [FILE]`
 - `cat test.txt`
`w1:w11:w111`
`w2:w22:w222`
`w3:w33:w333`
- A nivel de caracteres:
 - `cut -c2 test.txt (1 2 3)`
 - `cut -c1-2 test.txt (w1 w2 w3)`
 - `cut -c9- test.txt (111 222 333)`
- Mediante delimitadores:
 - `cut -d':' -f1,3 (solo columna 1 y 3)`
 - `cut -d':' -f1-3 (Desde la 1 hasta la 3)`



Monitorización de rendimiento y logs. Tr.

- Cambia determinada información de un archivo por otra:
 - `tr [OPTION]... SET1 [SET2]`
- Cambiar letras concretas (respectivamente en los dos set):
 - `echo "ejemplo" | tr 'eo' 'EO'`
- Sustituir múltiples caracteres seguidos por uno solo:
 - `echo "ejemplo--1----2---3" | tr -s "-"`
- Cambiar por ? los caracteres no encontrados:
 - `echo '#123#' | tr -c '[a-z][A-Z][0-9]\n' '?'`



Monitorización de rendimiento y logs. Sort.

- Ordenar líneas de los archivos de entrada a partir de criterios de ordenación.
Los espacios en blanco son tomados por defecto como separadores de campo.
- `sort [opciones] [archivo]`
 - `-b` Ignora espacios en blanco precedentes.
 - `-d` Ordena ignorando todos los caracteres salvo caracteres letras, números y espacios.
 - `-f` considera iguales las mayúsculas y minúsculas.
 - `-n` ordena por valor numérico.
 - `-r` invertirá el orden.
 - `-k n1,[n2]` Especifica un campo como clave de ordenación
 - `-o` fichero Salida a un fichero
- <https://www.computerhope.com/unix/usort.htm>



Monitorización de rendimiento y logs. Join.

- Combina ficheros por columnas/campos.
 - join fichero1 fichero2
- Los ficheros deben estar ordenados por campo que se va a mezclar.
- Especificar separador (por defecto espacio).
 - join -t":" profesores.txt grupos.txt
- Indicar que campos mezclar (por defecto siempre es el primero):
 - join -t":" -1 2 -2 2 profesores.txt grupos.txt

```
cat foodtypes.txt
1 Protein
2 Carbohydrate
3 Fat
```

```
cat foods.txt
1 Cheese
2 Potato
3 Butter
```

```
join foodtypes.txt foods.txt
1 Protein Cheese
2 Carbohydrate Potato
3 Fat Butter
```



Monitorización de rendimiento y logs. Awk.

- Lenguaje de programación para el procesado de textos.
 - awk 'programa' archivos de entrada
- Ejemplo sencillo:
 - awk '\$3 > 0 {print \$1, \$2 * \$3}' ArchivoAviones.dat

Jose	4.00	0	
Pedro	7.00	1	Pedro 7
Rosa	23.30	5	→ Rosa 116.5
Maria	28.10	6	Maria 168.6
Juan	29.32	0	

- <https://aprendiendoausarlinux.wordpress.com/2012/03/09/un-tutorial-de-awk/>
- <https://www.ibm.com/developerworks/library/l-awk1/index.html>
- <http://www.thegeekstuff.com/2010/01/8-powerful-awk-built-in-variables-fs-ofs-rs-ors-nr-nf-filenamena-tnr/>
- <http://www.thegeekstuff.com/2010/01/awk-introduction-tutorial-7-awk-print-examples>



Monitorización de rendimiento y logs. Monitorización.

- Existen diversos comandos para monitorización o información del sistema

Un resumen de ellos:

- Top (se verá en Gestión de procesos)
- Netstat
- Lshw
- history (ctrl + R)
- less



Monitorización de rendimiento y logs. Bash Shell Scripting.

- Es posible crear ejecutables interpretados usando los comandos anteriores
- Estos son programados con un lenguaje propio
 - Se puede usar sencillamente un editor de texto
- Se guardan en ficheros habitualmente con la extensión .sh
- Shebang más habitual: `#!/bin/bash` (se debe poner al principio del script)
- Debemos otorgarle permisos de ejecución:
 - `sudo chmod +x fichero.sh`
 - <https://www.usemoslinux.net/index.php/tutorial-chmod-permisos>



Monitorización de rendimiento y logs. Bash Shell Scripting.

- Variables:
 - `var1=ovtoaster`
 - `echo $var1`
 - `unset var1`
- Arrays:
 - `ciudad[0]="Madrid"`
`ciudad[1]="Tokyo"`
`ciudad[2]="Berlin"`
`ciudad[4]="Lima"`
`echo ${ciudad[4]}`
 - Alternativa: `ciudad=(Madrid Tokyo Berlin Lima)`



Monitorización de rendimiento y logs. Bash Shell Scripting.

- Detalle sobre las comillas:
 - `echo "La variable es: $variable"` (muestra el valor)
 - `echo 'La variable es: $variable'` (no muestra el valor)
- Si queremos mostrar el valor con más caracteres seguidos:
 - `touch "${USER_NAME}_file"`
- Si queremos que la variable sea el resultado de un comando:
 - `INPUT_STRING=`head -1 /etc/hosts``



Monitorización de rendimiento y logs. Bash Shell Scripting.

- Argumentos:
 - \$1 ... \$9 : Argumentos pasados al programa. Si usamos las llaves "{}" para nombrar las variables el límite se puede extender \${11}
 - \$0 : Nombre del programa que se está ejecutando.
 - \$# : Número de argumentos que se han pasado al programa.
 - \$*: Todos los argumentos pasados al programa.
 - \$@ : Todos los argumentos pasados al programa pero teniendo en cuenta su separación.



Monitorización de rendimiento y logs. Bash Shell Scripting.

- Primer script:
 - `#!/bin/sh`
 - `# Comentario del script`
 - `echo Hello World # Otro comentario del script`
- Guardar en fichero `first.sh`.
- Dar permisos de ejecución:
 - `chmod +x first.sh`



Monitorización de rendimiento y logs. Bash Shell Scripting.

If else:

Comparisons:

```
if [ ... ]; then
```

```
# if-code
```

```
else
```

```
# else-code
```

```
fi
```

```
-eq equal to
```

```
-ne not equal to
```

```
-lt less than
```

```
-le less than or equal to
```

```
-gt greater than
```

```
-ge greater than or equal to
```

File Operations:

```
-s file exists/not empty
```

```
-f file exists/not a directory
```

```
-d directory exists
```

```
-x file is executable
```

```
-w file is writable
```

```
-r file is readable
```

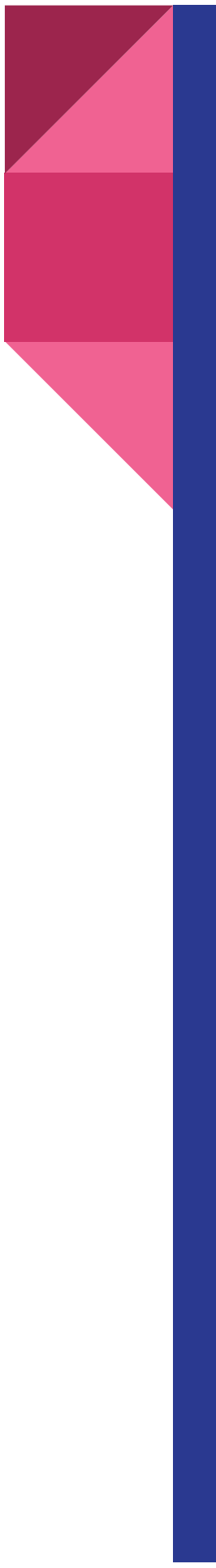
<https://www.shellscript.sh/quickref.html>

https://es.wikibooks.org/wiki/EI_Manual_de_BASH_Scripting_B%C3%A1sico_para_Principiantes



Monitorización de rendimiento y logs. Ejercicios.

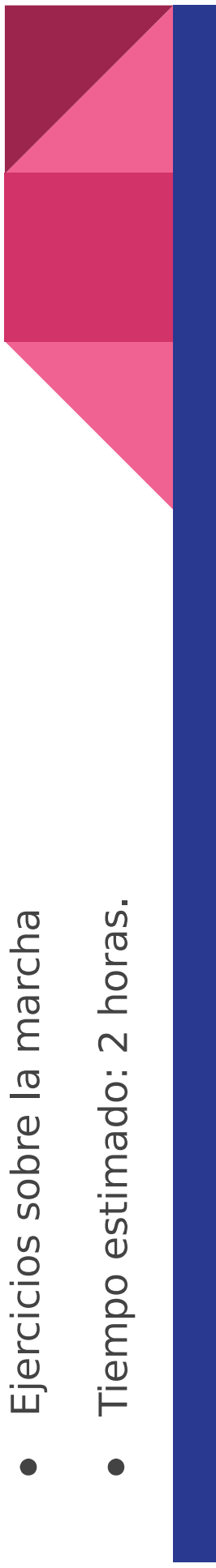
- Ejercicios Shell Scripts básicos
- Ejercicios de herramientas
 - <https://tinyurl.com/cpd-linux-ejer>



Gestión de procesos

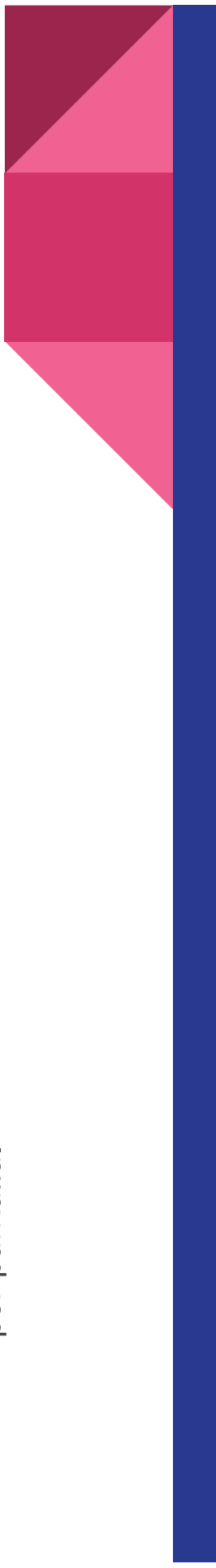
Gestión de procesos (Troubleshooting de procesos). Objetivos.

- Análisis de logs y pautas a seguir ante un mal funcionamiento del ordenador.
- ¿Cómo saber lo que está pasando y por qué puede estar producido?
- Qué estructura siguen los procesos.
- Tabla de procesos (top, htop, ps, pstree), dmesg (/var/log/syslog).
- Ejercicios sobre la marcha
- Tiempo estimado: 2 horas.



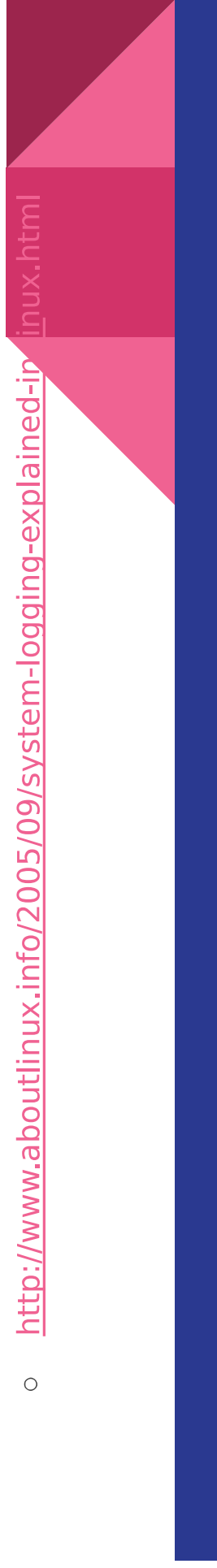
Gestión de procesos. Logs.

- <http://www.estrellateyarde.org/logs-en-linux>
- Procesos de logs:
 - syslogd: gestiona los logs del sistema. Distribuye los mensajes a archivos, tuberías, destinos remotos, terminales o usuarios, usando las indicaciones especificadas en su archivo de configuración `/etc/syslog.conf`, donde se indica que se loguea y a dónde se envían estos logs.
 - klogd: se encarga de los logs del kernel. Lo normal es que klogd envíe sus mensajes a syslogd pero no siempre es así, sobre todo en los eventos de alta prioridad, que salen directamente por pantalla.



Gestión de procesos. Logs.

- Nivel de importancia:
 - Menos a más prioritario: debug, info, notice, warning, warn, err, error, crit, alert, emerg y panic
- Tipos de mensajes:
 - auth, authpriv, cron, daemon, kern, lpr, mail, mark, news, security, syslog, user, uucp, local0-local7
- Explicación:
 - <http://www.aboutlinux.info/2005/09/system-logging-explained-in-linux.html>



Gestión de procesos. Logs.

<https://blog.desdelinux.net/donde-se-encuentran-los-archivos-de-registro-logs-en-linux/>
<https://www.eurovps.com/blog/important-linux-log-files-you-must-be-monitoring>

- **/var/log/syslog**: registro de mensajes generales del sistema
- **/var/log/auth.log**: log de autenticación
- **/var/log/kern.log**: registro del kernel
- **/var/log/dmesg**: almacena la información que genera el kernel durante el arranque del sistema (comando **dmesg**)
- **/var/log/cron.log**: registro de crond
- **/var/log/maillog**: registro del servidor de mails
- **/var/log/qmail/**: registro de Qmail
- **/var/log/httpd/**: registro de errores y accesos a Apache
- **/var/log/lighttpd**: registro de errores y accesos a Lighttpd
- **/var/log/boot.log**: registro de inicio del sistema
- **/var/log/mysqld.log**: registro de la base de datos MySQL
- **/var/log/secure**: log de autenticación
- **/var/log/utmp** or **/var/log/wtmp**: registro de logins



Gestión de procesos. Logs.

- Compresión diaria:
 - Se va generando una copia comprimida, para no llenar demasiado los logs.
 - En cualquier caso, no es recomendable abrirlo con editores visuales.
- Formas de ver un log:
 - `tail -f /var/log/messages`
 - `less /var/log/messages`
- Script para colores en tail:
 - <https://github.com/ryanoasis/public-bash-scripts/blob/master/color-log.sh>



Gestión de procesos. Logs.

- Herramientas visuales:
 - KSystemLog (paquete ksystemlog): monitor de logs de KDE.
 - GNOME-System-Log (paquete gnome-utils): monitor de logs de GNOME.
 - Xlogmaster (paquete xlogmaster): monitor de logs del GNU.
 - Xwatch (paquete xwatch): monitor de logs para las X.



Gestión de procesos. Logs.

```
auth.log - Systemprotokollbetrachter
Datei Bearbeiten Ansicht Filters Hilfe
Xorg.O.log
auth.log
daemon.log
debug
messages
syslog

Mar 2 10:02:01 Obelix CRON[3366]: pam_unix(cron:session): session opened for user root by (uid=0)
Mar 2 10:02:01 Obelix CRON[3366]: pam_unix(cron:session): session closed for user root
Mar 2 10:04:24 Obelix sudo: jhs : TTY=pts/6 ; PWD=/home/jhs/develop/anjuta-trunk/manuals/anjuta-manual ;
Mar 2 10:05:58 Obelix sudo: jhs : TTY=pts/6 ; PWD=/home/jhs/develop/anjuta-trunk/manuals/anjuta-manual ;
Mar 2 10:17:01 Obelix CRON[24165]: pam_unix(cron:session): session opened for user root by (uid=0)
Mar 2 10:17:01 Obelix CRON[24165]: pam_unix(cron:session): session closed for user root
Mar 2 10:27:50 Obelix gdm[5564]: pam_unix(gdm-autologin-session): session closed for user jhs
Mar 2 12:04:30 Obelix sshd[5031]: Server listening on : port 22.
Mar 2 12:04:40 Obelix gdm[5564]: pam_unix(gdm-autologin-session): session opened for user jhs by (uid=0)
Mar 2 12:04:41 Obelix gdm[5564]: pam_unix(gdm-autologin-session): session closed for user jhs
Mar 2 12:04:41 Obelix gdm[5564]: pam_unix(gdm-autologin-session): session opened for user root by (uid=0)
Mar 2 12:04:41 Obelix gdm[5564]: pam_unix(gdm-autologin-session): session closed for user root
Mar 2 12:17:01 Obelix CRON[19507]: pam_unix(cron:session): session opened for user root by (uid=0)
Mar 2 12:17:01 Obelix CRON[19507]: pam_unix(cron:session): session closed for user root
Mar 2 12:45:36 Obelix gdm[5536]: pam_unix(gdm-autologin-session): session closed for user jhs
Mar 2 18:04:05 Obelix sshd[5055]: Server listening on : port 22.
Mar 2 18:04:05 Obelix sshd[5055]: Server listening on 0.0.0.0 port 22.
Mar 2 18:04:15 Obelix gdm[5560]: pam_unix(gdm-autologin-session): session opened for user jhs by (uid=0)
Mar 2 18:04:16 Obelix gdm[5560]: pam_unix(gdm-autologin-session): session closed for user jhs
Mar 2 18:04:16 Obelix gdm[5560]: pam_unix(gdm-autologin-session): session opened for user root by (uid=0)
Mar 2 18:17:01 Obelix CRON[20164]: pam_unix(cron:session): session opened for user root by (uid=0)
Mar 2 18:17:01 Obelix CRON[20164]: pam_unix(cron:session): session closed for user root
Mar 2 18:35:01 Obelix sudo: jhs : TTY=pts/0 ; PWD=/home/jhs/openismus/libgda ; USER=root ; COMMAND=/usr/
Mar 2 18:39:21 Obelix sudo: jhs : TTY=pts/0 ; PWD=/home/jhs/openismus/libgda ; USER=root ; COMMAND=/usr/
Mar 2 18:39:57 Obelix sudo: jhs : TTY=pts/0 ; PWD=/home/jhs/openismus/libgda ; USER=root ; COMMAND=/usr/
Mar 2 18:42:29 Obelix sudo: jhs : TTY=pts/0 ; PWD=/home/jhs/openismus/libgda ; USER=root ; COMMAND=/usr/
Mar 2 18:45:10 Obelix sudo: jhs : TTY=pts/0 ; PWD=/home/jhs/openismus/libgda ; USER=root ; COMMAND=/usr/
Mar 2 18:49:33 Obelix sudo: jhs : TTY=pts/0 ; PWD=/home/jhs/openismus/libgda ; USER=root ; COMMAND=/usr/
Mar 2 19:17:01 Obelix CRON[14546]: pam_unix(cron:session): session opened for user root by (uid=0)
Mar 2 19:17:01 Obelix CRON[14546]: pam_unix(cron:session): session closed for user root
Mar 2 20:02:01 Obelix CRON[23521]: pam_unix(cron:session): session opened for user root by (uid=0)
Mar 2 20:02:01 Obelix CRON[23521]: pam_unix(cron:session): session closed for user root
Mar 2 20:03:37 Obelix sudo: jhs : TTY=pts/8 ; PWD=/home/jhs/develop/anjuta-trunk/plugins/sourceview ; USER=
Mar 2 20:08:53 Obelix sudo: jhs : TTY=pts/7 ; PWD=/home/jhs/develop/anjuta-trunk/plugins/sourceview ; USER=
Mar 2 20:17:01 Obelix CRON[2668]: pam_unix(cron:session): session opened for user root by (uid=0)
Mar 2 20:17:01 Obelix CRON[2668]: pam_unix(cron:session): session closed for user root
Mar 2 20:17:36 Obelix sudo: jhs : TTY=pts/6 ; PWD=/home/jhs/develop/anjuta-trunk ; USER=root ; COMMAND=/b/
Mar 2 20:36:30 Obelix sudo: pam_unix(sudo:auth): authentication failure; logname=jhs uid=0 euid=0 tty=/dev/p/
```

Gestión de procesos. dmesg (/var/log/syslog).

- Mensajes del Kernel
 - Kernel es el primer programa cargado en memoria, del que los demás procesos hacen uso para acceder a los recursos del sistema.
- La salida de dmesg se envía a /var/log/syslog
 - Antes se llamaba /var/log/messages
- Ejemplos para buscar fallas de algunos programas u otros elementos:
 - `cat /var/log/syslog | grep warn | less`
 - `dmesg | grep -i memory`
- Ejemplos y ejercicios de syslog:
 - http://homepage.smc.edu/morgan_david/linadmin/labs/syslog-exercise.htm

Gestión de procesos. Características del proceso.

- Process ID (PID) Group ID (GID) User ID (UID).
- Tiene una estructura jerárquica (unos procesos dependen de otros).
- Suelen tener estado: runnable sleeping ...
- Tiene permisos y propietario



Gestión de procesos. Ps.

- Muestra información sobre el estado de los procesos.
 - UID PID PPID C STIME TTY TIME CMD
- Una ejecución común es ps -eaf:
 - (-e muestra todos los procesos)
 - (-a muestra los de otros usuarios y tty)
 - (-f muestra opciones completas)
- Guía de ps:
 - https://www.linuxtotal.com.mx/?cont=info_admon_012



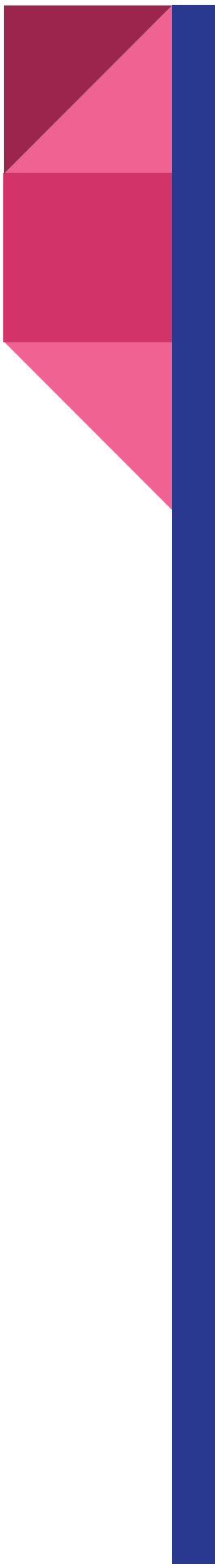
Gestión de procesos. Pstree.

- Muestra en vista de árbol (de forma jerárquica) una lista de los procesos en ejecución

- <https://www.serverdensity.com/monitor/linux/how-to/>

- El objetivo es más saber que procesos dependen de otros.

```
init--GoogleTalkPlug--3*[{GoogleTalkPlug}]
      |NetworkManager--dhclient
      |                  |2*[{NetworkManager}]
      |                  |
      |--acpid
      |--apache2--8*[apache2]
      |--atd
      |--avahi-daemon--avahi-daemon
      |--bamd
      |--bluetoothd
      |--bonobo-activati--2*[{bonobo-activat}]
      |--chrome
      |   |--chrome--16*[{chrome}]
      |   |--chrome--1*[{chrome}]
      |   |--21*[{chrome}]
      |   |--chrome--11*[{chrome--2*[{chrome}]]
      |   |--clock-applet--[{clock-applet}]
      |   |--console-kit-dae--64*[{console-kit-da}]
      |   |--cron
      |   |--cupsd
      |   |--daemon--java--22*[{java}]
      |   |--2*[dbus-daemon]
      |   |--dbus-launch
      |   |--dropbox--13*[{dropbox}]
```



Gestión de procesos. Ejemplo básico.

- En ocasiones un programa o proceso puede dar problemas.
 - Tenemos que identificar en qué estado está (ps), si consume mucha memoria (htop), si ha habido problemas al arrancar (dmesg (/var/log/syslog))...
- Una forma de solucionarlo en “matando” el proceso.
- En terminal:
 - `ps -eaf | grep <programa>`
 - Obtener el PID (2º dato)
 - `kill -9 <PID>`
- Si no podemos acceder a terminal: Ctrl + Alt + F1 para entrar en una sesión de consola (volver Ctrl + Alt + F7)



para entrar

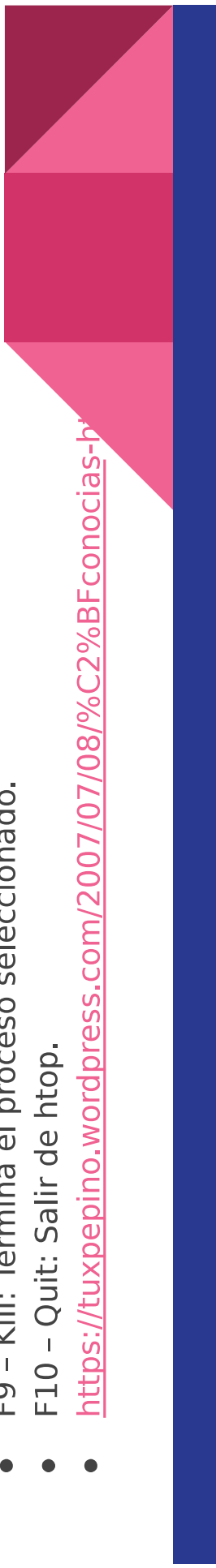
Gestión de procesos. Top/Htop.

- Muestra los recursos usados por tu sistema, procesos, métricas, etc.
- **htop** es una versión mejorada actualmente usada.
- Permite ordenar, parar, ver consumo en tiempo real, etc.
- Mejora las anteriores herramientas reuniendo varias de sus características (información de los procesos, vista jerárquica, etc.).
- Guía/ejemplos:
 - <http://www.thegeekstuff.com/2011/09/linux-htop-examples/>
 - <https://hipertextual.com/archivo/2010/03/comando-linux-htop-adaptado-para-interactiva-mente-lo-s-procesos-del-sistema/>



Gestión de procesos. Top/Htop.

- F1 – Help: Se lanzará en el navegador por defecto una página con la ayuda del programa.
- F2 – Setup: Configuración del programa, bastante completa y personalizable.
- F3 – Search: Busca un proceso por su nombre.
- F4 – Invert: Invierte el orden en el que se muestran los procesos. Si estás viendo, por ejemplo, de mayor a menor consumo de memoria, al pulsar F4 verás primero los que menos consumen.
- F5 – Tree: Al pulsar F5 se mostrarán los procesos en árbol.
- F6 – SortBy: Ordena los procesos teniendo en cuenta las diferentes columnas. Esto se puede hacer haciendo click con el ratón en las columnas.
- F7 – Nice –: Disminuye la prioridad del proceso seleccionado.
- F8 – Nice +: Al contrario que el anterior.
- F9 – Kill: Termina el proceso seleccionado.
- F10 – Quit: Salir de htop.
- <https://tuxpepino.wordpress.com/2007/07/08/%C2%BFconocias-htop/>



Gestión de procesos. Top/Htop.

CPU: 2.0%		Tasks: 16 total, 1 running									
Mem: 13/123MB		Load average: 0.37 0.12 0.04									
Swap: 0/109MB		Uptime: 00:00:50									
PID	USER	PRI	NI	UIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
3692	per	15	0	2424	1204	980	R	2.0	1.0	0:00.24	htop
1	root	16	0	2952	1852	532	S	0.0	1.5	0:00.77	/sbin/init
2236	root	20	-1	2316	728	472	S	0.0	0.6	0:01.06	/sbin/udevd --daemon
3224	dhcp	18	-2	2412	552	244	S	0.0	0.4	0:00.00	dhclient3 -e IF_ME
3488	root	18	0	1692	516	448	S	0.0	0.4	0:00.00	/sbin/getty 38400
3491	root	18	0	1696	520	448	S	0.0	0.4	0:00.01	/sbin/getty 38400
3497	root	18	0	1696	516	448	S	0.0	0.4	0:00.00	/sbin/getty 38400
3500	root	18	0	1692	516	448	S	0.0	0.4	0:00.00	/sbin/getty 38400
3501	root	16	0	2772	1196	936	S	0.0	0.9	0:00.04	/bin/login --
3504	root	18	0	1696	516	448	S	0.0	0.4	0:00.00	/sbin/getty 38400
3539	syslog	15	0	1916	704	564	S	0.0	0.6	0:00.12	/sbin/syslogd -u s
3561	root	18	0	1840	536	444	S	0.0	0.4	0:00.79	/bin/dd bs 1 if /p
3563	klog	18	0	2472	1376	408	S	0.0	1.1	0:00.37	/sbin/klogd -P /va
3590	daemon	25	0	1960	428	308	S	0.0	0.3	0:00.00	/usr/sbin/atd
3604	root	18	0	2336	792	632	S	0.0	0.6	0:00.00	/usr/sbin/cron
3645	per	15	0	5524	2924	1428	S	0.0	2.3	0:00.45	-bash
F1Help	F2Setup	F3Search	F4Invert	F5Tree	F6SortBy	F7Nice	-F8Nice	+F9Kill	F10Quit		

Gestión de procesos. Gnome system monitor.



Gestión de procesos. Pautas.

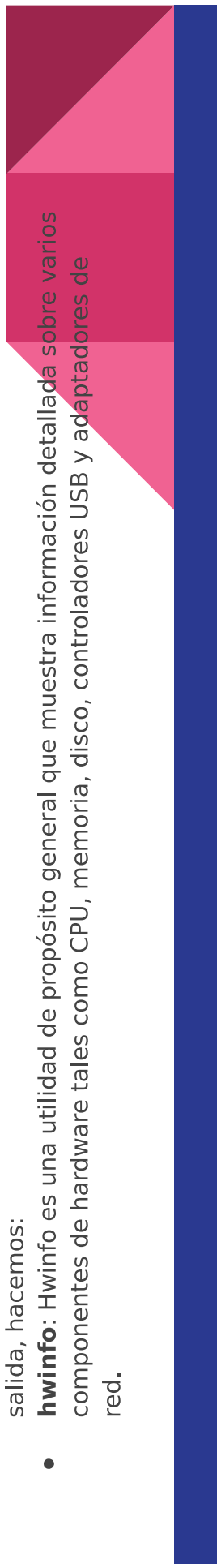
- Comprobaciones para resolver problemas:
 - Comprobar procesos, alguno abusa de los recursos, zombie, ... (htop, gnome system monitor).
 - Comprobar logs (grep, less, tail, herramienta visual).
 - Monitorizar rendimiento (**Sysstat**: mpstat, iostat, sar, uptime, time):
 - <http://www.dsi.uclm.es/personal/AntonioBueno/ESI/monitor%20en%20linux.pdf>
 - <https://www.genbeta.com/linux/como-monitorizar-constantemente-el-rendimiento-de-tu-distro-gnu-linux>
 - Problemas de red (netstat):
 - <http://www.fermu.com/articulos/windows/articulos-y-tutoriales/285-seguridad-en-internet-el-comando-netstat-puertos-y-comunicaciones>
 - Comprobar medidores, temperatura, ventiladores:
 - <https://www.adslzone.net/postt269189.html>
 - <https://blog.desdelinux.net/monitoreo-la-temperatura-voltaje-y-ventiladores-de-tu-ordenador-con-ubuntu/>

Gestión de procesos. Pautas.

- Comprobaciones para resolver problemas:
 - Usuarios en el sistema (w).
 - Problemas de disco (du, hdparm):
 - <http://www.dsi.ucm.es/personal/AntonioBueno/ESI/monitor%20en%20linux.pdf>
 - Fallos en el sistema de archivos (fsck)
 - <http://lasegundapuerta.com/index.php/informatica/linux-y-software-libre/2271-reparar-disco-duro-con-sectores-danados>
 - <http://www.laquialinux.es/e2fsck-reparar-sistema-de-archivos-ext-ext2-ext3-ext4>
 - Archivos perdidos (PhotoRec):
 - http://www.cgsecurity.org/wiki/PhotoRec_Paso_A_Paso
 - <http://www.taringa.net/posts/linux/7190804/Programas-para-recuperar-archivos-en-linux-ext2-ext3-ext4.html>
 - Comprobar memoria (memtest, free)
 - <http://es.ccm.net/faq/159-diagnosticar-el-estado-de-la-ram-del-pc-con-memtest>
 - <https://tuxfiles.wordpress.com/2012/01/07/entendiendo-el-comando-free/>

Gestión de procesos. Otras herramientas informativas.

- <http://andresbarroso.es/ty.es/post/post/32/linux-para-novatos-10-comandos-para-obtener-informacion-del-sistema>
- **lspci**: El comando lspci lista todos los buses PCI y muestra los detalles acerca de los dispositivos conectados a ellos. Por ejemplo, el adaptador VGA, la tarjeta gráfica, el adaptador de red, los puertos USB y controladores SATA.
- **lsusb**: Este comando muestra todos los puertos USB y los detalles acerca de los dispositivos conectados a ellos. Para ver su salida solo debemos ejecutar. La información que imprime el comando por defecto es general.
- **fdisk**: Fdisk es un gestor de particiones. Nos permite modificar las particiones en los discos duros pero también se puede utilizar para enumerar la información de la partición así como su cantidad de bloques y sistema de archivos.
- **lscpu**: El comando lscpu muestra información sobre el procesador del sistema, como su velocidad, número de núcleos y fabricante. No tiene ninguna opción o funcionalidades adicionales. Para ver su salida, hacemos:
- **hwinfo**: Hwinfo es una utilidad de propósito general que muestra información detallada sobre varios componentes de hardware tales como CPU, memoria, disco, controladores USB y adaptadores de red.



Gestión de procesos. Información extra.

- <https://blog.desdelinux.net/donde-se-encuentran-los-archivos-de-registro-logs-en-linux/>
- <http://www.tuxradar.com/content/how-fix-most-common-linux-problems>
- <https://itvision.altervista.org/why.linux.is.not.ready.for.the.desktop.current.html>
- <https://www.digitalocean.com/community/tutorials/how-to-troubleshoot-common-site-issues-on-a-linux-server>
- <https://www.vidarholen.net/contents/linuxtips/>
- <https://www.digitalocean.com/community/tutorials/how-to-troubleshoot-common-site-issues-on-a-linux-server>
- <http://www.tuxradar.com/content/linux-tips-every-geek-should-know>
- <https://wiki.ubuntu.com/Recuperar%20Grub> (cuando instalas Windows después de Linux)
- <https://linux-tips.com/>
- <https://unix.stackexchange.com/questions/50098/linux-network-troubleshooting-and-debugging>
- <http://www.dedoimedo.com/computers/linux-hardware-troubleshooting.html>
- <http://lasegundapuerta.com/index.php/informatica/linux-y-software-libre/271-reparar-disco-duro-con-sectores-danados>
- <http://www.hirensbootcd.org/>
- <http://es.ccm.net/faq/3435-linux-comandos-para-monitorear-el-sistema>
- <https://gist.github.com/celtha/1214442> (script de monitorización del sistema)
- <https://blog.desdelinux.net/script-muestra-informacion-del-sistema/>
- <https://www.ibaivalencia.com/script-monitorizar-carga-cpu-servidor-linux/>
- <https://www.tecmint.com/linux-server-health-monitoring-script/>
- <https://www.cyberciti.biz/tips/shell-script-to-watch-the-disk-space.html>
- <http://www.systeme.com/2016/05/07/bash-script-monitor-cpu-memory-disk-usage-linux/>
- <http://www.solarwinds.com/documentation/en/flarehelp/sam/content/sam-linux-unix-script-monitor-sw3260.htm>
- <https://www.activexperts.com/support/network-monitor/online/linux/>