2020

Practicas 1-4

ADMINISTRACION DE SISTEMAS Y REDES

AITOR LLANOS – IRAZOLA UO264476

Contenido

Practica 1

1.2.- Systemd

Entregar: Anota el PID del proceso systemd. Anota el runlevel en el que está el sistema, antes y después de hacer systemctl isolate (comando who –a). ¿Qué diferencias hay en el arranque cuando el runlevel por defecto es 1? ¿Qué ocurre al hacer systemctl isolate runlevel6? target?

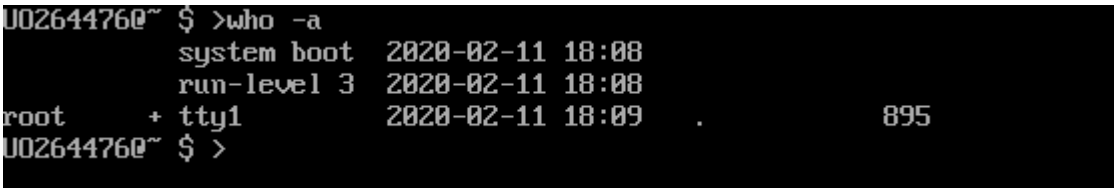
Ejecutamos el comando ps ax y mostramos la lista de procesos. El PID del proceso "systemd" se asocia al 1.

```
UO264476@~ $ >ps ax
  PID TTY      STAT   TIME COMMAND
    1 ?        Ss     0:01 /usr/lib/systemd/systemd --switched-root --system --deserialize 17
    2 ?        S      0:00 [kthreadd]
    3 ?        I<     0:00 [rcu_gp]
    4 ?        I<     0:00 [rcu_par_gp]
    5 ?        I      0:00 [kworker/0:0-ata_sff]
    6 ?        I<     0:00 [kworker/0:0H-kblockd]
    8 ?        I<     0:00 [mm_percpu_wq]
    9 ?        S      0:00 [ksoftirqd/0]
   10 ?        R      0:00 [rcu_sched]
   11 ?        S      0:00 [migration/0]
   12 ?        S      0:00 [watchdog/0]
   13 ?        S      0:00 [cpuhp/0]
   15 ?        S      0:00 [kdevtmpfs]
   16 ?        I<     0:00 [netns]
   17 ?        S      0:00 [kauditd]
   18 ?        S      0:00 [khungtaskd]
   19 ?        S      0:00 [oom_reaper]
   20 ?        I<     0:00 [writeback]
   21 ?        S      0:00 [kcompactd0]
   22 ?        SN     0:00 [ksmd]
   23 ?        SN     0:00 [khugepaged]
   24 ?        I<     0:00 [crypto]
   25 ?        I<     0:00 [kintegrityd]
   26 ?        I<     0:00 [kblockd]
   27 ?        I<     0:00 [tpm_dev_wq]
   28 ?        I<     0:00 [md]
   29 ?        I<     0:00 [edac-poller]
   30 ?        S      0:00 [watchdogd]
   39 ?        S      0:00 [kswapd0]
   58 ?        I      0:00 [kworker/u2:1-events_unbound]
   90 ?        I<     0:00 [kthrotld]
   91 ?        I<     0:00 [acpi_thermal_pm]
   92 ?        I<     0:00 [kmpath_rdacd]
   93 ?        I<     0:00 [kaluad]
   94 ?        I<     0:00 [ipv6_addrconf]
```

Posteriormente ejecutamos la orden systemctl get-default para comprobar que el sistema esta en modo multi usuario.

```
 1583 tty1     R+     0:00 ps ax
UO264476@~ $ >systemctl get-default
multi-user.target
UO264476@~ $ >
```

Y aplicamos el comando who -a para obtener el run-level actual del sistema 3.



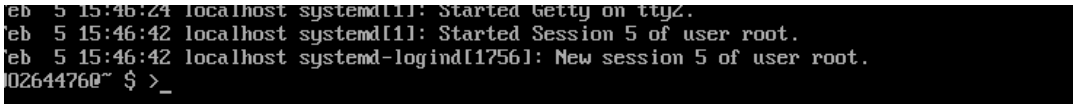
Aplicamos el comando *systemctl isolate runlevel1.target* para cambiar a modo single-user, y aplicamos el comando who -a para ver que efectivamente el runlevel es 1.

¿Qué ocurre al hacer *systemctl isolate runlevel6? target*?

La máquina se reinicia

1.4.-Syslog

Ejecutamos el comando "last" para ver los inicios recientes de sesión y las caídas del sistema (reboot)



***Entregar*: ¿Cuál es el motivo de la última caída del sistema, de acuerdo con la orden *last*?**

Ahora como indica la practica apagamos la maquina de forma anómala desde VirtualBox y posterior a su inicio ejecutamos el comando "last" nuevamente.

```
U0264476@~ $ >last
root     tty1                          Tue Feb 11 18:17   still logged in
reboot   system boot  4.18.0-147.el8.x Tue Feb 11 18:17   still running
root     tty3                          Tue Feb 11 18:16 - crash  (00:01)
root     tty3                          Tue Feb 11 18:15 - 18:15  (00:00)
root     tty1                          Tue Feb 11 18:14 - crash  (00:02)
reboot   system boot  4.18.0-147.el8.x Tue Feb 11 18:12   still running
root     tty1                          Tue Feb 11 18:09 - 18:11  (00:02)
reboot   system boot  4.18.0-147.el8.x Tue Feb 11 18:08 - 18:12  (00:03)
root     tty2                          Wed Feb  5 15:56 - crash (6+02:11)
root     tty1                          Wed Feb  5 15:53 - crash (6+02:15)
reboot   system boot  4.18.0-147.el8.x Wed Feb  5 15:51 - 18:12 (6+02:21)
root     tty2                          Wed Feb  5 15:46 - 15:50  (00:03)
root     tty1                          Wed Feb  5 15:45 - 15:50  (00:04)
root     tty1                          Wed Feb  5 15:44 - 15:45  (00:00)
reboot   system boot  4.18.0-147.el8.x Wed Feb  5 15:44 - 15:50  (00:05)
root     tty1                          Wed Feb  5 15:42 - 15:44  (00:02)
root     tty1                          Wed Feb  5 15:31 - 15:41  (00:10)
reboot   system boot  4.18.0-147.el8.x Wed Feb  5 15:31 - 15:44  (00:13)
root     tty1                          Wed Jan 29 18:05 - 18:28  (00:23)
reboot   system boot  4.18.0-147.el8.x Wed Jan 29 18:05 - 15:44 (6+21:39)
```

Al ejecutar el comando “last” vemos que la causa del apagado es un “CRASH”.

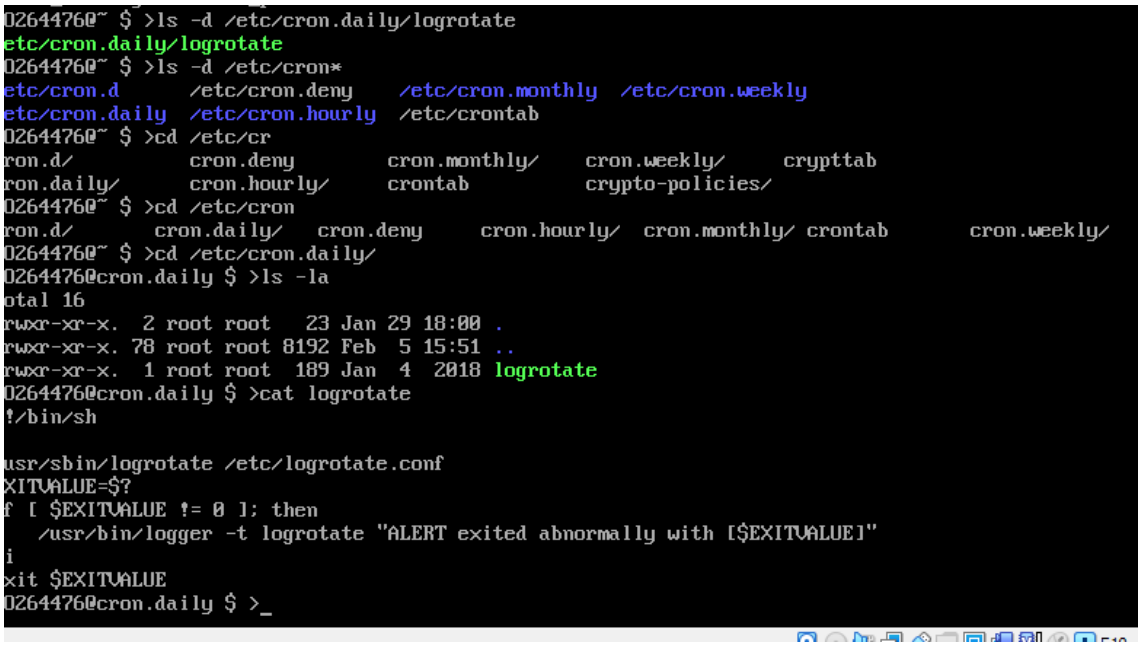
1.5.-Ejecucion periódica de comandos

Entregar: **En el directorio /etc/cron.daily hay un script encargado de borrar los ficheros con los logs más antiguos y de rotar cada día los ficheros de log. ¿Cuál es el nombre de este script?**

Accedemos al repositorio /etc/ y mostramos todas las posibles opciones de cron con /etc/cron*

```
U0264476@~ $ >ls -d /etc/cron*
ron.d/       cron.daily/   cron.deny     cron.hourly/  cron.monthly/ crontab       cron.weekly/
```

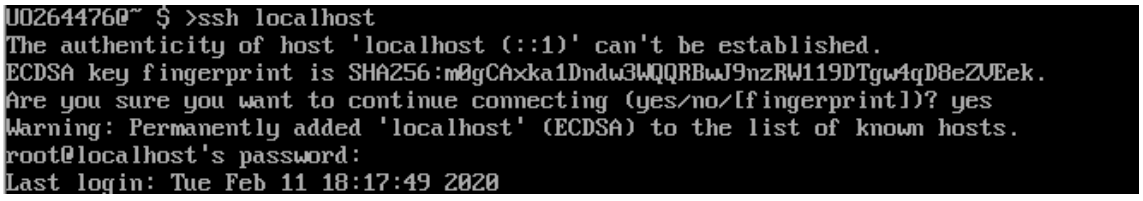
Posteriormente accedemos al cron. daily y vemos que contiene un fichero LOGROTATE, este fichero como podemos comprobar con el cat posterior es el encargado de borrar los ficheros con los logs más antiguos y de rotar cada día los ficheros de log



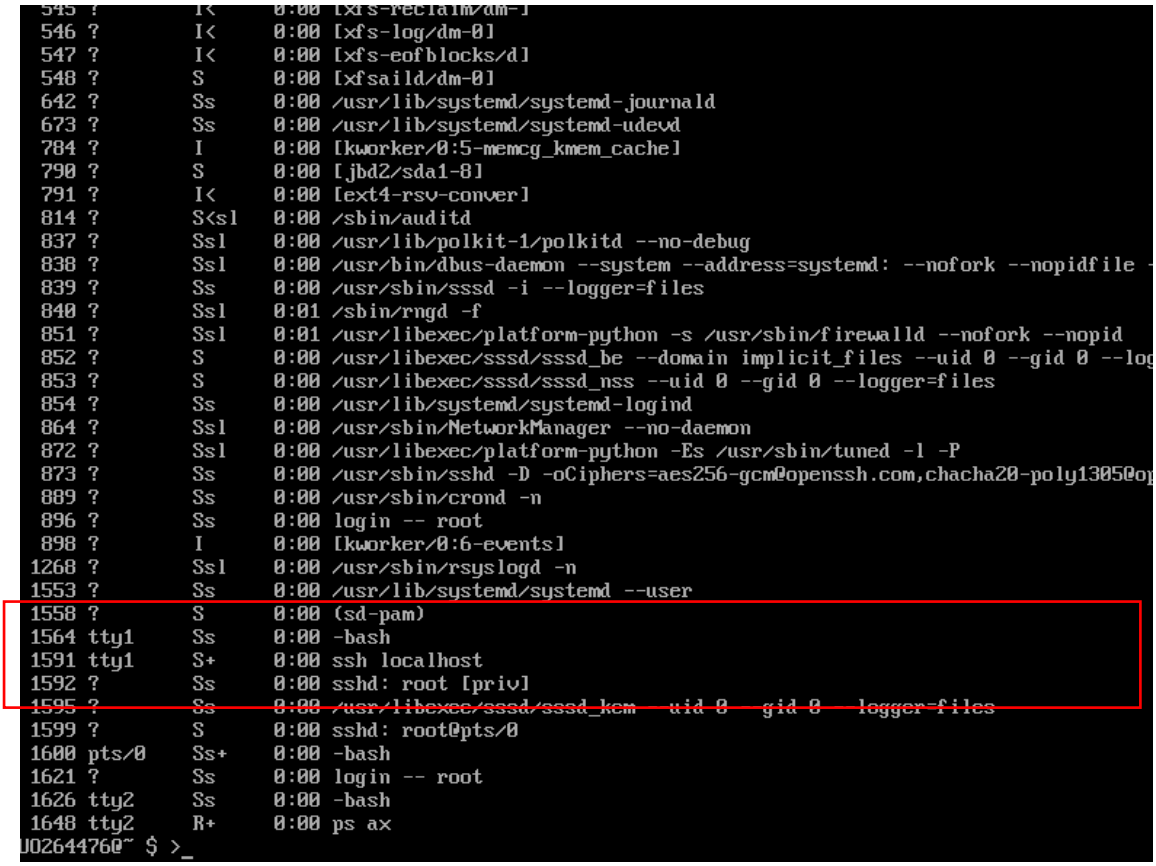
1.6.- Login desde red

***Entregar:* Haz ssh localhost desde la terminal a la que has accedido con ALT-F2. Haz *ps ax* y busca los procesos *sshd*. ¿En qué terminal figura el segundo proceso *sshd*?**

Hacemos el ssh localhost para conectarnos con ssh y hacer una nueva sesión



Hacemos “ps ax” en una nueva terminal para ver que existe el proceso sshd.

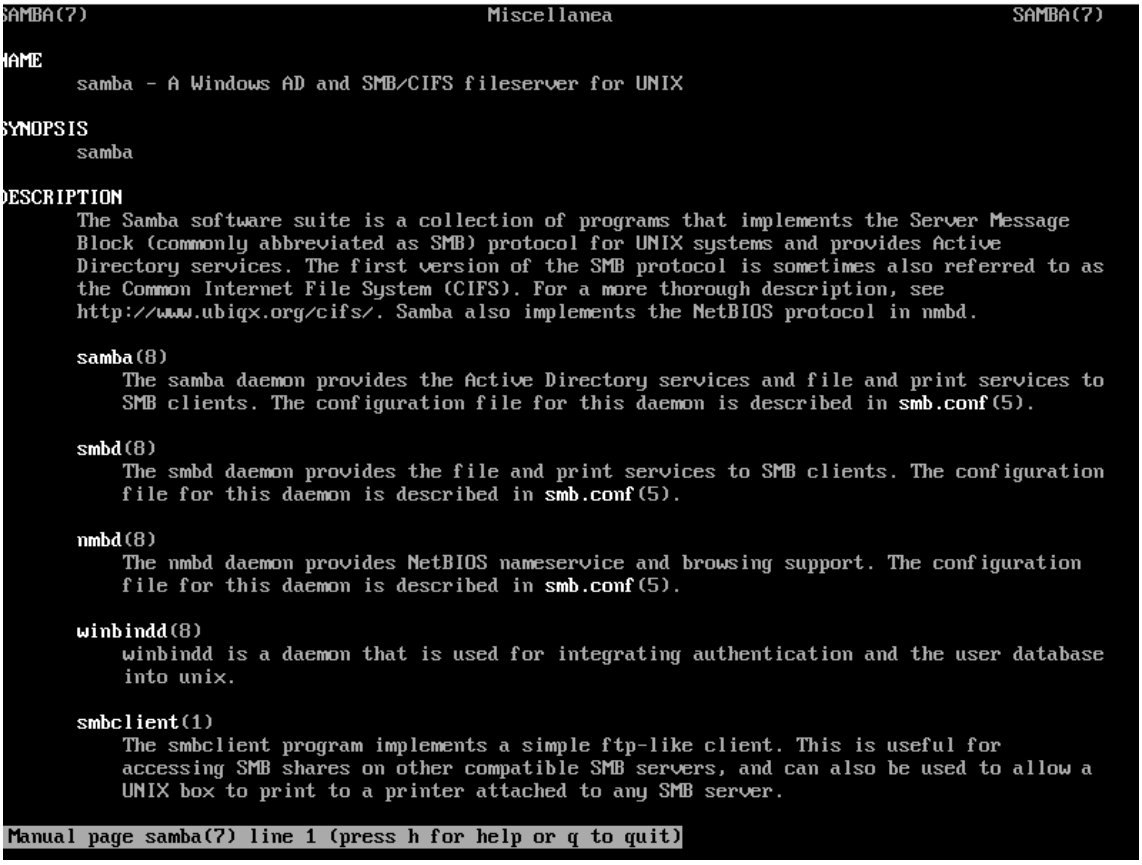


El segundo proceso ssh localhost se ha creado en el mismo terminal en el que se ha ejecutado el proceso, en el terminal 2.

1.7.- Sistemas de ficheros en red

Instalamos samba

Mostramos el man de samba



1.8.- Correo electrónico

Entregar: Consulta la ayuda de mail. ¿Cuál es el comando para salir de la orden mail?

Para salir de la orden mail hay que ejecutar el comando q.

```
MAILX(1)                          User Commands                          MAILX(1)

NAME
       mailx - send and receive Internet mail

SYNOPSIS
       mailx [-BDdEFintv~] [-s subject] [-a attachment ] [-c cc-addr] [-b bcc-addr] [-r from-
             addr] [-h hops] [-A account] [-S variable[=value]] to-addr . . .
       mailx [-BDdeEHiInNRv~] [-T name] [-A account] [-S variable[=value]] -f [name]
       mailx [-BDdeEinNRv~] [-A account] [-S variable[=value]] [-u user]

DESCRIPTION
       Mailx is an intelligent mail processing system, which has a command syntax reminiscent  of
       ed(1)  with  lines replaced by messages.  It is based on Berkeley Mail 8.1, is intended to
       provide the functionality of the POSIX mailx command,  and  offers  extensions  for  MIME,
       IMAP,  POP3, SMTP, and S/MIME.  Mailx provides enhanced features for interactive use, such
       as caching and disconnected operation for IMAP, message threading, scoring, and filtering.
       It is also usable as a mail batch language, both for sending and receiving mail.

       The following options are accepted:

       -A name
              Executes  an account command (see below) for name after the startup files have been
              read.

       -a file
              Attach the given file to the message.

       -B     Make standard input and standard output line-buffered.

       -b address
              Send blind carbon copies to list.  List should be a comma-separated list of names.

       -c address
              Send carbon copies to list of users.

Manual page mail(1) line 1 (press h for help or q to quit)
```

1.9.- servicios de impresión

Busca CUPS en Wikipedia.

Common Unix Printing System

Common Unix Printing System (Sistema de impresión común de Unix, abreviado **CUPS**) es un sistema de impresión modular para sistemas operativos de tipo Unix que permite que un computador actúe como servidor de impresión. Un computador que ejecuta CUPS actúa como un servidor que puede aceptar tareas de impresión desde otros computadores clientes, los procesa y los envía a la impresora apropiada.

CUPS está compuesto por una cola de impresión con su planificador, un sistema de filtros que convierte datos para imprimir hacia formatos que la impresora conozca, y un sistema de soporte que envía los datos al dispositivo de impresión. CUPS utiliza el protocolo IPP (Internet Printing Protocol) como base para el manejo de tareas de impresión y de colas de impresión. También provee los comandos tradicionales de línea de comandos de impresión de los sistemas Unix, junto a un soporte limitado de operaciones bajo el protocolo server message block (SMB). Los controladores de dispositivos de impresión que CUPS provee pueden ser configurados utilizando archivos de texto con formato PostScript Printer Description (PPD) de Adobe Systems. Existen varias interfaces de usuario para diferentes plataformas para configurar CUPS; cuenta también con una interfaz como aplicación Web. CUPS es software libre y se distribuye bajo licencia GNU General Public License y GNU Lesser General Public License, versión 2.

Índice [ocultar]

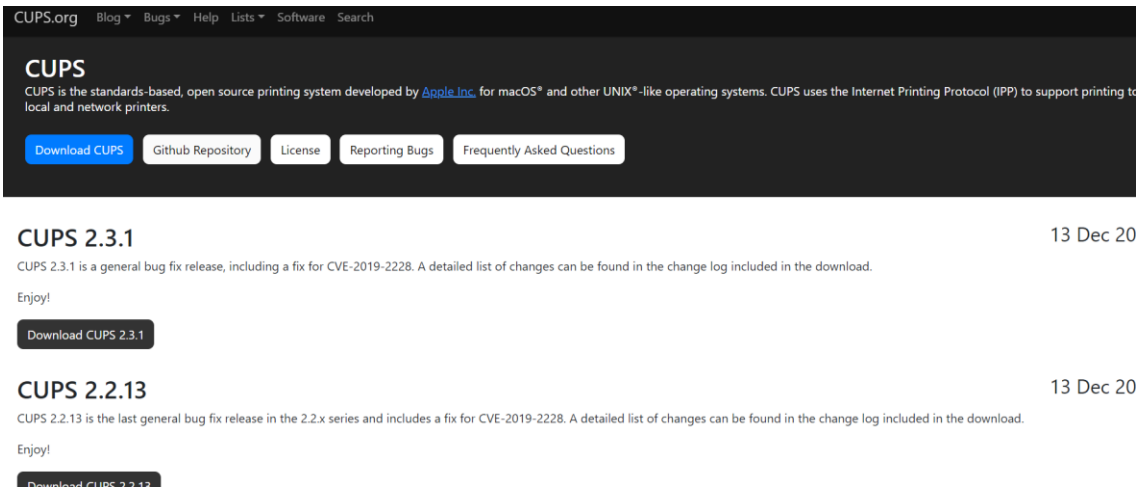
Common Unix Printing System

www.cups.org

[[Archivo:CUPS 1.4.6 admin rus.png
CUPS Logo|245px]]

Tipo de programa	Servidor de impresión software libre
Desarrollador	Michael Sweet Apple
Autor	Michael Sweet (Easy Software Products)
Lanzamiento	9 de junio de 1999
Última versión estable	2.3.1 (info) 13 de diciembre de 2019 (1 mes y 29 días)
Género	Servidor de impresión
Programado en	C C++
Sistema operativo	tipo-Unix
Licencia	GNU GPL GNU LGPL

Entra en www.cups.org para obtener más información.



Opcionales Practica 1

2) Documentación y ayuda:

1.1) Ejecuta el comando *mandb*

Instalamos el man-pages

```
[root@\ U0264476 ]$ yum install man-pages
Last metadata expiration check: 0:00:37 ago on Sun 23 Feb 2020 01:07:30 PM EST.
Dependencies resolved.
================================================================================
 Package                 Architecture    Version              Repository    Size
================================================================================
Installing:
 man-pages               x86_64          4.15-6.el8           BaseOS       5.9 M
Installing weak dependencies:
 man-pages-overrides     noarch          8.1.0.0-2.el8        AppStream     17 k

Transaction Summary
================================================================================
Install  2 Packages

Total download size: 6.0 M
Installed size: 5.5 M
Is this ok [y/N]: y
Downloading Packages:
(1/2): man-pages-overrides-8.1.0.0-2.el8.noarch.rpm  77 kB/s |  17 kB     00:00
(2/2): man-pages-4.15-6.el8.x86_64.rpm               2.9 MB/s | 5.9 MB     00:02
--------------------------------------------------------------------------------
Total                                                2.0 MB/s | 6.0 MB     00:02
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Installing       : man-pages-overrides-8.1.0.0-2.el8.noarch               1/2
  Installing       : man-pages-4.15-6.el8.x86_64 [=========================] 2/2
```

Ejecutamos el comando mandb

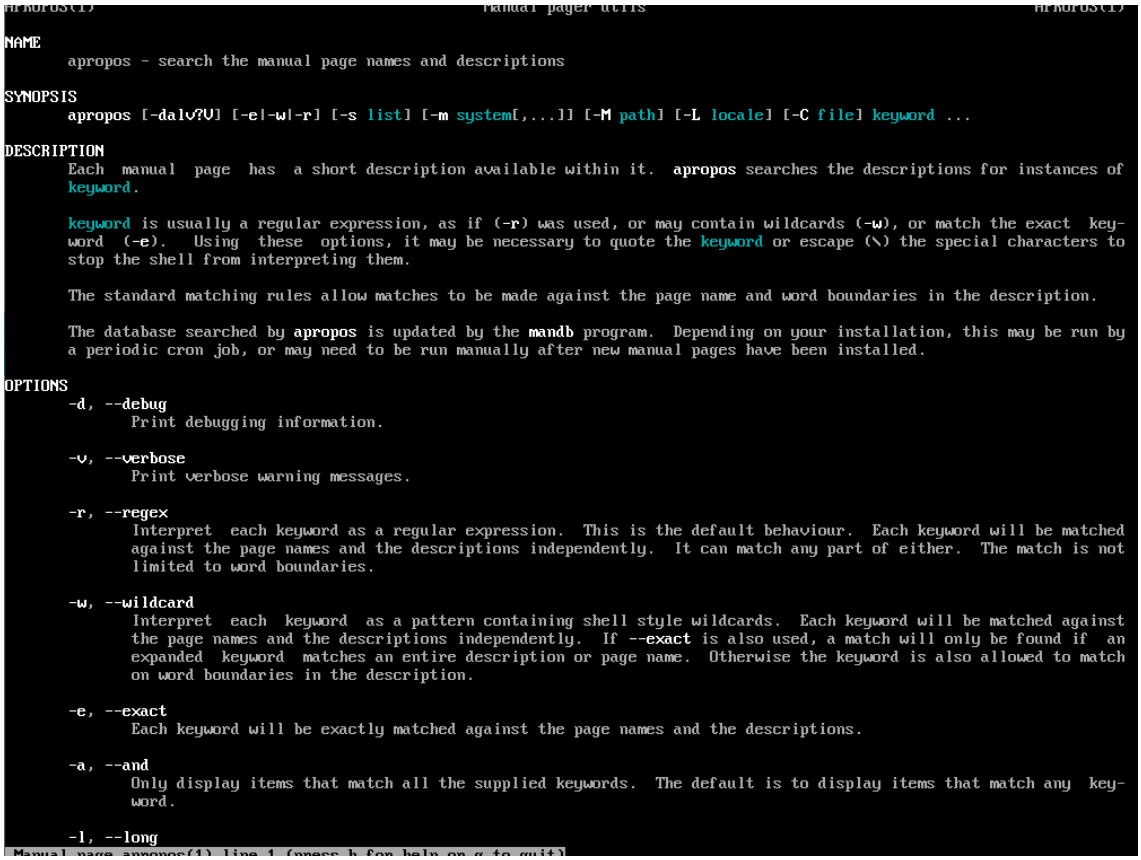
```
[root@\ UO264476~]$ mandb
Purging old database entries in /usr/share/man/overrides...
Processing manual pages under /usr/share/man/overrides...
Purging old database entries in /usr/share/man...
Processing manual pages under /usr/share/man...
Purging old database entries in /usr/share/man/ru...
Processing manual pages under /usr/share/man/ru...
Purging old database entries in /usr/share/man/fr...
Processing manual pages under /usr/share/man/fr...
Purging old database entries in /usr/share/man/ja...
Processing manual pages under /usr/share/man/ja...
Purging old database entries in /usr/share/man/ko...
Processing manual pages under /usr/share/man/ko...
Purging old database entries in /usr/share/man/pl...
Processing manual pages under /usr/share/man/pl...
Purging old database entries in /usr/share/man/sk...
Processing manual pages under /usr/share/man/sk...
Purging old database entries in /usr/share/man/cs...
Processing manual pages under /usr/share/man/cs...
Purging old database entries in /usr/share/man/da...
Processing manual pages under /usr/share/man/da...
Purging old database entries in /usr/share/man/de...
Processing manual pages under /usr/share/man/de...
Purging old database entries in /usr/share/man/hu...
Processing manual pages under /usr/share/man/hu...
Purging old database entries in /usr/share/man/id...
Processing manual pages under /usr/share/man/id...
Purging old database entries in /usr/share/man/it...
Processing manual pages under /usr/share/man/it...
Purging old database entries in /usr/share/man/pt_BR...
Processing manual pages under /usr/share/man/pt_BR...
```

1.2) Usa las órdenes *man* e *info* para conocer el significado de los términos *whatis* y *apropos* y haz una lista de las órdenes del sistema que hacen referencia al término *reboot*. Escribe el comando que necesitas para mostrar cada una de las páginas de manual que aparece en esa lista.

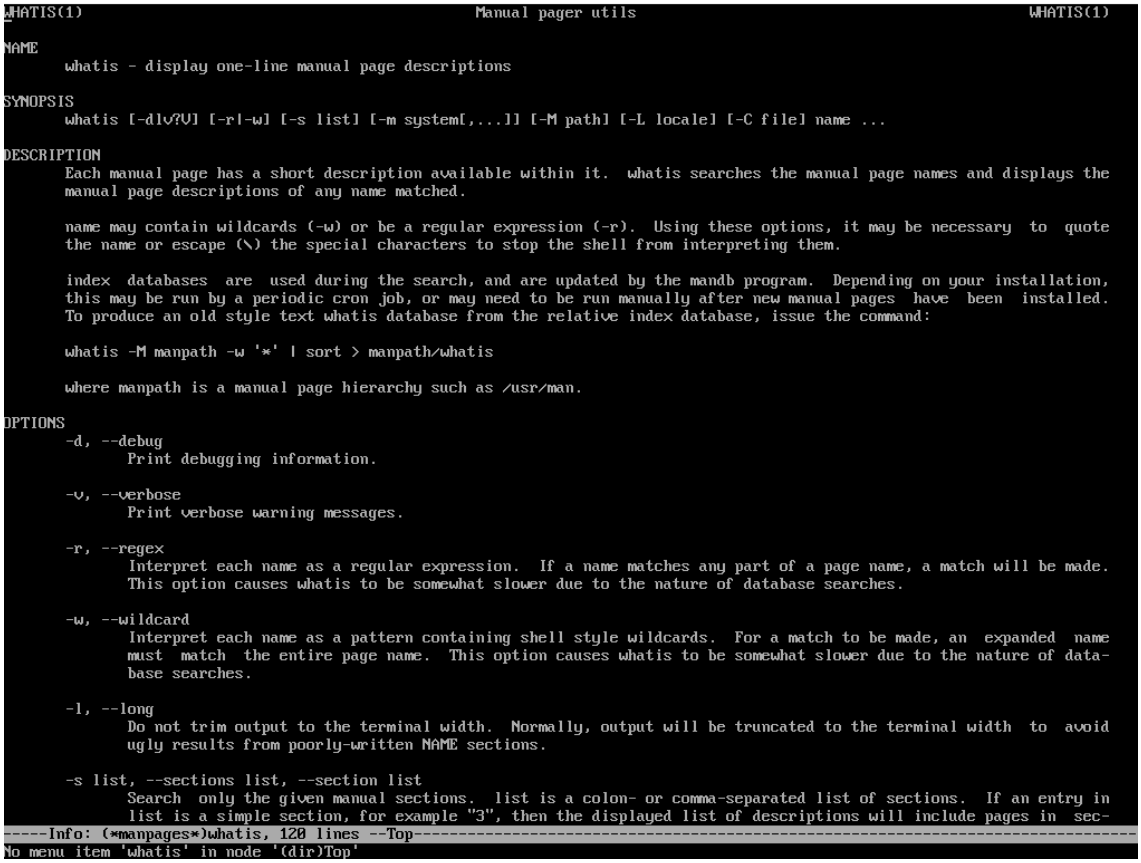
Primero ejecutamos el comando man whatis y sale la siguiente pantalla. Obtenemos que Whatis imprime descripciones de páginas de manual.

```
WHATIS(1)                    Manual pager utils                    WHATIS(1)

NAME
       whatis - display one-line manual page descriptions

SYNOPSIS
       whatis [-dlv?V] [-r|-w] [-s list] [-m system[,...]] [-M path] [-L locale] [-C file] name ...

DESCRIPTION
       Each manual page has a short description available within it.  whatis searches the manual page names and displays the
       manual page descriptions of any name matched.

       name may contain wildcards (-w) or be a regular expression (-r).  Using these options, it may be necessary  to  quote
       the name or escape (\) the special characters to stop the shell from interpreting them.

       index  databases  are  used during the search, and are updated by the mandb program.  Depending on your installation,
       this may be run by a periodic cron job, or may need to be run manually after new manual pages  have  been  installed.
       To produce an old style text whatis database from the relative index database, issue the command:

       whatis -M manpath -w '*' | sort > manpath/whatis

       where manpath is a manual page hierarchy such as /usr/man.

OPTIONS
       -d, --debug
              Print debugging information.

       -v, --verbose
              Print verbose warning messages.

       -r, --regex
              Interpret each name as a regular expression.  If a name matches any part of a page name, a match will be made.
              This option causes whatis to be somewhat slower due to the nature of database searches.

       -w, --wildcard
              Interpret each name as a pattern containing shell style wildcards.  For a match to be made, an  expanded  name
              must  match  the entire page name.  This option causes whatis to be somewhat slower due to the nature of data-
              base searches.

       -l, --long
              Do not trim output to the terminal width.  Normally, output will be truncated to the terminal width  to  avoid
              ugly results from poorly-written NAME sections.

       -s list, --sections list, --section list
              Search  only the given manual sections.  list is a colon- or comma-separated list of sections.  If an entry in
              list is a simple section, for example "3", then the displayed list of descriptions will include pages in  sec-
              tions "3", "3perl", "3x", and so on; while if an entry in list has an extension, for example "3perl", then the
 Manual page whatis(1) line 1 (press h for help or q to quit)
```

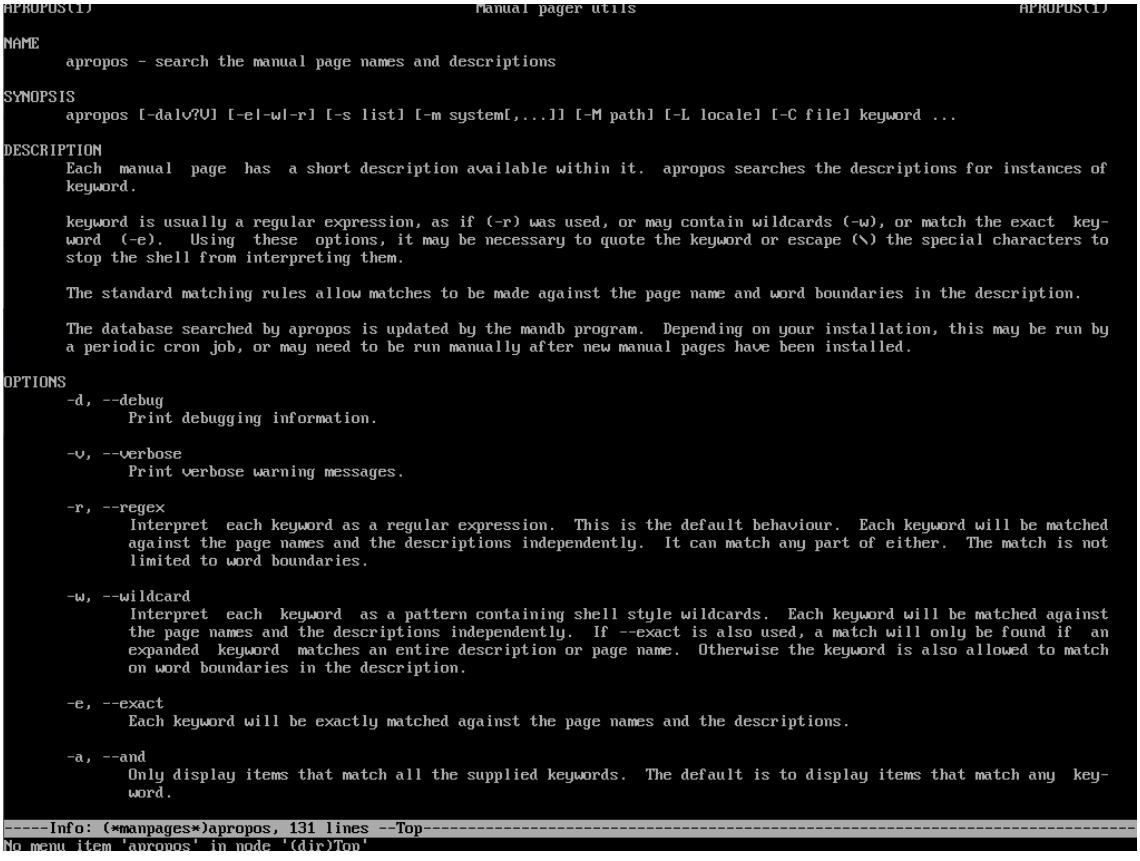
Ahora ejecutamos el comando man apropos. Descubrimos que Apropos busca entre las páginas del manual y las descripciones.



Ahora ejecutamos el comando info whatis y obtenemos prácticamente el mismo resultado

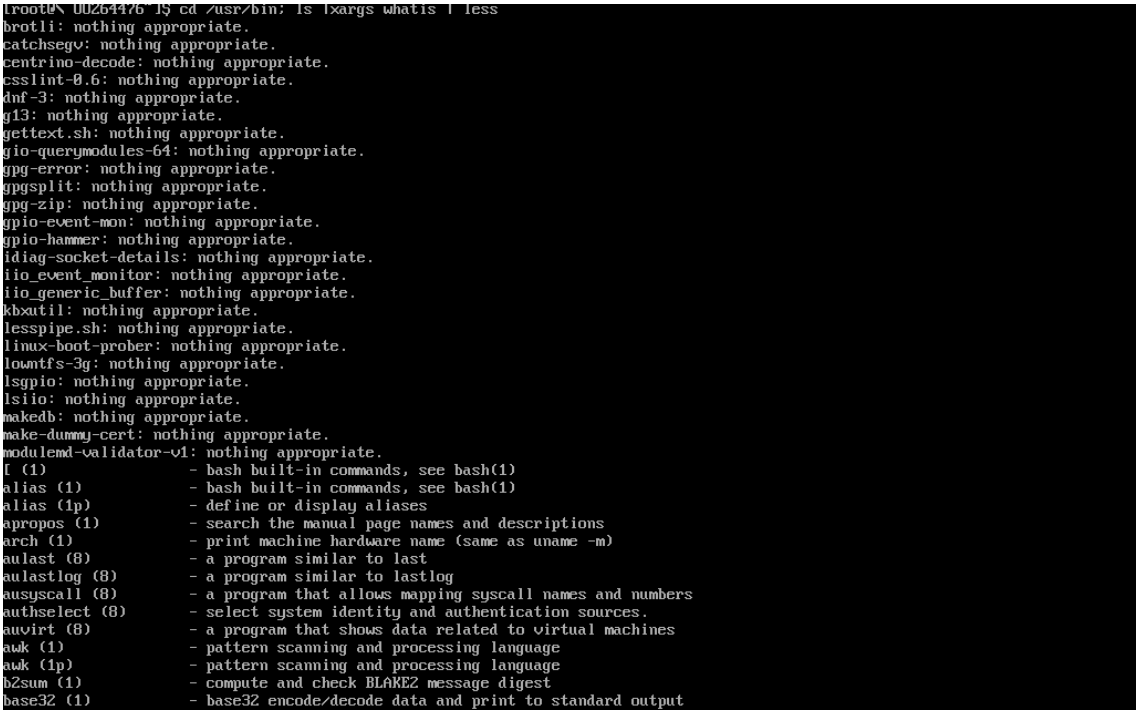


Realizamos lo mismo con info apropos



1.3) Explica qué hace el comando *cd /usr/bin; ls | xargs whatis | less*

La primera parte de este comando está formada por "cd /usr/bin" y nos lleva al directorio bin, que se encuentra dentro del directorio usr. La segunda parte, muestra todas las ordenes que pueden ser ejecutadas por el sistema y una breve explicación sobre cada una. A su vez, la orden less nos permite hacer un pequeño scroll por la información ofrecida.



3) Conceptos básicos de administración de paquetes

2.1) Haz una lista de todos los paquetes del sistema, cuenta cuántos hay con wc

Ejecutando el comando **rpm -qa --last** obtenemos la lista solicitada de paquetes del sistema

```
libmnl-1.0.4-6.el8.x86_64                    Mon 27 Jan 2020 04:01:04 PM EST
libcap-ng-0.7.9-4.el8.x86_64                 Mon 27 Jan 2020 04:01:04 PM EST
grub2-common-2.02-78.el8.noarch              Mon 27 Jan 2020 04:01:04 PM EST
gmp-6.1.2-10.el8.x86_64                      Mon 27 Jan 2020 04:01:04 PM EST
audit-libs-3.0-0.10.20180831git0047a6c.el8.x86_64 Mon 27 Jan 2020 04:01:04 PM EST
sqlite-libs-3.26.0-3.el8.x86_64              Mon 27 Jan 2020 04:01:03 PM EST
libstdc++-8.3.1-4.5.el8.x86_64               Mon 27 Jan 2020 04:01:03 PM EST
libgcrypt-1.8.3-4.el8.x86_64                 Mon 27 Jan 2020 04:01:03 PM EST
libffi-3.1-21.el8.x86_64                     Mon 27 Jan 2020 04:01:03 PM EST
expat-2.2.5-3.el8.x86_64                     Mon 27 Jan 2020 04:01:03 PM EST
elfutils-libelf-0.176-5.el8.x86_64           Mon 27 Jan 2020 04:01:03 PM EST
chkconfig-1.11-1.el8.x86_64                  Mon 27 Jan 2020 04:01:03 PM EST
readline-7.0-10.el8.x86_64                   Mon 27 Jan 2020 04:01:02 PM EST
libxml2-2.9.7-5.el8.x86_64                   Mon 27 Jan 2020 04:01:02 PM EST
libxcrypt-4.1.1-4.el8.x86_64                 Mon 27 Jan 2020 04:01:02 PM EST
libuuid-2.32.1-17.el8.x86_64                 Mon 27 Jan 2020 04:01:02 PM EST
libgpg-error-1.31-1.el8.x86_64               Mon 27 Jan 2020 04:01:02 PM EST
libcap-2.26-1.el8.x86_64                     Mon 27 Jan 2020 04:01:02 PM EST
info-6.5-4.el8.x86_64                        Mon 27 Jan 2020 04:01:02 PM EST
zlib-1.2.11-10.el8.x86_64                    Mon 27 Jan 2020 04:01:01 PM EST
xz-libs-5.2.4-3.el8.x86_64                   Mon 27 Jan 2020 04:01:01 PM EST
popt-1.16-14.el8.x86_64                      Mon 27 Jan 2020 04:01:01 PM EST
libsepol-2.9-1.el8.x86_64                    Mon 27 Jan 2020 04:01:01 PM EST
libcom_err-1.44.6-3.el8.x86_64               Mon 27 Jan 2020 04:01:01 PM EST
bzip2-libs-1.0.6-26.el8.x86_64               Mon 27 Jan 2020 04:01:01 PM EST
bash-4.4.19-10.el8.x86_64                    Mon 27 Jan 2020 04:01:01 PM EST
glibc-2.28-72.el8.x86_64                     Mon 27 Jan 2020 04:01:00 PM EST
glibc-langpack-en-2.28-72.el8.x86_64         Mon 27 Jan 2020 04:00:59 PM EST
glibc-common-2.28-72.el8.x86_64              Mon 27 Jan 2020 04:00:59 PM EST
xkeyboard-config-2.24-3.el8.noarch           Mon 27 Jan 2020 04:00:58 PM EST
publicsuffix-list-dafsa-20180723-1.el8.noarch Mon 27 Jan 2020 04:00:58 PM EST
pcre2-10.32-1.el8.x86_64                     Mon 27 Jan 2020 04:00:58 PM EST
ncurses-libs-6.1-7.20180224.el8.x86_64       Mon 27 Jan 2020 04:00:58 PM EST
ncurses-base-6.1-7.20180224.el8.noarch       Mon 27 Jan 2020 04:00:58 PM EST
libselinux-2.9-2.1.el8.x86_64                Mon 27 Jan 2020 04:00:58 PM EST
filesystem-3.8-2.el8.x86_64                  Mon 27 Jan 2020 04:00:57 PM EST
efi-filesystem-3-2.el8.noarch                Mon 27 Jan 2020 04:00:57 PM EST
basesystem-11-5.el8.noarch                   Mon 27 Jan 2020 04:00:57 PM EST
centos-repos-8.1-1.1911.0.8.el8.x86_64       Mon 27 Jan 2020 04:00:56 PM EST
centos-release-8.1-1.1911.0.8.el8.x86_64     Mon 27 Jan 2020 04:00:56 PM EST
tzdata-2019c-1.el8.noarch                    Mon 27 Jan 2020 04:00:55 PM EST
python3-setuptools-wheel-39.2.0-5.el8.noarch Mon 27 Jan 2020 04:00:55 PM EST
python3-pip-wheel-9.0.3-15.el8.noarch        Mon 27 Jan 2020 04:00:55 PM EST
centos-gpg-keys-8.1-1.1911.0.8.el8.noarch    Mon 27 Jan 2020 04:00:55 PM EST
geolite2-city-20180605-1.el8.noarch          Mon 27 Jan 2020 04:00:53 PM EST
libgcc-8.3.1-4.5.el8.x86_64                  Mon 27 Jan 2020 04:00:49 PM EST
geolite2-country-20180605-1.el8.noarch       Mon 27 Jan 2020 04:00:49 PM EST
[root@\ U0264476bin]$
```

Añadimos |wc -l para obtener el número de paquetes.

```
[root@\ U0264476bin]$ rpm -qa -last |wc -l
420
[root@\ U0264476bin]$
```

2.2) Comprueba qué paquetes están sin actualizar (no los actualices)

```
[root@\ U0264476bin]$ yum check-update
Last metadata expiration check: 0:06:06 ago on Sun 23 Feb 2020 01:07:30 PM EST.

epel-release.noarch                                 8-8.el8                                 epel
glibc.x86_64                                        2.28-72.el8_1.1                         BaseOS
glibc-common.x86_64                                 2.28-72.el8_1.1                         BaseOS
glibc-langpack-en.x86_64                            2.28-72.el8_1.1                         BaseOS
grub2-common.noarch                                 1:2.02-78.el8_1.1                       BaseOS
grub2-efi-x64.x86_64                                1:2.02-78.el8_1.1                       BaseOS
grub2-pc.x86_64                                     1:2.02-78.el8_1.1                       BaseOS
grub2-pc-modules.noarch                             1:2.02-78.el8_1.1                       BaseOS
grub2-tools.x86_64                                  1:2.02-78.el8_1.1                       BaseOS
grub2-tools-extra.x86_64                            1:2.02-78.el8_1.1                       BaseOS
grub2-tools-minimal.x86_64                          1:2.02-78.el8_1.1                       BaseOS
kernel.x86_64                                       4.18.0-147.5.1.el8_1                    BaseOS
kernel-core.x86_64                                  4.18.0-147.5.1.el8_1                    BaseOS
kernel-modules.x86_64                               4.18.0-147.5.1.el8_1                    BaseOS
kernel-tools.x86_64                                 4.18.0-147.5.1.el8_1                    BaseOS
kernel-tools-libs.x86_64                            4.18.0-147.5.1.el8_1                    BaseOS
libarchive.x86_64                                   3.3.2-8.el8_1                           BaseOS
openldap.x86_64                                     2.4.46-11.el8_1                         BaseOS
openssh.x86_64                                      8.0p1-4.el8_1                           BaseOS
openssh-clients.x86_64                              8.0p1-4.el8_1                           BaseOS
openssh-server.x86_64                               8.0p1-4.el8_1                           BaseOS
policycoreutils.x86_64                              2.9-3.el8_1.1                           BaseOS
python3-perf.x86_64                                 4.18.0-147.5.1.el8_1                    BaseOS
sqlite-libs.x86_64                                  3.26.0-4.el8_1                          BaseOS
systemd.x86_64                                      239-18.el8_1.2                          BaseOS
systemd-libs.x86_64                                 239-18.el8_1.2                          BaseOS
systemd-pam.x86_64                                  239-18.el8_1.2                          BaseOS
systemd-udev.x86_64                                 239-18.el8_1.2                          BaseOS
tuned.noarch                                        2.12.0-3.el8_1.1                        BaseOS
Obsoleting Packages
grub2-tools.x86_64                                  1:2.02-78.el8_1.1                       BaseOS
   grub2-tools.x86_64                               1:2.02-78.el8                           @anaconda
grub2-tools-efi.x86_64                              1:2.02-78.el8_1.1                       BaseOS
   grub2-tools.x86_64                               1:2.02-78.el8                           @anaconda
grub2-tools-extra.x86_64                            1:2.02-78.el8_1.1                       BaseOS
   grub2-tools.x86_64                               1:2.02-78.el8                           @anaconda
grub2-tools-minimal.x86_64                          1:2.02-78.el8_1.1                       BaseOS
   grub2-tools.x86_64                               1:2.02-78.el8                           @anaconda
[root@\ U0264476bin]$
```

2.3) Instala el paquete emacs

```
jasper-libs-2.0.14-4.el8.x86_64                     jbigkit-libs-2.1-14.el8.x86_64
lcms2-2.9-2.el8.x86_64                              libICE-1.0.9-15.el8.x86_64
libSM-1.2.3-1.el8.x86_64                            libX11-1.6.7-1.el8.x86_64
libX11-common-1.6.7-1.el8.noarch                    libX11-xcb-1.6.7-1.el8.x86_64
libXau-1.0.8-13.el8.x86_64                          libXaw-1.0.13-10.el8.x86_64
libXcomposite-0.4.4-14.el8.x86_64                   libXcursor-1.1.15-3.el8.x86_64
libXdamage-1.1.4-14.el8.x86_64                      libXext-1.3.3-9.el8.x86_64
libXfixes-5.0.3-7.el8.x86_64                        libXft-2.3.2-10.el8.x86_64
libXi-1.7.9-7.el8.x86_64                            libXinerama-1.1.4-1.el8.x86_64
libXmu-1.1.2-12.el8.x86_64                          libXpm-3.5.12-7.el8.x86_64
libXrandr-1.5.1-7.el8.x86_64                        libXrender-0.9.10-7.el8.x86_64
libXt-1.1.5-12.el8.x86_64                           libXtst-1.2.3-7.el8.x86_64
libXv-1.0.11-7.el8.x86_64                           libXxf86vm-1.1.4-9.el8.x86_64
libdatrie-0.2.9-7.el8.x86_64                        libdrm-2.4.98-2.el8.x86_64
libepoxy-1.5.2-1.el8.x86_64                         libglvnd-1:1.0.1-0.9.git5baa1e5.el8.x86_64
libglvnd-egl-1:1.0.1-0.9.git5baa1e5.el8.x86_64      libglvnd-gles-1:1.0.1-0.9.git5baa1e5.el8.x86_64
libglvnd-glx-1:1.0.1-0.9.git5baa1e5.el8.x86_64      libjpeg-turbo-1.5.3-10.el8.x86_64
liblockfile-1.14-1.el8.x86_64                       libnotify-0.7.7-5.el8.x86_64
libogg-2:1.3.2-10.el8.x86_64                        libotf-0.9.13-11.el8.x86_64
librsvg2-2.42.7-3.el8.x86_64                        libthai-0.1.27-2.el8.x86_64
libtheora-1:1.1.1-21.el8.x86_64                     libtiff-4.0.9-15.el8.x86_64
libvisual-1:0.4.0-24.el8.x86_64                     libvorbis-1:1.3.6-2.el8.x86_64
libwayland-client-1.15.0-1.el8.x86_64               libwayland-cursor-1.15.0-1.el8.x86_64
libwayland-egl-1.15.0-1.el8.x86_64                  libwayland-server-1.15.0-1.el8.x86_64
libwebp-1.0.0-1.el8.x86_64                          libxcb-1.13-5.el8.x86_64
libxshmfence-1.3-2.el8.x86_64                       m17n-db-1.8.0-3.el8.noarch
m17n-lib-1.8.0-2.el8.x86_64                         mesa-libEGL-19.1.4-3.el8_1.x86_64
mesa-libGL-19.1.4-3.el8_1.x86_64                    mesa-libgbm-19.1.4-3.el8_1.x86_64
mesa-libglapi-19.1.4-3.el8_1.x86_64                 openjpeg2-2.3.1-2.el8_1.x86_64
opus-1.3-0.4.beta.el8.x86_64                        orc-0.4.28-2.el8.x86_64
pango-1.42.4-6.el8.x86_64                           pixman-0.36.0-1.el8.x86_64
rest-0.8.1-2.el8.x86_64                             webkit2gtk3-2.24.4-2.el8_1.x86_64
webkit2gtk3-jsc-2.24.4-2.el8_1.x86_64               woff2-1.0.2-4.el8.x86_64
ModemManager-glib-1.10.4-1.el8.x86_64               avahi-glib-0.7-19.el8.x86_64
bubblewrap-0.3.0-1.el8.x86_64                       dejavu-fonts-common-2.35-6.el8.noarch
dejavu-sans-mono-fonts-2.35-6.el8.noarch            emacs-filesystem-1:26.1-5.el8.noarch
fontconfig-2.13.1-3.el8.x86_64                      fontpackages-filesystem-1.44-22.el8.noarch
gdk-pixbuf2-2.36.12-5.el8.x86_64                    glib-networking-2.56.1-1.1.el8.x86_64
gsettings-desktop-schemas-3.32.0-3.el8.x86_64       json-glib-1.4.4-1.el8.x86_64
libgusb-0.3.0-1.el8.x86_64                          libmodman-2.0.1-17.el8.x86_64
libpciaccess-0.14-1.el8.x86_64                      libpkgconf-1.4.2-1.el8.x86_64
libproxy-0.4.15-5.2.el8.x86_64                      libsoup-2.62.3-1.el8.x86_64
libxslt-1.1.32-3.el8.x86_64                         pkgconf-1.4.2-1.el8.x86_64
pkgconf-m4-1.4.2-1.el8.noarch                       pkgconf-pkg-config-1.4.2-1.el8.x86_64
xml-common-0.6.3-50.el8.noarch

Complete!
[root@\ U0264476bin]$ yum install emacs
```

3) Opciones del kernel. Mostrar la versión del kernel

3.1) Encuentra una orden para mostrar en pantalla la versión de kernel.

Usamos el comando uname -a para saber toda la información sobre el kernel.

```
[root@localhost ~]# uname -a
Linux localhost.localdomain 3.10.0-957.5.1.el7.x86_64 #1 SMP Fri Feb 1 14:54:57
UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
[root@localhost ~]# _
```

4) Mensaje de presentación /etc/motd, /etc/issue

4.1) Descubre la función de los ficheros */etc/motd* y */etc/issue*

Los ficheros /etc/motd y /etc/issue contienen los mensajes de bienvenida al arrancar la máquina Linux. Si los modificamos, se cambia el saludo.

```
[root@\ UO264476bin]$ cat /etc/motd
HOLA MUNDO
[root@\ UO264476bin]$ cat /etc/issue
\S
Kernel \r on an \m

[root@\ UO264476bin]$
```

Practica 2

Empezamos creando un disco duro de 8GB en la máquina.

Ahora con el comando parted /dev/sda unit MB print free y con lsblk -f comprobamos las particiones actuales del disco principal.

```
U0264476@~ $ >parted /dev/sda unit MB print free
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sda: 42950MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End      Size     Type     File system  Flags
        0.00MB  1.05MB   1.05MB             Free Space
 1      1.05MB  1075MB   1074MB   primary  ext4         boot
 2      1075MB  42950MB  41875MB  primary               lvm

U0264476@~ $ >
```

```
U0264476@~ $ >lsblk -f
NAME        FSTYPE      LABEL UUID                                   MOUNTPOINT
sda
├─sda1      ext4              65d08fe1-1415-4c54-b3b9-0d3a32d63b77   /boot
└─sda2      LVM2_member       giZrxY-7FnP-ngSb-EKKy-AzVy-Pcdc-mAKATw
  ├─cl-root xfs               adfd9fd1-d5a4-4f86-b8ff-2e47628e0124   /
  └─cl-swap swap              411f9f03-3782-44d8-9809-082a8db5883e   [SWAP]
sdb
sr0
U0264476@~ $ >
```

Dispositivos de almacenamiento: ¿Cuál es el nombre del fichero de dispositivo del nuevo disco?
El nombre del nuevo disco creado anteriormente es sdb.

A. Particionamiento: Uso de gdisk*:*

Instala gdisk con dnf

```
U02644760@~ $ >dnf install gdisk
Last metadata expiration check: 1 day, 22:53:13 ago on Tue 11 Feb 2020 06:27:36 PM GMT.
Dependencies resolved.
================================================================================
 Package             Architecture         Version                  Repository            Size
================================================================================
Installing:
 gdisk               x86_64               1.0.3-6.el8              BaseOS               240 k

Transaction Summary
================================================================================
Install  1 Package

Total download size: 240 k
Installed size: 873 k
Is this ok [y/N]: y
Downloading Packages:
gdisk-1.0.3-6.el8.x86_64.rpm                                 129 kB/s | 240 kB     00:01
--------------------------------------------------------------------------------
Total                                                         99 kB/s | 240 kB     00:02
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Installing       : gdisk-1.0.3-6.el8.x86_64                               1/1
  Running scriptlet: gdisk-1.0.3-6.el8.x86_64                               1/1
  Verifying        : gdisk-1.0.3-6.el8.x86_64                               1/1

Installed:
  gdisk-1.0.3-6.el8.x86_64

Complete!
```

Haz gdisk /dev/XXX (/dev/XXX es el nombre del segundo dispositivo).

```
U02644760@~ $ >gdisk /dev/sdb
GPT fdisk (gdisk) version 1.0.3

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries.

Command (? for help): _
```

Usa el comando 'p' para ver la tabla de particiones

Usamos el comando p para consultar la tabla de particiones.

```
Command (? for help): p
Disk /dev/sdb: 16777216 sectors, 8.0 GiB
Model: VBOX HARDDISK
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 5833430C-7012-48FC-AEF7-0A01D337D4FD
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 16777182
Partitions will be aligned on 2048-sector boundaries
Total free space is 16777149 sectors (8.0 GiB)

Number  Start (sector)    End (sector)  Size       Code  Name

Command (? for help): _
```

Usa el comando 'n' para crear una partición de 512MiB de tipo Linux filesystem

Creamos la partición de 512 MiB con el numero 1 y de tipo Linux filesystem

```
Command (? for help): n
Partition number (1-128, default 1): 1
First sector (34-16777182, default = 2048) or {+-}size{KMGTP}:
Last sector (2048-16777182, default = 16777182) or {+-}size{KMGTP}: +512MiB
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'

Command (? for help): _
```

Crea una nueva partición de 3GiB

Creamos la partición de 3GiB con el numero 2 y de tipo Linux filesystem

```
Command (? for help): n
Partition number (2-128, default 2):
First sector (34-16777182, default = 1050624) or {+-}size{KMGTP}:
Last sector (1050624-16777182, default = 16777182) or {+-}size{KMGTP}: +3GiB
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'

Command (? for help): _
```

Y una tercera con el resto del espacio disponible de tipo Microsoft Basic data

```
Command (? for help): n
Partition number (3-128, default 3):
First sector (34-16777182, default = 7342080) or {+-}size{KMGTP}:
Last sector (7342080-16777182, default = 16777182) or {+-}size{KMGTP}:
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300): _
```

Buscamos el código de Microsoft Basic Data, con el comando L en gdisk, en este caso 0700.

```
Hex code or GUID (L to show codes, Enter = 8300): L
0700 Microsoft basic data  0c01 Microsoft reserved    2700 Windows RE
3000 ONIE boot             3001 ONIE config           3900 Plan 9
4100 PowerPC PReP boot     4200 Windows LDM data      4201 Windows LDM metadata
4202 Windows Storage Spac  7501 IBM GPFS              7f00 ChromeOS kernel
7f01 ChromeOS root         7f02 ChromeOS reserved     8200 Linux swap
8300 Linux filesystem      8301 Linux reserved        8302 Linux /home
8303 Linux x86 root (/)    8304 Linux x86-64 root (/  8305 Linux ARM64 root (/)
8306 Linux /srv            8307 Linux ARM32 root (/)  8400 Intel Rapid Start
8e00 Linux LVM             a000 Android bootloader    a001 Android bootloader 2
a002 Android boot          a003 Android recovery      a004 Android misc
a005 Android metadata      a006 Android system        a007 Android cache
a008 Android data          a009 Android persistent    a00a Android factory
a00b Android fastboot/ter  a00c Android OEM           a500 FreeBSD disklabel
a501 FreeBSD boot          a502 FreeBSD swap          a503 FreeBSD UFS
a504 FreeBSD ZFS           a505 FreeBSD Vinum/RAID    a580 Midnight BSD data
```

```
fb01 VMware reserved       fc00 VMware kcore crash p  fd00 Linux
Hex code or GUID (L to show codes, Enter = 8300): 0700
Changed type of partition to 'Microsoft basic data'

Command (? for help): _
```

Graba las particiones a disco con el comando 'w'.

Usamos el comando w para grabar las particiones

```
Command (? for help): w

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTIN
PARTITIONS!!

Do you want to proceed? (Y/N): y
OK; writing new GUID partition table (GPT) to /dev/sdb.
The operation has completed successfully.
UO2644760~ $ >_
```

Usa el comando parted /dev/XXX unit MB print y anota el resultado

Después de usar el comando observamos como salen las particiones hechas con gdisk en el disco nuevo.

```
U02644760~ $ >parted /dev/sdb unit MB print
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sdb: 8590MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name                  Flags
 1      1.05MB  538MB   537MB                Linux filesystem
 2      538MB   3759MB  3221MB               Linux filesystem
 3      3759MB  8590MB  4831MB               Microsoft basic data  msftdata

U02644760~ $ >
```

Haz mkfs /dev/XXX1 (/dev/XXX es el nombre del dispositivo, 1 es el número de la primera partición) para crear un sistema de archivos de tipo ext2 en la partición de 512Mb del disco.

```
U02644760~ $ >mkfs.ext2 /dev/sdb1
mke2fs 1.44.6 (5-Mar-2019)
Creating filesystem with 131072 4k blocks and 32768 inodes
Filesystem UUID: dfb193d0-3d9b-4203-b72c-93cc1590b901
Superblock backups stored on blocks:
        32768, 98304

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

U02644760~ $ >_
```

Haz e2label /dev/XXX1 disco2a para darle una etiqueta al disco

```
U02644760~ $ >e2label /dev/sdb1 disco2a
```

Haz tune2fs -j /dev/XXX1 para crear el archivo de journal y hacer que el filesystem sea de tipo ext3

```
U02644760~ $ >tune2fs -j /dev/sdb1
tune2fs 1.44.6 (5-Mar-2019)
Creating journal inode: done
U02644760~ $ >
```

Crea un sistema de archivos xfs en la segunda partición. Ponle como etiqueta disco2b con el comando xfs_admin

```
U0264476@~ $ >mkfs.xfs /dev/sdb2
meta-data=/dev/sdb2              isize=512    agcount=4, agsize=196608 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1
data     =                       bsize=4096   blocks=786432, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
U0264476@~ $ >
```

```
U0264476@~ $ >xfs_admin -L disco2b /dev/sdb2
writing all SBs
new label = "disco2b"
U0264476@~ $ >
```

Crea un sistema de archivos fat32 en la tercera partición del disco. Usa fatlabel para etiquetar la partición con DISCO2C

Repetimos el mismo proceso que en los anteriores y finalmente tenemos estas particiones

```
[root@\ U0264476~]$ parted /dev/sdb unit MB print
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sdb: 8590MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name                  Flags
 1      1.05MB  538MB   537MB   ext3         Linux filesystem
 2      538MB   3759MB  3221MB  xfs          Linux filesystem
 3      3759MB  8590MB  4831MB  fat32        Microsoft basic data  msftdata

[root@\ U0264476~]$
```

Haz mkdir /mnt/prueba para crear un punto de montaje

```
U0264476@~ $ >mkdir /mnt/prueba
U0264476@~ $ >
```

Haz mount /dev/XXX1 /mnt/prueba para montar el nuevo filesystem en el árbol de directorios

```
UO264476@~ $ >mkdir /mnt/prueba
UO264476@~ $ >mount /dev/sdb1 /mnt/prueba
UO264476@~ $ >cd /mnt/prueba
UO264476@prueba $ >mkdir dir1
UO264476@prueba $ >ls -la
total 24
drwxr-xr-x. 4 root root  4096 Feb 13 17:30 .
drwxr-xr-x. 3 root root    20 Feb 13 17:29 ..
drwxr-xr-x. 2 root root  4096 Feb 13 17:30 dir1
drwx------. 2 root root 16384 Feb 13 17:25 lost+found
UO264476@prueba $ >_
```

Crea algún archivo en el directorio /mnt/prueba

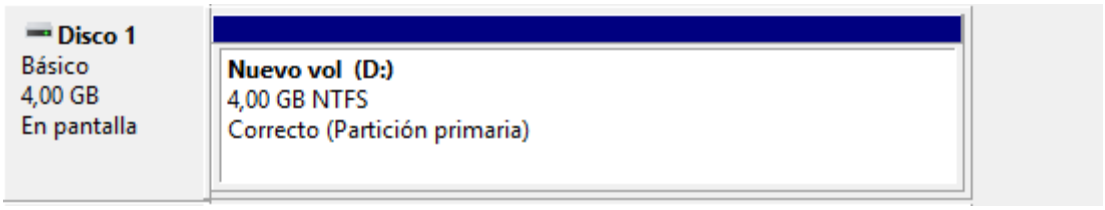
```
UO264476@prueba $ >mount /dev/sdb2 /mnt/prueba
UO264476@prueba $ >cd /mnt/prueba
UO264476@prueba $ >mkdir dir2
UO264476@prueba $ >
```

Repite el proceso con los restantes filesystems usando otros puntos de montaje. Cuando hayas terminado, ejecuta lsblk -f y anota el resultado.

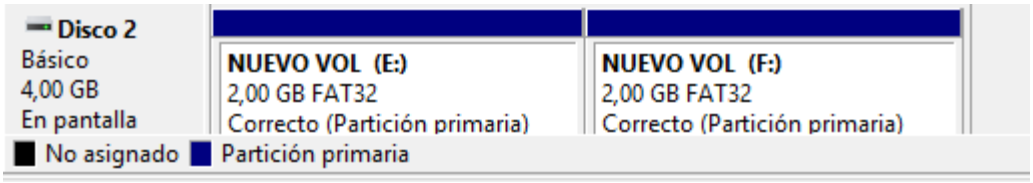
```
[root@\ UO264476~]$ lsblk -f
NAME        FSTYPE      LABEL   UUID                                   MOUNTPOINT
sda
├─sda1      vfat                F64E-CE13                              /boot/efi
├─sda2      ext4                e3fc24a7-7dda-4a46-b504-f5f8bff351bc   /boot
└─sda3      LVM2_member         dzWDb7-AT8N-le7L-EE62-qFx5-NNXn-0MhS8t
  ├─cl-root xfs                 b0d65fd0-9e74-4ab8-8db6-37de1cff28df   /
  └─cl-swap swap                155084c4-3e7b-495f-adc1-1924d02ae34b   [SWAP]
sdb
├─sdb1      ext3        disco2a 3af0b81f-da8e-49b7-a5dd-df99476b4643
├─sdb2      xfs         disco2b 3b84c80f-b433-454b-9c2d-7a83ab56bc4b
└─sdb3      vfat        DISCO2C 91FB-4203
sdc
sdd
sr0
[root@\ UO264476~]$
```

Adición de nuevos discos a un sistema Windows ya instalado

Con el primer disco, haz un único volumen simple y formatéalo como NTFS



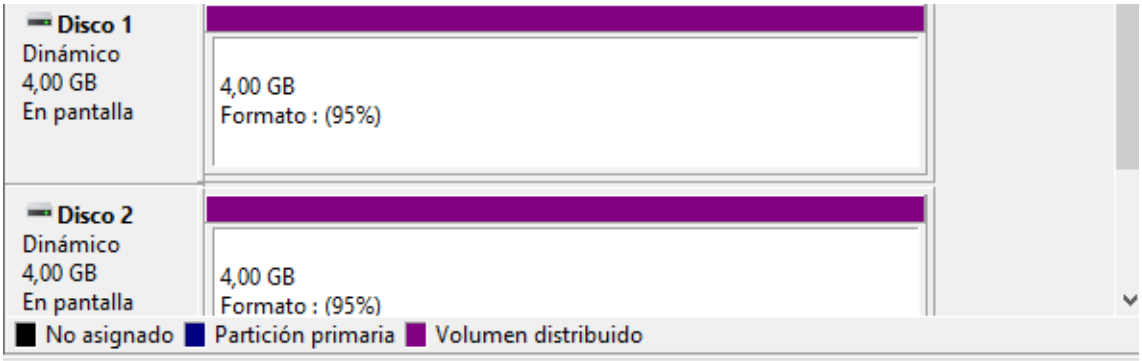
Con el segundo disco, haz dos particiones de igual tamaño. Formatea ambas como FAT32. Captura la pantalla.



Elimina la partición del disco 1 y las dos particiones del disco 2, dejando todo el espacio como no asignado

Eliminamos ambos discos

Une ambos discos con volúmenes NTFS distribuidos, de forma que exista una única unidad E. ¿Qué tamaño tiene la nueva unidad?



Elimina el volumen distribuido y une de nuevo ambos discos con volúmenes distribuidos NTFS en un volumen reflejado (RAID1). ¿Qué tamaño tiene la nueva unidad?

Para que ésta reconozca la partición NFTS debemos instalar el driver ntfs-3g que se encuentra en el repositorio epel.

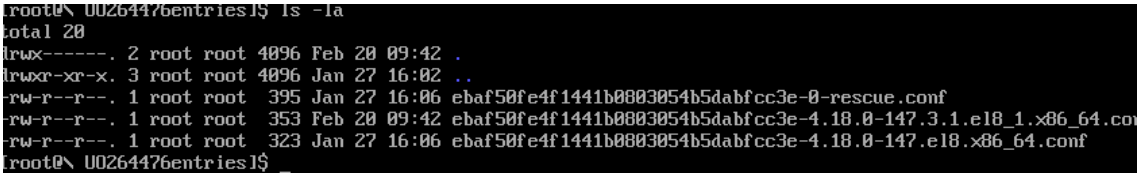
```
[root@\ U0264476~]$ dnf install -y epelrelease
CentOS-8 - AppStream                                            18 kB/s | 4.3 kB     00:00
^CCentOS-8 - AppStream                                          2.5 MB/s | 3.6 MB     00:01

Failed to download metadata for repo 'AppStream'
Error: Failed to download metadata for repo 'AppStream'
[root@\ U0264476~]$ dnf install -y epel-release
CentOS-8 - AppStream                                            8.7 kB/s | 4.3 kB     00:00
CentOS-8 - AppStream                                            3.3 MB/s | 6.5 MB     00:01
CentOS-8 - Base                                                  15 kB/s | 3.8 kB     00:00
CentOS-8 - Base                                                 3.9 MB/s | 5.0 MB     00:01
CentOS-8 - Extras                                               5.3 kB/s | 1.5 kB     00:00
Extra Packages for Enterprise Linux 8 - x86_64                   43 kB/s |  34 kB     00:00
Extra Packages for Enterprise Linux 8 - x86_64                  3.5 MB/s | 5.9 MB     00:01
Package epel-release-8-5.el8.noarch is already installed.
Dependencies resolved.
================================================================================================
 Package                  Architecture            Version                 Repository          Size
================================================================================================
Upgrading:
 epel-release             noarch                  8-8.el8                 epel                22 k

Transaction Summary
================================================================================================
Upgrade  1 Package

Total download size: 22 k
Downloading Packages:
epel-release-8-8.el8.noarch.rpm                                  61 kB/s |  22 kB     00:00
------------------------------------------------------------------------------------------------
Total                                                            19 kB/s |  22 kB     00:01
Running transaction check
```

```
[root@\ U0264476~]$ dnf install -y ntfs-3g
Extra Packages for Enterprise Linux Modular 8 - x86_64
Package ntfs-3g-2:2017.3.23-11.el8.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@\ U0264476~]$
```

Practica 3

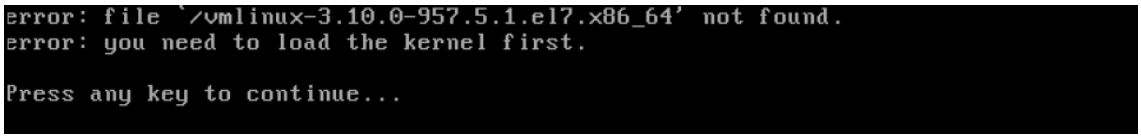
A. Recuperación básica de errores durante el inicio

Arranca CentOS normalmente (con EFI activado y sin el disco de instalación) y examina el contenido de /boot/loader/entries. Cada fichero de ese directorio es una de las opciones de arranque que nos aparecen al inicio. Corresponden a varios kernel y al de rescate.



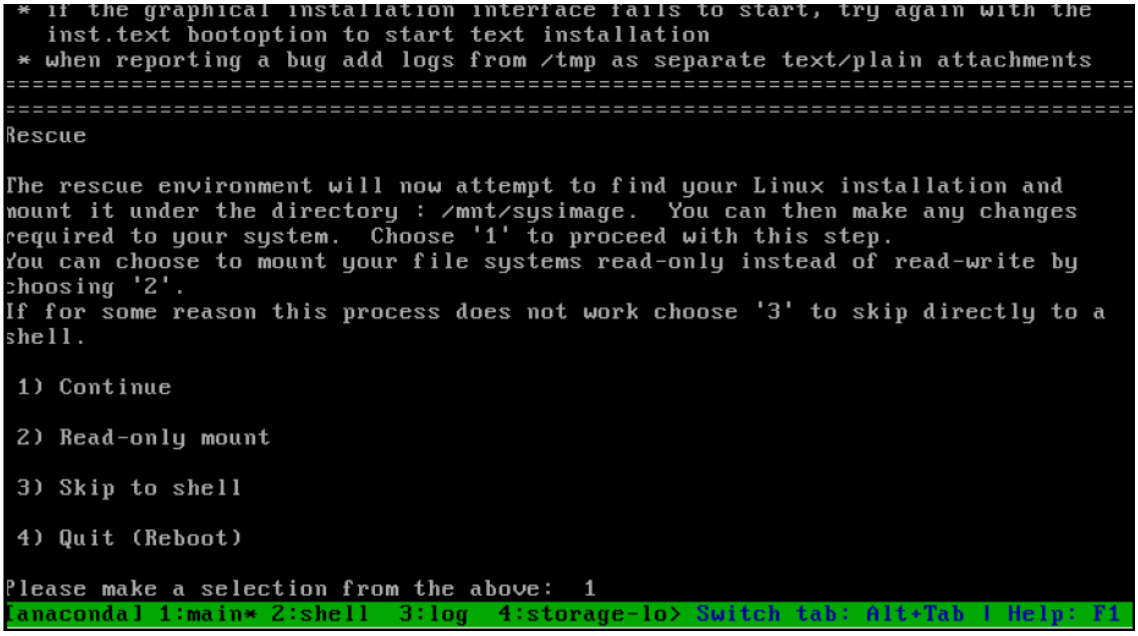
Editamos el fichero de en medio, el más reciente

Modificamos la x en vmlinux por vmlinuz y arrancamos la máquina de nuevo, y vemos como no reconoce el kernel.

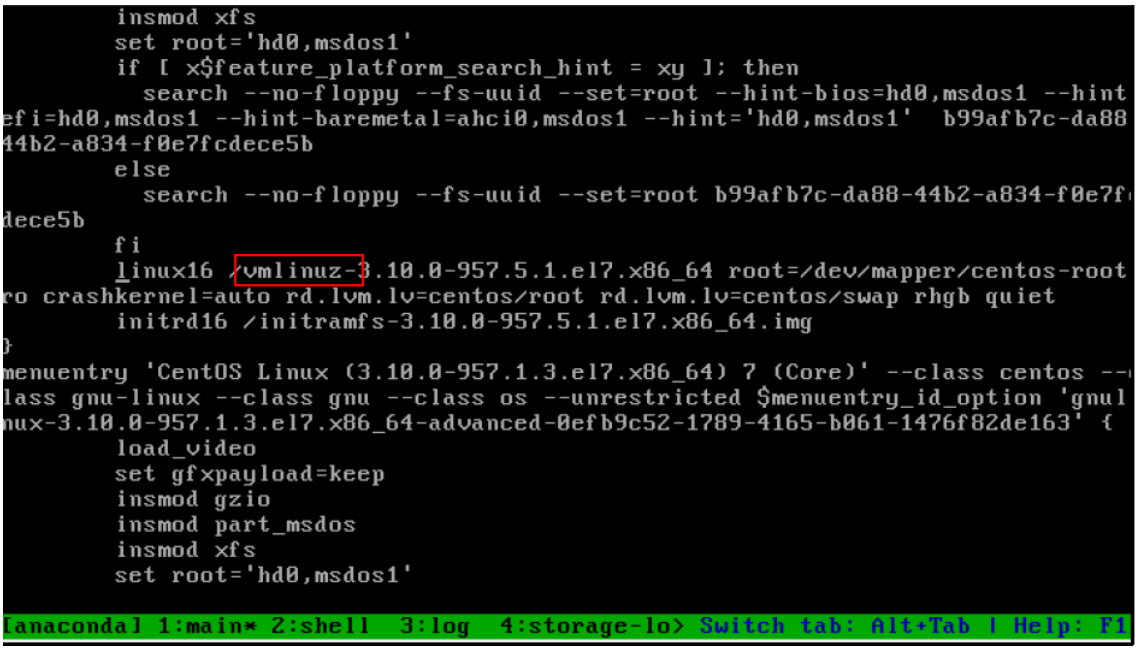


Bota en modo de recuperación y monta el disco con el sistema defectuoso, haz chroot a /mnt/sysimage, carga el teclado español (loadkeys es) y edita y corrige el fichero del punto anterior. Quita el disco de arranque, activa EFI, rebota y comprueba que el problema está solucionado. Ignora los mensajes de reetiquetado de SELinux.

Ejecutamos el Troubleshooting -> Rescue a CentOS system en el menú de instalación.



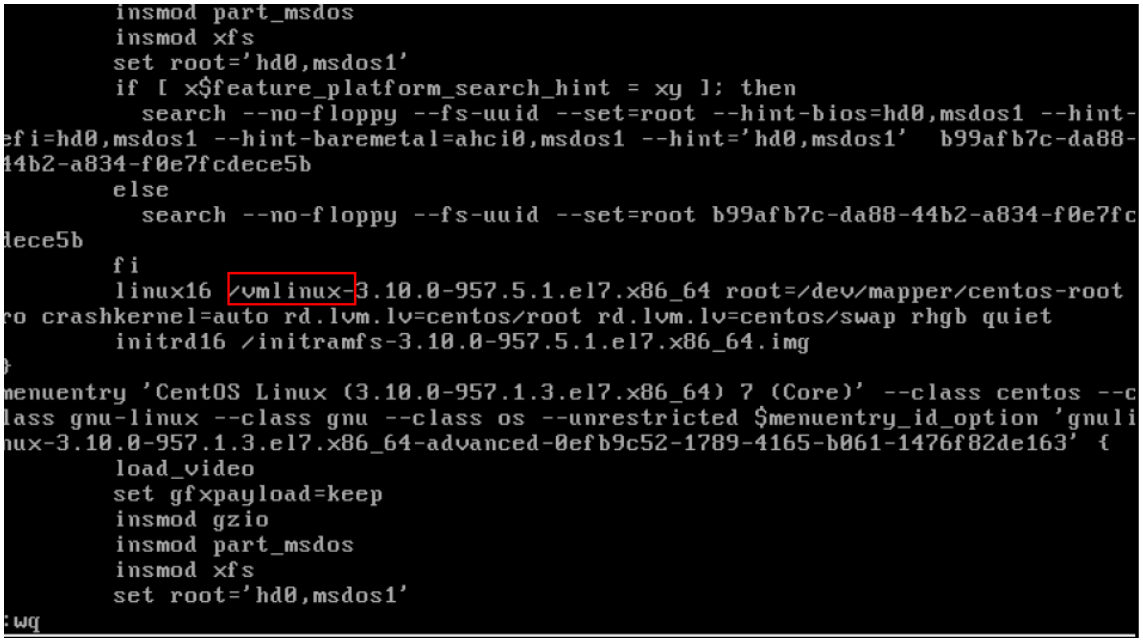
Accedemos al fichero /mnt/sysimage/boot/grub.cfg y volvemos a hacer el cambio.



Y la maquina funciona de igual manera.

Vuelve a repetir lo del punto primero y cambia de nuevo vmlinuz por vmlinux. Reinicia y desde la pantalla de arranque (sin usar el disco de instalación) modifica el nombre del kernel en la entrada del menú de forma que el servidor arranque

Deshacemos el cambio anterior y hacemos el mismo cambio en el fichero de arranque y la maquina continúa funcionando correctamente.

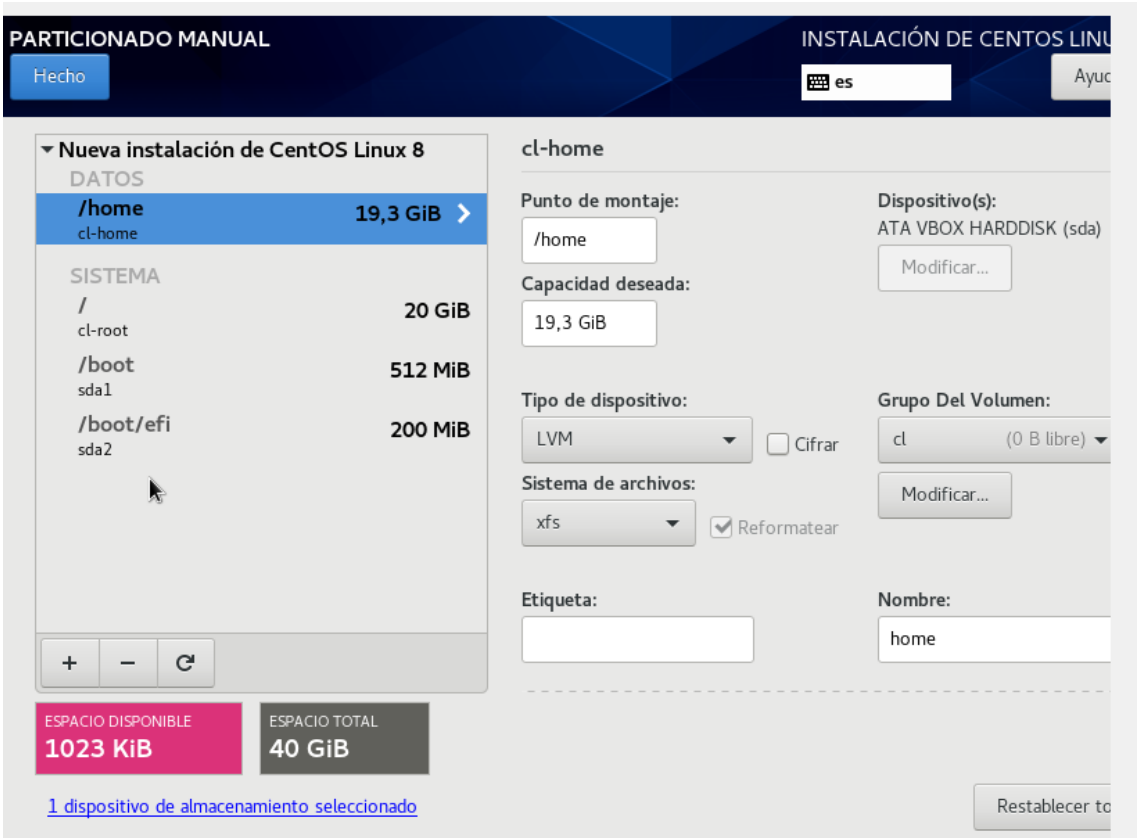


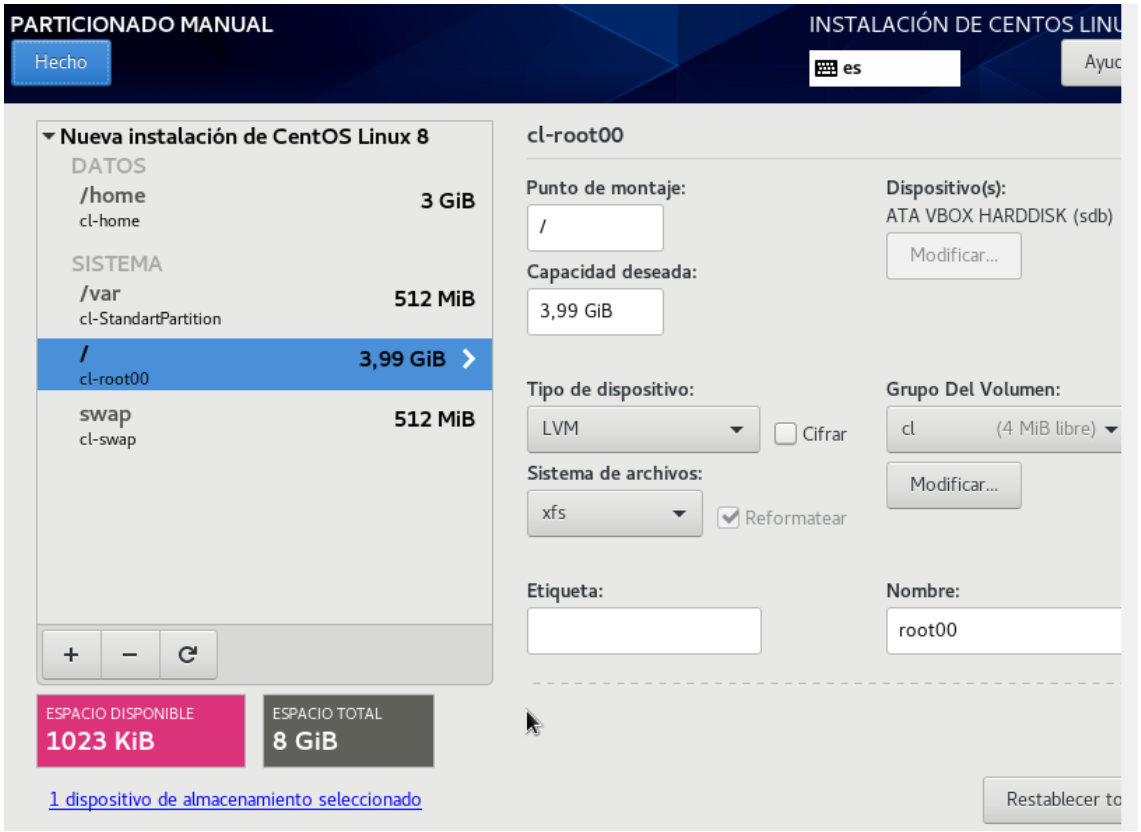
Vuelve a examinar otra vez el fichero que modificaste. ¿Es correcto o sigue conteniendo la palabra "vmlinux"? ¿Por qué?

El fichero de arranque es un fichero temporal, si examinamos el grub.cfg vemos que no se ha modificado el kerner del sistema, para que este cambio sea permanente debería hacerse en el grub.cfg del kernel no en el fichero de arranque.

B. Instalación de Linux con particionamiento dinámico

Creamos el sistema con los parámetros de la practica





Anota en el documento el resultado final, o haz una copia de la pantalla de programa de instalación. A continuación, instala el operativo e inicia sesión, para comprobar que todo es correcto. Haz df y copia la pantalla.

```
[U0264476@~]df
S.ficheros          bloques de 1K  Usados Disponibles Uso% Montado en
devtmpfs                   919172       0      919172   0% /dev
tmpfs                      936776       0      936776   0% /dev/shm
tmpfs                      936776    8636      928140   1% /run
tmpfs                      936776       0      936776   0% /sys/fs/cgroup
/dev/mapper/cl-root      20961280 1434220    19527060   7% /
/dev/mapper/cl-home      20228096  174144    20053952   1% /home
/dev/sda1                  499656  136552      326408  30% /boot
/dev/sda2                  204580       8      204572   1% /boot/efi
tmpfs                      187352       0      187352   0% /run/user/0
[U0264476@~]
```

Elimina con gdisk las particiones del tercer disco si las hubiere

Ejecutamos el comando p en gdisk y vemos que no hay particiones.

```
GPT fdisk (gdisk) version 1.0.3

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries.

Command (? for help): p
Disk /dev/sdc: 16777216 sectors, 8.0 GiB
Model: VBOX HARDDISK
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): ADD38987-66B0-4A0E-83AC-B3B99AE99BCB
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 16777182
Partitions will be aligned on 2048-sector boundaries
Total free space is 16777149 sectors (8.0 GiB)

Number  Start (sector)    End (sector)  Size       Code  Name

Command (? for help): _
```

Crea con gdisk una partición en el tercer disco, que ocupe todo su espacio, y dale el tipo "Linux LVM"

```
a903 NetBSD LFS           a904 NetBSD concatenated   a905 NetBSD encrypted
a906 NetBSD RAID          ab00 Recovery HD           af00 Apple HFS/HFS+
af01 Apple RAID           af02 Apple RAID offline    af03 Apple label
Press the <Enter> key to see more codes:
af04 AppleTV recovery     af05 Apple Core Storage    af06 Apple SoftRAID Statu
af07 Apple SoftRAID Scrat af08 Apple SoftRAID Volum  af09 Apple SoftRAID Cache
6300 QNX6 Power-Safe      bc00 Acronis Secure Zone   be00 Solaris boot
bf00 Solaris root         bf01 Solaris /usr & Mac Z  bf02 Solaris swap
bf03 Solaris backup       bf04 Solaris /var          bf05 Solaris /home
bf06 Solaris alternate se bf07 Solaris Reserved 1    bf08 Solaris Reserved 2
bf09 Solaris Reserved 3   bf0a Solaris Reserved 4    bf0b Solaris Reserved 5
c001 HP-UX data           c002 HP-UX service         e100 ONIE boot
e101 ONIE config          ea00 Freedesktop $BOOT     eb00 Haiku BFS
ed00 Sony system partitio ed01 Lenovo system partit  ef00 EFI System
ef01 MBR partition scheme ef02 BIOS boot partition   f800 Ceph OSD
f801 Ceph dm-crypt OSD    f802 Ceph journal          f803 Ceph dm-crypt journa
f804 Ceph disk in creatio f805 Ceph dm-crypt disk i  fb00 VMWare VMFS
fb01 VMWare reserved      fc00 VMWare kcore crash p  fd00 Linux RAID
Hex code or GUID (L to show codes, Enter = 8300): 8e00
Changed type of partition to 'Linux LVM'

Command (? for help): p
Disk /dev/sdc: 16777216 sectors, 8.0 GiB
Model: VBOX HARDDISK
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): ADD38987-66B0-4A0E-83AC-B3B99AE99BCB
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 16777182
Partitions will be aligned on 2048-sector boundaries
Total free space is 2014 sectors (1007.0 KiB)

Number  Start (sector)    End (sector)  Size       Code  Name
   1            2048        16777182   8.0 GiB     8E00  Linux LVM

Command (? for help):
```

Crea un volumen físico en esa partición, usando la orden pvcreate

```
[UO264476@~]pvcreate  /dev/sdc1
  Physical volume "/dev/sdc1" successfully created.
[UO264476@~]
```

Añade con la orden vgextend el volumen físico al grupo de volúmenes lógicos

```
[UO264476@~]vgextend cl /dev/sdc1
  Volume group "cl" successfully extended
[UO264476@~]
```

Extiende el volumen lógico que contiene a /home para que utilice otros 4GB del tercer disco (usaremos la mitad del nuevo disco que acabamos de incorporar).

```
[UO264476@~]mount /home
[UO264476@~]xfs_growfs
Usage: xfs_growfs [options] mountpoint

Options:
        -d          grow data/metadata section
        -l          grow log section
        -r          grow realtime section
        -n          don't change anything, just show geometry
        -i          convert log from external to internal format
        -t          alternate location for mount table (/etc/mtab)
        -x          convert log from internal to external format
        -D size     grow data/metadata section to size blks
        -L size     grow/shrink log section to size blks
        -R size     grow realtime section to size blks
        -e size     set realtime extent size to size blks
        -m imaxpct  set inode max percent to imaxpct
        -V          print version information
[UO264476@~]xfs_growfs /home
meta-data=/dev/mapper/cl-home    isize=512    agcount=4, agsize=1264896 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1
data     =                       bsize=4096   blocks=5059584, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
[UO264476@~]
```

Anota en el documento las órdenes que has ejecutado y el resultado de ejecutar las órdenes pvscan, lvscan y df /home al terminar el proceso.

```
[UO264476@~]df /home
S.ficheros          bloques de 1K Usados Disponibles Uso% Montado en
/dev/mapper/cl-home      20228096 174144    20053952   1% /home
[UO264476@~]
```

Practica 4

Instala el gdisk si no está ya instalado y crea dos particiones primarias en el segundo disco: la primera de 6GB de tamaño de tipo Linux y la segunda con el espacio restante (2GB) de tipo Linux LVM. Crea un filesystem en la primera partición

```
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'

Command (? for help): n
Partition number (2-128, default 2):
First sector (34-16777182, default = 12584960) or {+-}size{KMGTP}:
Last sector (12584960-16777182, default = 16777182) or {+-}size{KMGTP}: +2GiB
Last sector (12584960-16777182, default = 16777182) or {+-}size{KMGTP}:
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'

Command (? for help): p
Disk /dev/sdb: 16777216 sectors, 8.0 GiB
Model: VBOX HARDDISK
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): A4583DD1-EB3F-405D-BAAC-10E492ED7939
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 16777182
Partitions will be aligned on 2048-sector boundaries
Total free space is 2014 sectors (1007.0 KiB)

Number  Start (sector)    End (sector)  Size       Code  Name
   1            2048        12584959   6.0 GiB     8300  Linux filesystem
   2        12584960        16777182   2.0 GiB     8300  Linux filesystem

Command (? for help): w

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING
PARTITIONS!!

Do you want to proceed? (Y/N): y
OK; writing new GUID partition table (GPT) to /dev/sdb.
The operation has completed successfully.
[UO264476@ ~]_
```

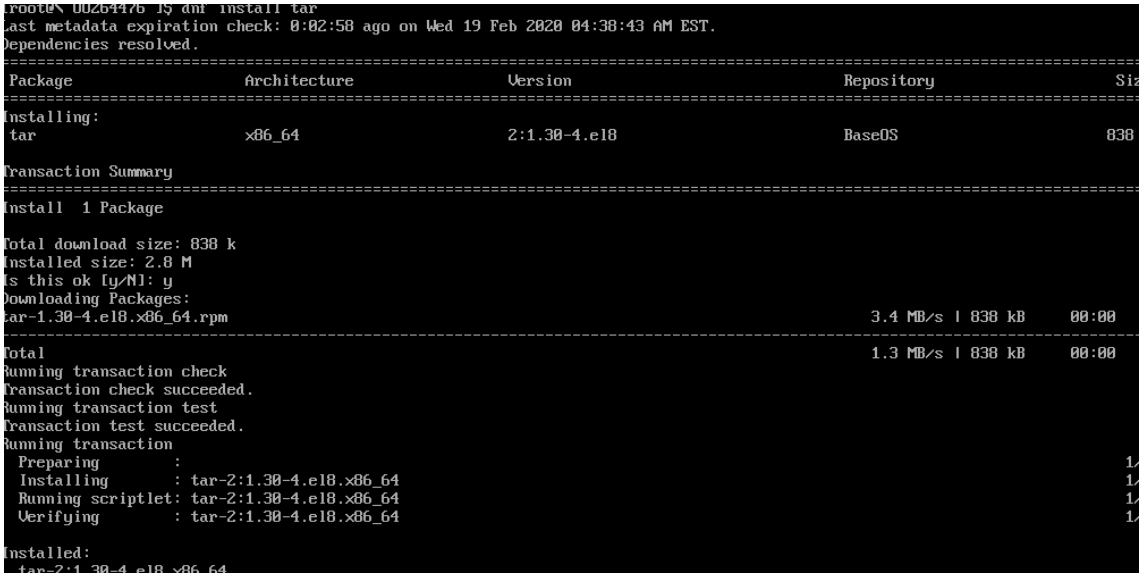
Crea el punto de montaje /mnt/backup y monta la partición a la que acabas de dar formato en /mnt/backup

```
[root@\ UO264476~]$ mkfs /dev/sdb1
mke2fs 1.44.6 (5-Mar-2019)
/dev/sdb1 contains a ext3 file system labelled 'disco2a'
        last mounted on Mon Feb  3 10:13:02 2020
Proceed anyway? (y,N) y
Creating filesystem with 1572864 4k blocks and 393216 inodes
Filesystem UUID: 2ae6b8e8-6d8a-4a8b-9a0c-ad08af07d624
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

[root@\ UO264476~]$ mount /dev/sdb2 /mnt/backup
mount: /mnt/backup: wrong fs type, bad option, bad superblock on /dev/sdb2, missing codepage or helper program, or other
[root@\ UO264476~]$ mount /mnt/backup /dev/sdb2
mount: /dev/sdb2: /mnt/backup is not a block device.
[root@\ UO264476~]$ mount /mnt/backup /dev/sdb1
mount: /dev/sdb1: /mnt/backup is not a block device.
[root@\ UO264476~]$ mount /dev/sdb1 /mnt/backup
[root@\ UO264476~]$
```

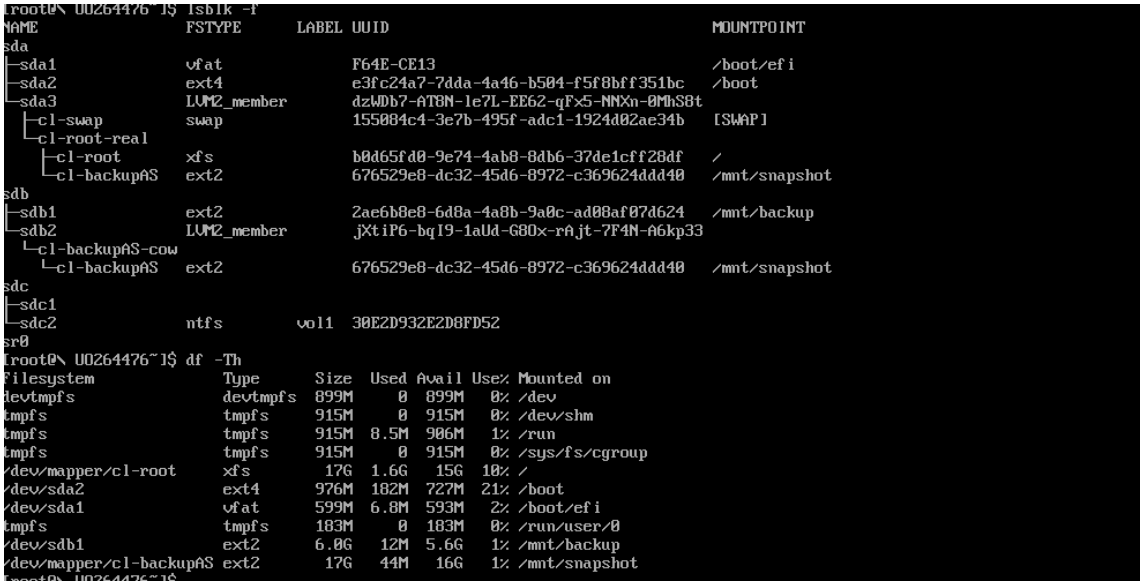
Antes de modificar la estructura del grupo de volúmenes CentOS, guarda el archivo de configuración de LVM (.vg) para restaurarlo después en el nuevo servidor.
Crea un volumen físico en la segunda partición del disco:

```
[root@\ UO264476~]$ cp /etc/lvm/archive/* /mnt/backupç
[root@\ UO264476~]$ cp /etc/lvm/archive/* /mnt/backup
[root@\ UO264476~]$ pvcreate /dev/sdb2
  Physical volume "/dev/sdb2" successfully created.
[root@\ UO264476~]$ vgextend cl /dev/sdb2
  Volume group "cl" successfully extended
[root@\ UO264476~]$ lvcreate -L1000M -s -n backupAS /dev/cl/root
  Logical volume "backupAS" created.
[root@\ UO264476~]$ lvs
  LV       VG Attr       LSize    Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  backupAS cl swi-a-s--- 1000.00m      root   0.01
  root     cl owi-aos---   16.41g
  swap     cl -wi-ao----    2.00g
[root@\ UO264476~]$ mkdir /mnt/snapshot
[root@\ UO264476~]$ mount /dev/cl/backupAS /mnt/snapshot/
mount: /mnt/snapshot: wrong fs type, bad option, bad superblock on /dev/mapper/cl-backupAS, missing codepage or helper
or other error.
[root@\ UO264476~]$ mount /dev/cl/backupAS /mnt/snapshot
mount: /mnt/snapshot: wrong fs type, bad option, bad superblock on /dev/mapper/cl-backupAS, missing codepage or helper
or other error.
[root@\ UO264476~]$ mkfs /dev/cl/backupAS
mke2fs 1.44.6 (5-Mar-2019)
/dev/cl/backupAS contains a xfs file system
Proceed anyway? (y,N) y
Creating filesystem with 4301824 4k blocks and 1077120 inodes
Filesystem UUID: 676529e8-dc32-45d6-8972-c369624ddd40
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

[root@\ UO264476~]$ mount /dev/cl/backupAS /mnt/snapshot
[root@\ UO264476~]$ mount -o nouuid /dev/cl/backupAS /mnt/snapshot/
```

Instala el tar



Captura las salidas de los comandos lsblk –f y df –Th.



Haz un backup de la partición /boot (porque sólo hemos guardado copia del filesystem raíz)

```
root@\ UO264476~]$ tar -cvpzf /mnt/backup/boot.tgz /boot/
tar: Removing leading `/' from member names
/boot/
/boot/initramfs-4.18.0-147.el8.x86_64.img
/boot/initramfs-4.18.0-147.el8.x86_64kdump.img
/boot/initramfs-4.18.0-147.3.1.el8_1.x86_64kdump.img
/boot/.vmlinuz-4.18.0-147.el8.x86_64.hmac
/boot/initramfs-0-rescue-ebaf50fe4f1441b0803054b5dabfcc3e.img
/boot/loader/
/boot/loader/entries/
/boot/loader/entries/ebaf50fe4f1441b0803054b5dabfcc3e-4.18.0-147.el8.x86_64.conf
/boot/loader/entries/ebaf50fe4f1441b0803054b5dabfcc3e-0-rescue.conf
/boot/loader/entries/ebaf50fe4f1441b0803054b5dabfcc3e-4.18.0-147.3.1.el8_1.x86_64.conf
/boot/.vmlinuz-4.18.0-147.3.1.el8_1.x86_64.hmac
/boot/System.map-4.18.0-147.3.1.el8_1.x86_64
/boot/config-4.18.0-147.el8.x86_64
/boot/lost+found/
/boot/vmlinuz-4.18.0-147.3.1.el8_1.x86_64
/boot/efi/
/boot/efi/EFI/
/boot/efi/EFI/BOOT/
/boot/efi/EFI/BOOT/BOOTX64.EFI
/boot/efi/EFI/BOOT/fbx64.efi
/boot/efi/EFI/centos/
/boot/efi/EFI/centos/shimx64-centos.efi
/boot/efi/EFI/centos/BOOTX64.CSV
/boot/efi/EFI/centos/mmx64.efi
/boot/efi/EFI/centos/fonts/
/boot/efi/EFI/centos/grubx64.efi
/boot/efi/EFI/centos/grubenv
/boot/efi/EFI/centos/shimx64.efi
/boot/efi/EFI/centos/grub.cfg
/boot/vmlinuz-0-rescue-ebaf50fe4f1441b0803054b5dabfcc3e
/boot/System.map-4.18.0-147.el8.x86_64
/boot/grub2/
/boot/grub2/grubenv
/boot/initramfs-4.18.0-147.3.1.el8_1.x86_64.img
/boot/config-4.18.0-147.3.1.el8_1.x86_64
/boot/vmlinuz-4.18.0-147.el8.x86_64
root@\ UO264476~]$
```

Finalmente, para guardar el backup, se desmonta el disco sdb2 y después se elimina el snapshot de grupo de volúmenes.

```
[root@\ UO264476~]$ umount /mnt/snapshot
[root@\ UO264476~]$ lvremove /dev/cl/backupAS
Do you really want to remove active logical volume cl/backupAS? [y/n]: y
  Logical volume "backupAS" successfully removed
[root@\ UO264476~]$ vgreduce cl /dev/sdb2
  Removed "/dev/sdb2" from volume group "cl"
[root@\ UO264476~]$ umount /mnt/backup
[root@\ UO264476~]$
```