HowTo LPIC-1

Exam 101-500

Introduction **2**
- Documentation 2
  - LPI Documentation 2
  - PUE: Linux VM & resources 2
  - Fedora / Debian / Ubuntu VM qcow2 Cloud Images 3
  - Training / Exam Dumps Online 4
- LPIC-1 101 Exam topics / objectives 4
- LPIC-1 102 Exam topics / objectives 4

Topic 101: (8) System Architecture **6**
- 101.1 (2) Determine and configure hardware settings 8
- 101.2 (3) Boot the system 9
- 101.3 (3) Change runlevels / boot targets and shutdown or reboot system 10

Topic 102: (12) Linux Installation and Package Management **11**
- 102.1 (2) Design hard disk layout 13
- 102.2 (2) Install a boot manager 14
- 102.3 (1) Manage shared libraries 15
- 102.4 (3) Use Debian package management 16
- 102.5 (3) Use RPM and YUM package management 17
- 102.6 (1) Linux as a virtualization guest 18

Topic 103: (26) GNU and Unix Commands **19**
- 103.1 (4) Work on the command line 23
- 103.2 (2) Process text streams using filters 25
- 103.3 (4) Perform basic file management 27
- 103.4 (4) Use streams, pipes and redirects 31
- 103.5 (4) Create, monitor and kill processes 33
- 103.6 (2) Modify process execution priorities 35
- 103.7 (3) Search text files using regular expressions 36
- 103.8 (3) Basic file editing 40

Topic 104: (14) Devices, Linux Filesystems, Filesystem Hierarchy Standard **42**
- 104.1 (2) Create partitions and filesystems 44
- 104.2 (2) Maintain the integrity of filesystems 45
- 104.3 (3) Control mounting and unmounting of filesystems 46
- 104.5 (3) Manage file permissions and ownership 47
- 104.6 (2) Create and change hard and symbolic links 48

Introduction

Documentation

LPI Documentation

LPI Documentation

- Linux Professional Institute
  https://www.lpi.org
- LPI LPIX-1 overview
  https://www.lpi.org/our-certifications/lpic-1-overview
- LPIC1 Exam 101
  https://www.lpi.org/our-certifications/exam-101-objectives
- LPIC1 Exam 102
  https://www.lpi.org/our-certifications/exam-102-objectives

IBM Developerworks LPI Documentation

- Learn Linux, 101: A roadmap for LPIC-1
  https://developer.ibm.com/tutorials/l-lpic1-map/

PUE: Linux VM & resources

PUE LAB

- https://labs.pue.es
  VM visor: guacamole. Ctrl+Shift+Alt to open menú (cut&paste)

VM Virtual Machines

- PUE i386 Centos + Debian
  Dual boot CentOS 7 and Debian 10 i386 32 bits 5.1GiB *centian_i386.ova*
  https://u.pcloud.link/publink/show?code=XZcRI2kZmrD0lK2zjE7A3sY5xbdEVJKYWIty

- PUE x64 Centos + Debian
  Dual boot CentOS 7 and Debian 9 x64 64 bits 5.2GiB *centian_x64.ova*
  https://u.pcloud.link/publink/show?code=XZ0DKtkZax09QKYCk3hG3G6MuFQyhzbeECSk

- CentOS minimal (netinstall)
  CentOs 7 x86_64 64 bits 2009 973MiB
  https://u.pcloud.link/publink/show?code=XZ952BXZWRF0MNlrvD7IzVqiRd0ejmrp5rvV

- ** Usuarios de las máquinas: linux / linux root / toor**

Resources

- Retos del CLI https://cmdchallenge.com/
- Regular expressions https://regex101.com/
- Cheat Sheets https://cheatography.com/

Linux online terminals

- List of some linux web terminals geekfare
- Copy.sh http://copy.sh/v86/?profile=linux26
- JSLinux (select Fedora 33) http://bellard.org/jslinux/
- Distronet (select Fedora 33 Server) https://distrotest.net/index.php
- Webminal (register) https://www.webminal.org/
- Cocalc (register) https://cocalc.com/features/terminal

Fedora / Debian / Ubuntu VM qcow2 Cloud Images

Debian Official Cloud Images

- Repo: http://cloud.debian.org/images/cloud/
- Debian 10 Bullseye: http://cloud.debian.org/images/cloud/bullseye/latest/
  Image: *debian-11-nocloud-amd64.qcow2*
  This image has root access passwordless (user: root, no password)

Fedora

- Fedora Cloud 32 Repo: /pub/fedora/linux/releases/32/Cloud/x86_64/images
  https://dl.fedoraproject.org/pub/fedora/linux/releases/32/Cloud/x86_64/images
  Image: *Fedora-Cloud-Base-32-1.6.x86_64.qcow2*

  Fedora images have no root/user access. Image should be modified by virt-sysprep to set the root password:

virt-sysprep* belongs to the *libguestfs-tools* package

$ sudo virt-sysprep -a Fedora-name-image.qcow2 --root-password password:newpasswd

Ubuntu

- Ubuntu Cloud Images Repo: https://cloud-images.ubuntu.com/

Training / Exam Dumps Online

- Examtopics 101-500 → https://www.examtopics.com/exams/lpi/101-500/
- Exam4training 101-500 → https://www.exam4training.com/package-management/
- ITExams 101-500 → https://www.itexams.com/exam/101-500
- Prepare4sure 101-500 → https://www.prepare4sure.com/101-500-braindump.html
- Examtopics 102-500 → https://www.examtopics.com/exams/lpi/102-500/
- Exam4training 102-500 →https://www.exam4training.com/which-command-makes-the-shell-variable-named-variable-visible-to-subshells-3/
- ITExams 102-500 → https://www.itexams.com/exam/102-500
- Prepare4sure 102-500 → https://www.prepare4sure.com/102-500-braindump.html

LPIC-1 101 Exam topics / objectives

- 101 (8) System Architecture
- 102 (12) Linux Installation and Package Management
- 103 (26) GNU and Unix Commands
- 104 (14) Devices, Linux Filesystems, Filesystem Hierarchy Standard

View detailed exam objectives
View exam study resources LPI

LPIC-1 102 Exam topics / objectives

- 10 Shells and Shell Scripting
- Interfaces and Desktops
- Administrative Tasks
- Essential System Services
- Networking Fundamentals
- Security

View detailed exam objectives

View exam study resources LPI

Topic 101: (8) System Architecture

Topic 101: (8) System Architecture
- 101.1 (2) Determine and configure hardware settings
- 101.2 (3) Boot the system
- 101.3 (3) Change runlevels / boot targets and shutdown or reboot system

101.1 (2) Determine and configure hardware settings

Description: Candidates should be able to determine and configure fundamental system hardware

Key Knowledge Areas:
- Enable and disable integrated peripherals.
- Differentiate between the various types of mass storage devices.
- Determine hardware resources for devices.
- Tools and utilities to list various hardware information (e.g. lsusb, lspci, etc.).
- Tools and utilities to manipulate USB devices.
- Conceptual understanding of sysfs, udev and dbus.

The following is a partial list of the used files, terms and utilities:
- /sys/
- /proc/
- /dev/
- modprobe
- lsmod
- lspci
- lsusb

101.2 (3) Boot the system

Description: Candidates should be able to guide the system through the booting process.

Key Knowledge Areas:
- Provide common commands to the boot loader and options to the kernel at boot time.
- Demonstrate knowledge of the boot sequence from BIOS/UEFI to boot completion.
- Understanding of SysVinit and systemd.
- Awareness of Upstart.
- Check boot events in the log files.

The following is a partial list of the used files, terms and utilities:
- dmesg
- journalctl
- BIOS
- UEFI
- bootloader
- kernel
- initramfs
- init
- SysVinit
- systemd

101.3 (3) Change runlevels / boot targets and shutdown or reboot system

Description: Candidates should be able to manage the SysVinit runlevel or systemd boot target of the system. This objective includes changing to single user mode, shutdown or rebooting the system. Candidates should be able to alert users before switching runlevels / boot targets and properly terminate processes. This objective also includes setting the default SysVinit runlevel or systemd boot target. It also includes awareness of Upstart as an alternative to SysVinit or systemd.

Key Knowledge Areas:

- Set the default runlevel or boot target.
- Change between runlevels / boot targets including single user mode.
- Shutdown and reboot from the command line.
- Alert users before switching runlevels / boot targets or other major system events.
- Properly terminate processes.
- Awareness of acpid.

The following is a partial list of the used files, terms and utilities:

- /etc/inittab
- shutdown
- init
- /etc/init.d/
- telinit
- systemd
- systemctl
- /etc/systemd/
- /usr/lib/systemd/
- wall

101.1 (2) Determine and configure hardware settings

101.1 (2) Determine and configure hardware settings

Description: Candidates should be able to determine and configure fundamental system hardware

Key Knowledge Areas:
- Enable and disable integrated peripherals.
- Differentiate between the various types of mass storage devices.
- Determine hardware resources for devices.
- Tools and utilities to list various hardware information (e.g. lsusb, lspci, etc.).
- Tools and utilities to manipulate USB devices.
- Conceptual understanding of sysfs, udev and dbus.

The following is a partial list of the used files, terms and utilities:
- /sys/
- /proc/
- /dev/
- modprobe
- lsmod
- lspci
- lsusb

101.2 (3) Boot the system

101.2 (3) Boot the system

Description: Candidates should be able to guide the system through the booting process.

Key Knowledge Areas:

- Provide common commands to the boot loader and options to the kernel at boot time.
- Demonstrate knowledge of the boot sequence from BIOS/UEFI to boot completion.
- Understanding of SysVinit and systemd.
- Awareness of Upstart.
- Check boot events in the log files.

The following is a partial list of the used files, terms and utilities:

- dmesg
- journalctl
- BIOS
- UEFI
- bootloader
- kernel
- initramfs
- init
- SysVinit
- systemd

101.3 (3) Change runlevels / boot targets and shutdown or reboot system

101.3 (3) Change runlevels / boot targets and shutdown or reboot system

Description: Candidates should be able to manage the SysVinit runlevel or systemd boot target of the system. This objective includes changing to single user mode, shutdown or rebooting the system. Candidates should be able to alert users before switching runlevels / boot targets and properly terminate processes. This objective also includes setting the default SysVinit runlevel or systemd boot target. It also includes awareness of Upstart as an alternative to SysVinit or systemd.

Key Knowledge Areas:

- Set the default runlevel or boot target.
- Change between runlevels / boot targets including single user mode.
- Shutdown and reboot from the command line.
- Alert users before switching runlevels / boot targets or other major system events.
- Properly terminate processes.
- Awareness of acpid.

The following is a partial list of the used files, terms and utilities:

- /etc/inittab
- shutdown
- init
- /etc/init.d/
- telinit
- systemd
- systemctl
- /etc/systemd/
- /usr/lib/systemd/
- wall

Topic 102: (12) Linux Installation and Package Management

Topic 102: (12) Linux Installation and Package Management
- 102.1 (2) Design hard disk layout
- 102.2 (2) Install a boot manager
- 102.3 (1) Manage shared libraries
- 102.4 (3) Use Debian package management
- 102.5 (3) Use RPM and YUM package management
- 102.6 (1) Linux as a virtualization guest

102.1 (2) Design hard disk layout

Description: Candidates should be able to design a disk partitioning scheme for a Linux system.

Key Knowledge Areas:
- Allocate filesystems and swap space to separate partitions or disks.
- Tailor the design to the intended use of the system.
- Ensure the /boot partition conforms to the hardware architecture requirements for booting.
- Knowledge of basic features of LVM.

The following is a partial list of the used files, terms and utilities:
- / (root) filesystem
- /var filesystem
- /home filesystem
- /boot filesystem
- EFI System Partition (ESP)
- swap space
- mount points
- partitions

102.2 (2) Install a boot manager

Description: Candidates should be able to select, install and configure a boot manager.

Key Knowledge Areas:
- Providing alternative boot locations and backup boot options.
- Install and configure a boot loader such as GRUB Legacy.
- Perform basic configuration changes for GRUB 2.
- Interact with the boot loader.

The following is a partial list of the used files, terms and utilities:
- menu.lst, grub.cfg and grub.conf
- grub-install
- grub-mkconfig
- MBR

102.3 (1) Manage shared libraries

Description: Candidates should be able to determine the shared libraries that executable programs depend on and install them when necessary.

Key Knowledge Areas:
- Identify shared libraries.
- Identify the typical locations of system libraries.

- Load shared libraries.

The following is a partial list of the used files, terms and utilities:
- ldd
- ldconfig
- /etc/ld.so.conf
- LD_LIBRARY_PATH

102.4 (3) Use Debian package management

Description: Candidates should be able to perform package management using the Debian package tools.

Key Knowledge Areas:
- Install, upgrade and uninstall Debian binary packages.
- Find packages containing specific files or libraries which may or may not be installed.
- Obtain package information like version, content, dependencies, package integrity and installation status (whether or not the package is installed).
- Awareness of apt.

The following is a partial list of the used files, terms and utilities:
- /etc/apt/sources.list
- dpkg
- dpkg-reconfigure
- apt-get
- apt-cache

102.5 (3) Use RPM and YUM package management

Description: Candidates should be able to perform package management using RPM, YUM and Zypper.

Key Knowledge Areas:
- Install, re-install, upgrade and remove packages using RPM, YUM and Zypper.
- Obtain information on RPM packages such as version, status, dependencies, integrity and signatures.
- Determine what files a package provides, as well as find which package a specific file comes from.
- Awareness of dnf.

The following is a partial list of the used files, terms and utilities:
- rpm
- rpm2cpio
- /etc/yum.conf
- /etc/yum.repos.d/
- yum
- zypper

102.6 (1) Linux as a virtualization guest

Description: Candidates should understand the implications of virtualization and cloud computing on a Linux guest system.

Key Knowledge Areas:
- Understand the general concept of virtual machines and containers.
- Understand common elements virtual machines in an IaaS cloud, such as computing instances, block storage and networking.
- Understand unique properties of a Linux system which have to changed when a system is cloned or used as a template.
- Understand how system images are used to deploy virtual machines, cloud instances and containers.
- Understand Linux extensions which integrate Linux with a virtualization product.
- Awareness of cloud-init.

The following is a partial list of the used files, terms and utilities:
- Virtual machine
- Linux container
- Application container
- Guest drivers
- SSH host keys
- D-Bus machine id

102.1 (2) Design hard disk layout

102.1 (2) Design hard disk layout

Description: Candidates should be able to design a disk partitioning scheme for a Linux system.

Key Knowledge Areas:

- Allocate filesystems and swap space to separate partitions or disks.
- Tailor the design to the intended use of the system.
- Ensure the /boot partition conforms to the hardware architecture requirements for booting.
- Knowledge of basic features of LVM.

The following is a partial list of the used files, terms and utilities:

- / (root) filesystem
- /var filesystem
- /home filesystem
- /boot filesystem
- EFI System Partition (ESP)
- swap space
- mount points
- partitions

102.2 (2) Install a boot manager

102.2 (2) Install a boot manager

Description: Candidates should be able to select, install and configure a boot manager.

Key Knowledge Areas:
- Providing alternative boot locations and backup boot options.
- Install and configure a boot loader such as GRUB Legacy.
- Perform basic configuration changes for GRUB 2.
- Interact with the boot loader.

The following is a partial list of the used files, terms and utilities:
- menu.lst, grub.cfg and grub.conf
- grub-install
- grub-mkconfig
- MBR

102.3 (1) Manage shared libraries

102.3 (1) Manage shared libraries

Description: Candidates should be able to determine the shared libraries that executable programs depend on and install them when necessary.

Key Knowledge Areas:
- Identify shared libraries.
- Identify the typical locations of system libraries.
- Load shared libraries.

The following is a partial list of the used files, terms and utilities:
- ldd
- ldconfig
- /etc/ld.so.conf
- LD_LIBRARY_PATH

102.4 (3) Use Debian package management

102.4 (3) Use Debian package management

Description: Candidates should be able to perform package management using the Debian package tools.

Key Knowledge Areas:

- Install, upgrade and uninstall Debian binary packages.
- Find packages containing specific files or libraries which may or may not be installed.
- Obtain package information like version, content, dependencies, package integrity and installation status (whether or not the package is installed).
- Awareness of apt.

The following is a partial list of the used files, terms and utilities:

- /etc/apt/sources.list
- dpkg
- dpkg-reconfigure
- apt-get
- apt-cache

102.5 (3) Use RPM and YUM package management

102.5 (3) Use RPM and YUM package management

Description: Candidates should be able to perform package management using RPM, YUM and Zypper.

Key Knowledge Areas:

- Install, re-install, upgrade and remove packages using RPM, YUM and Zypper.
- Obtain information on RPM packages such as version, status, dependencies, integrity and signatures.
- Determine what files a package provides, as well as find which package a specific file comes from.
- Awareness of dnf.

The following is a partial list of the used files, terms and utilities:

- rpm
- rpm2cpio
- /etc/yum.conf
- /etc/yum.repos.d/
- yum
- zypper

102.6 (1) Linux as a virtualization guest

102.6 (1) Linux as a virtualization guest

Description: Candidates should understand the implications of virtualization and cloud computing on a Linux guest system.

Key Knowledge Areas:

- Understand the general concept of virtual machines and containers.
- Understand common elements virtual machines in an IaaS cloud, such as computing instances, block storage and networking.
- Understand unique properties of a Linux system which have to changed when a system is cloned or used as a template.
- Understand how system images are used to deploy virtual machines, cloud instances and containers.
- Understand Linux extensions which integrate Linux with a virtualization product.
- Awareness of cloud-init.

The following is a partial list of the used files, terms and utilities:

- Virtual machine
- Linux container
- Application container
- Guest drivers
- SSH host keys
- D-Bus machine id

Topic 103: (26) GNU and Unix Commands

Topic 103: (26) GNU and Unix Commands
- 103.1 (4) Work on the command line
- 103.2 (2) Process text streams using filters
- 103.3 (4) Perform basic file management
- 103.4 (4) Use streams, pipes and redirects
- 103.5 (4) Create, monitor and kill processes
- 103.6 (2) Modify process execution priorities
- 103.7 (3) Search text files using regular expressions
- 103.8 (3) Basic file editing

103.1 (4) Work on the command line

Description: Candidates should be able to interact with shells and commands using the command line. The objective assumes the Bash shell.

Key Knowledge Areas:
- Use single shell commands and one line command sequences to perform basic tasks on the command line.
- Use and modify the shell environment including defining, referencing and exporting environment variables.
- Use and edit command history.
- Invoke commands inside and outside the defined path.

The following is a partial list of the used files, terms and utilities:
- bash
- echo
- env
- export
- pwd
- set
- unset
- type
- which
- man
- uname
- history
- .bash_history
- Quoting

103.2 (2) Process text streams using filters

Description: Candidates should be able to apply filters to text streams.

Key Knowledge Areas:
- Send text files and output streams through text utility filters to modify the output using standard UNIX commands found in the GNU textutils package.

The following is a partial list of the used files, terms and utilities:
- bzcat
- cat
- cut
- head
- less
- md5sum
- nl
- od

- paste
- sed
- sha256sum
- sha512sum
- sort
- split
- tail
- tr
- uniq
- wc
- xzcat
- zcat

103.3 (4) Perform basic file management

Description: Candidates should be able to use the basic Linux commands to manage files and directories.

Key Knowledge Areas:
- Copy, move and remove files and directories individually.
- Copy multiple files and directories recursively.
- Remove files and directories recursively.
- Use simple and advanced wildcard specifications in commands.
- Using find to locate and act on files based on type, size, or time.
- Usage of tar, cpio and dd.

The following is a partial list of the used files, terms and utilities:
- cp
- find
- mkdir
- mv
- ls
- rm
- rmdir
- touch
- tar
- cpio
- dd
- file
- gzip
- gunzip
- bzip2
- bunzip2
- xz
- unxz
- file globbing

103.4 (4) Use streams, pipes and redirects

Description: Candidates should be able to redirect streams and connect them in order to efficiently process textual data. Tasks include redirecting standard input, standard output and standard error, piping the output of one command to the input of another command, using the output of one command as arguments to another command and sending output to both stdout and a file.

Key Knowledge Areas:
- Redirecting standard input, standard output and standard error.
- Pipe the output of one command to the input of another command.
- Use the output of one command as arguments to another command.
- Send output to both stdout and a file.

The following is a partial list of the used files, terms and utilities:
- tee
- xargs

103.5 (4) Create, monitor and kill processes

Description: Candidates should be able to perform basic process management.

Key Knowledge Areas:
- Run jobs in the foreground and background.
- Signal a program to continue running after logout.
- Monitor active processes.
- Select and sort processes for display.
- Send signals to processes.

The following is a partial list of the used files, terms and utilities:

- &
- bg
- fg
- jobs
- kill
- nohup
- ps
- top
- free
- uptime
- pgrep
- pkill
- killall
- watch
- screen
- tmux

103.6 (2) Modify process execution priorities

Description: Candidates should should be able to manage process execution priorities.

Key Knowledge Areas:

- Know the default priority of a job that is created.
- Run a program with higher or lower priority than the default.
- Change the priority of a running process.

The following is a partial list of the used files, terms and utilities:

- nice
- ps
- renice
- top

103.7 (3) Search text files using regular expressions

Description: Candidates should be able to manipulate files and text data using regular expressions. This objective includes creating simple regular expressions containing several notational elements as well as understanding the differences between basic and extended regular expressions. It also includes using regular expression tools to perform searches through a filesystem or file content.

Key Knowledge Areas:

- Create simple regular expressions containing several notational elements.
- Understand the differences between basic and extended regular expressions.
- Understand the concepts of special characters, character classes, quantifiers and anchors.
- Use regular expression tools to perform searches through a filesystem or file content.
- Use regular expressions to delete, change and substitute text.

The following is a partial list of the used files, terms and utilities:

- grep
- egrep
- fgrep
- sed
- regex(7)

103.8 (3) Basic file editing

Description: Candidates should be able to edit text files using vi. This objective includes vi navigation, vi modes, inserting, editing, deleting, copying and finding text. It also includes awareness of other common editors and setting the default editor.

Key Knowledge Areas:

- Navigate a document using vi.
- Understand and use vi modes.
- Insert, edit, delete, copy and find text in vi.
- Awareness of Emacs, nano and vim.
- Configure the standard editor.

Terms and Utilities:

- vi
- /, ?
- h,j,k,l
- i, o, a
- d, p, y, dd, yy

- ZZ, :w!, :q!
- EDITOR

103.1 (4) Work on the command line

103.1 (4) Work on the command line

Description: Candidates should be able to interact with shells and commands using the command line. The objective assumes the Bash shell.

Key Knowledge Areas:

- Use single shell commands and one line command sequences to perform basic tasks on the command line.
- Use and modify the shell environment including defining, referencing and exporting environment variables.
- Use and edit command history.
- Invoke commands inside and outside the defined path.

The following is a partial list of the used files, terms and utilities:

- bash
- echo
- env
- export
- pwd
- set
- unset
- type
- which
- man
- uname
- history
- .bash_history
- Quoting

Exercices

1. Show the current directory
2. Show the current date
3. List de active directory
4. Show all the information about the system (uname)
5. Show the kernel release
6. Show the users logged in the system
7. Who are you? :)

8. List the history
9. List the last 5 commands
10. Run the last command
11. Run the command number nº (choose one)

12. User ctrl+r to locate the previous uname commands

13. Show the help (--help) of the commands: pwd, who, whereis
14. Show the man pages of the command hostname
15. Show the sections of the manual pages.
16. Show the manual page of system the file /etc/passwd
17. Show the calendar (cal), the calendar of this year and the calendar of your birth month.

18. List the information of the executable, man pages and configuration of the commands: date, hostname, ls
19. Show the executable of the commands: cat, head, less
20. How many manual pages has the word passwd? Identify them using the whatis command. Which option of the man command does the same?.
21. Show all the pages containing the key word passwd. Which option of the man command does the same?

22. Show all the variables
23. Show all the environment variables
24. Show the value of the PATH variable
25. Assign to the name variable your name, and show it.
26. Assign to the fullname variable tour full name, and show it.
27. List (long list) the file .bash_history in your home directory.
28. Show the value of the shell variables: HISTFILESIZE, HISTFILE, HISTCONTROL and HISTIGNORE.

103.2 (2) Process text streams using filters

103.2 (2) Process text streams using filters

Description: Candidates should be able to apply filters to text streams.

Key Knowledge Areas:

- Send text files and output streams through text utility filters to modify the output using standard UNIX commands found in the GNU textutils package.

The following is a partial list of the used files, terms and utilities:

- bzcat
- cat
- cut
- head
- less
- md5sum
- nl
- od
- paste
- sed
- sha256sum
- sha512sum
- sort
- split
- tail
- tr
- uniq
- wc
- xzcat
- zcat

Exercices (1)

1. Show the contents of the file /etc/passwd
2. Show the contents of the file /etc/os-release and /etc/issue.
3. Show the contents of the file /etc/group page by page.
4. Using the cat command show the contents of the file /etc/os-release numbering the lines.

5. Show the first 8 lines of the file containing the user's accounts.
6. Show the last 12 lines of the file containing the group's accounts.
7. Show all the lines of the file /etc/group except the 5 last lines.

8. Count the lines of the file containing the user's accounts.

9. Count the line, words and characters of the file containing the group's accounts.
10. Use the command "ps ax" to show all the processes. Then count them.
11. Show the first 10 lines of the file /etc/passwd numbering the lines.
12. Show the file /etc/os-release numbering the lines

13. Show the contents of the last 3 lines of the file /etc/group in octal.
14. Show the contexts in HEX using hexdump command of the first 10 lines of the file /etc/passwd.
15. Locate the executable file of the command date. Issue the command "file /usr/bin/date" and then "hedump -C /usr/bin/date | head".

16. Show the contents of the file /etc/passwd ordered.
17. Show all the lines of the file /etc/service containing the string HTTP.
18. Show all the lines of the file /etc/service containing the strings FTP or ftp.
19. Show only the first column of the system users.
20. Show the login (1), uid(3) and (gid) information of the first 10 users.
21. Show the first 10 characters listing (long list) the root directory.

22. Calculate the SHA512 sum of the file /etc/services
23. Calculate the md5sum of the word "patata".

Exercices (2)

24. paste
25. join
26. tr
27. sed
28. cut
29. sort
30. uniq
31. split

103.3 (4) Perform basic file management

103.3 (4) Perform basic file management

Description: Candidates should be able to use the basic Linux commands to manage files and directories.

Key Knowledge Areas:
- Copy, move and remove files and directories individually.
- Copy multiple files and directories recursively.
- Remove files and directories recursively.
- Use simple and advanced wildcard specifications in commands.
- Using find to locate and act on files based on type, size, or time.
- Usage of tar, cpio and dd.

The following is a partial list of the used files, terms and utilities:
- cp
- find
- mkdir
- mv
- ls
- rm
- rmdir
- touch
- tar
- cpio
- dd
- file
- gzip
- gunzip
- bzip2
- bunzip2
- xz
- unxz
- file globbing

Exercicies (1):

1. Show the current directory.
2. Change to the root directory.
3. Change to the user's home directory.
4. Go to the parent directory. Show the current working directory.
5. Go to the directory /tmp
6. Create the directory lpi and make it the active directory

[active directory is lpi]
7. Create the directory system.

8. Create the directories monday, tuesday and test.
9. Delete the directories monday, tuesday.
10. Create the directories network/captures (in one shot).

[active directory is /tmp/lpi]

11. Using the tree command to show the directory structure.
12. Copy the file /etc/services to the active directory
13. Copy the file /etc/issue to the active directory and call it benvinguda.txt.
14. Copy the file benvinguda.txt to the directory /tmp/lpi/network/captures.
15. Copy the file /usr/bin/date tho the directory network //tmp/lpi/network).

16. Change the name of the file services for myservices.txt.
17. Change the directory name network for mynetwork
18. Move directory capture (/etc/lpi/captures) to test (/etc/lpi/test). Verify the contents of test, using ls and tree.
19. Move the directory test /tmp/lpi/test) to mynetrok (/tmp/lpi/mynetwork) changing his name to mytest. Verify the directory structure using tree /tmp/lpi.

[active directory is /tmp/lpi]

20. Using rmdir remove the directory /tmp/lpi/mynetwork/mytest/captures. Is possible?
21. Using rm -r (caution!) remove the directory captures (/tmp/lpi/mynetwork/mytest/captures).
22. Remove the contents of the directory mytest but not the directory.
23. Remove the directory mynetwork, the directory and all its contents.

[active directory is lpi]

24. Use: touch file1.txt file2.txt file35.txt file1a.txt file1b.txt filea.txt filea1.txt filename.pdf carta.pdf oldcarta.pdf to create the requested files.
25. Show all the filenames with txt extension.
26. Show all the filenames beginning with f.
27. Show all the filenames containing carta in the name.
28. Show all the filenames beginning with the letter c or o or m and pdf extension.
29. Show all the filenames beginning with file followed by 2 characters and txt extension.
30. Show all the filenames with exactly 5 characters in the name and whatever extension.
31. Show the filenames beginning with file followed by two digits and txt extension.
32. Show the filenames containing two digits in the name and text extension.

Exercicices (2):

33. Make active the home directory
34. Using find search for the file .bash_history. Show a long list.
35. Using find search for the files in the /usr/sbin directory starting with user.
36. As user root find the files in the /etc and /usr/sbin directories containing user in the filename.

37. Find the files in /tmp owned by your user.

38. Find the files in the /tmp directory uned by root.
39. Using find list the directory entries of /etc directory.
40. Using find list the block devices entries in the /dev directory. Idem as char devices. Is there any socket device?

41. List the /boot directory using long and human options.
42. Using find list the files in the /boot directory less than 15M.
43. Using fins list the files in the /boot directory with size between 5M and 20M.
44. List all the files in the system greater than 200M. Send the stderr to /dev/null.

45. Using find list the files newer than 5 days in the /var directory (mtime).
46. Using find list the files in /tmp newer than 1h (mmin).
47. Using find list the files in your home newer than the file .bashrc (newer).

48. Use updatedb to create de locate database.
49. Using locate list all the files with the string ifcfg in the name.
50. Using locate list all the filenames containing the string user.
51. Using locate identify the file containing the filesystem tabulation information: fstab.

Exercicis(3)

52. Change to the /tmp/lpi directory, create if doesn't exist.
53. Copy the file /usr/bin/date to the directory.
54. Split the file in three parts.
55. Concatenate the three parts again in a file called mydate.
56. Compare date and mydate.

57. Copy the file /etc/passwd and rename the copy as passwd.txt
58. Using gzip compress the file. Observe the new name, where is the original?
59. Use file to show information of the new file. Use ls -lh tho show the size.
60. Use zcat to show the contents of the compressed file.
61. Using gunzip decompress the file. Use the command file to show information of passwd.txt. Use ls -ls to show the size.
62. Repeat the last two exercicis using bzip2 and bunzip2.
63. Locate the man page of the date command. Show the contents of the file containing the man page.

64. Create a copy of the file date and call it date.original.
65. Using the local copy of the file date, compress it using gzip.
66. Split the compressed file in 20K portions. How many chunks are created?
67. Join the portions concatenating them in a new file called newdate.gz.
68. Compare that file with the compressed one.
69. Uncompress the file newdate.gz and compare it with the original.

Exercices (4)

70. tar
71. cpio
72. dd

103.4 (4) Use streams, pipes and redirects

103.4 (4) Use streams, pipes and redirects

Description: Candidates should be able to redirect streams and connect them in order to efficiently process textual data. Tasks include redirecting standard input, standard output and standard error, piping the output of one command to the input of another command, using the output of one command as arguments to another command and sending output to both stdout and a file.

Key Knowledge Areas:

- Redirecting standard input, standard output and standard error.
- Pipe the output of one command to the input of another command.
- Use the output of one command as arguments to another command.
- Send output to both stdout and a file.

The following is a partial list of the used files, terms and utilities:

- tee
- xargs

Exercices:

1. List the /bin and the /patatum directories
2. Repeat the previous order sending the error messages to /dev/null.
3. Find the ifcfg files in /boot discarting the error messages.

4. Count the users logged in the system.
5. Create a file with the first 15 lines of the user`s account file.
6. Create a file containing the concatenation of the files /etc/passwd and /etc/group.
7. Create a file containing the list of all current processes, using ps ax.

8. Append to the previous file the output of the execution of the command free.
9. Execute a tree command of the root directory and save the output to a file and the errors to another.
10. List the home directory and the /patata directory. Send all the output (stdout and stderr) to the same file.
11. Who is incorrect the next command:
    command 1>&2 2> file.log

12. Show to ‘the screen’ (stdout) the last eleven groups in the system and put them in a file. Both actions at the same time, show and save.
13. Filter the lines of the file /etc/passwd containing the value “1”, show the lines to stdout and append them to a file (at the same time).
14. Filter the lines of the file /etc/passwd containing the value “1”, put them into a file and show to stdout the number of lines.

15. Create the files carta.txt treball.pdf fit1 fit2 fit2 using echo and xargs.
16. Using ls -d and xargs calculate the disk usage (command du -sh) of tall the home directories starting with D.

Alternate exercises:

1. Llistar el número major i el número menor dels dispositius corresponents a la entrada estàndard, sortida estàndard i d'error. Seguir el link fins identificar el device real on esta lligat.
2. Desar en un fitxer de nom http.txt tots els serveis que continguin la cadena http.
3. Desar en un fitxer de nom http.txt tots els serveis que continguin la cadena http però que al mateix temps es mostri per pantalla.
4. Desar en un fitxer de nom ftp.txt el llistat de tots els serveis que contenen la cadena ftp ordenats lexicogràficament. La sortida s'ha de mostrar simultàniament per pantalla.
5. Ídem exercici anterior però mostrant per pantalla únicament quants serveis hi ha.
6. Ídem anterior però comptant únicament quants contenen la descrició TLS/SSL.
7. Llista l'ocupació d'espai del directori tmp fent que els missatges d'error s'ignorin.
8. Ídem anterior desant el resultat a disc.txt i ignorant els errors.
9. Ídem enviant tota la sortida (errors i dades) al fitxer disc.txt
10. Afegir al fitxer disc.txt el sumari de l'ocupació de disc dels directoris /boot i /mnt. Els errors cal ignorar-los.

103.5 (4) Create, monitor and kill processes

103.5 (4) Create, monitor and kill processes

Description: Candidates should be able to perform basic process management.

Key Knowledge Areas:
- Run jobs in the foreground and background.
- Signal a program to continue running after logout.
- Monitor active processes.
- Select and sort processes for display.
- Send signals to processes.

The following is a partial list of the used files, terms and utilities:
- &
- bg
- fg
- jobs
- kill
- nohup
- ps
- top
- free
- uptime
- pgrep
- pkill
- killall
- watch
- screen
- tmux

Alternate exercises:
1. Mostrar tots els processos del sistema.
2. Mostrar tot l'arbre de processos incloent el pid i la ordre.
3. Prova les ordres: ps, ps a, ps x, ps ax, ps -fu pere, ps -Fu pere.
4. Llistar els processos fent un llistat llarg. el PID i el PPID.
5. Entrar en un subshell i fer un llistat llarg dels processos.
6. Identificar el PID del procés pare del shell actual.
7. Identifica el PID del procés init usant l'ordre pidof.
8. Identifica el pid del servei d'impressió cupsd amb l'ordre pidof.
9. Usant l'ordre pgrep llista els processos de l'usuari root.
10. Usant l'ordre pgrep localitza el procés init.
11. Utilitzant l'ordre fuser per saber quins processos utilitzen el directori /tmp.
12. Llista tots els senyals de l’ordre kill.

13. Genera un procés sleep 10000 i mata'l amb kill.
14. Mata el bash actual.
15. Llista tots els processos mingetty i mata'ls de cop tots usant una sola ordre kill.
16. Genera 3 processos sleeep 10000 i mata'ls tots de cop usant killall.

17. Executa tres ordres sleep en segon pla i llista els treballs.
18. Inicia l'edició d'un fitxer amb vi i deixa'l suspès d'execució en segon pla. Mostrar els treballs.
19. Mata el segon dels treballs (un sleep).
20. Passa a primer pla el primer dels treballs (un sleep), i mata'l amb ctrl+c.
21. Passa a primer pla el treball més recent. Quin és. Acabar.
22. Llistar tota l'estructura de directoris partint de l'arrel. Que no es generin missatges d'error i enviar la sortida a null (no volem desar res és només per fer-lo treballar!). Un cop iniciat aturar el proces. Llistar els treballs.
23. Reanudar l'execució del tree anterior en segon pla.

24. Executar l'ordre que monitoritza els processos. Llistar-los per prioritat.
25. Executar l'ordre vmstat. Descriu almenys tres dels elements dels que informa.
26. Executar l'ordre free i descriure la informació que mostra.
27. Digues quanta estona fa que el sistema està engegat ininterrompudament.

103.6 (2) Modify process execution priorities

103.6 (2) Modify process execution priorities

Description: Candidates should should be able to manage process execution priorities.

Key Knowledge Areas:
- Know the default priority of a job that is created.
- Run a program with higher or lower priority than the default.
- Change the priority of a running process.

The following is a partial list of the used files, terms and utilities:
- nice
- ps
- renice
- top

Exercices

1. List the processes using ps -l and observe the NI field.

2. Execute the commands sleep 12345 & and sleep 67890 &. List the processes with ps -l. Which is the nice value of the two sleeps?

3. Change the nice value of the first sleep to 5 and the second to 15.

4. Watch the nice values using ps -l and top.

5. Execute the sleep 99999 & with a priority of 19. Verify the process priority.

6. Execute the command nice sleep 44444 &. Which is it’s priority?

103.7 (3) Search text files using regular expressions

103.7 (3) Search text files using regular expressions

Description: Candidates should be able to manipulate files and text data using regular expressions. This objective includes creating simple regular expressions containing several notational elements as well as understanding the differences between basic and extended regular expressions. It also includes using regular expression tools to perform searches through a filesystem or file content.

Key Knowledge Areas:

- Create simple regular expressions containing several notational elements.
- Understand the differences between basic and extended regular expressions.
- Understand the concepts of special characters, character classes, quantifiers and anchors.
- Use regular expression tools to perform searches through a filesystem or file content.
- Use regular expressions to delete, change and substitute text.

The following is a partial list of the used files, terms and utilities:

- grep
- egrep
- fgrep
- sed
- regex(7)

Exercices

[Grep]

1. Search all the lines in the file /etc/passwd containing the string root.

2. Search in the first 12 lines of the file /etc/passwd starting with the word root.

3. Search in the last 16 lines of the file /etc/passwd the lines ending with nologin.

4. Search in the file /etc/group for a group with the GID 42.

5. Search in the file /etc/passwd all the groups with GID from 20 to 29.

6. Using the echo and grep commands, validate if a DNI has an appropriate value (echo "12345688A" | grep ...).

7. Using the echo and grep commands, validate if a date has the format dd-mm-aa.

8. Filter the file /etc/services showing all the lines containing http or HTTP.

9. Filter the lines of the file /etc/fstab showing al the lines without the char #.

[tr]

10. Show the file /etc/group using one tabulation as a delimiter.

11. Show the lines of the file /etc/passwd ending with /bin/bash in upper case.

12. Show the last 3 lines of the file /etc/group transforming the vowels in numbers.

13. Filter the output of the command uname -a deleting all the numbers.

14. Filter the first 5 lines of the file /etc/passwd deleting all the vowels.

15. Squeeze all the spaces in the output of the ls -l / command.

16. Show the last 5 lines of the file /etc/fstab in a normalized format (using only one space or one tabulation as a delimiter).

[cut]

17. List (long list) the root directory and show only the characters from 2 to 10.

18. Show the login, uid and gid of the users accounts.

19. List the fields from the gid to the end, of all users using the /bin/bash shell.

20. List the gname, gid and user-list of the last 15 groups.

21. (*difficult*) List the root directory and show the fields: type/permissions, owner and name. It is necessary to normalize the blanks before cutting the fields.

[sort / uniq]

22. List by gname order all the system groups.

23. List the system groups sorted by gid in descendant order.

24. List the users using gid as a primary order key and uid as a secondary.

25. List the users accounts grouped by gid and sorted by login name in descendant order.

26. List all the different gids.

27. List all the different shells used by users.

[sed]

28. Change the /etc/passwd delimiter to --.

29. Delete all lines in the /etc/passwd containing the word root.

30. Change all occurrences of the string bin for the word BIN in the /etc/passwd file.

31. Do the command echo "935550055" and transform the output to (93) 555 00 55.

32. Cut the login and uid of the first 10 users in the password accounts file and show it in the format login(uid).

[paste]

33 Create the two next files and paste the contents to a new file.

```
cat <<EOF > file1.txt
IBM
INTC
SAP
VMW
EOF

cat <<EOF > file2.txt
174.99
22.99
59.37
102.92
EOF
```

34. Paste the two previous files using the -s option.

35. Create a file named fit1.txt containing the login, passwd and uid of the then first system users. Create a file named fit2 containing he gid, home and shell of the first then system users. Show the two files pasted.

[join]

36 Create the files noms.txt and cognoms.txt and join them. Observe the difference.

```
1 pere
3 marta
4 borrell
5 pau
9 rosa
```

```
1 puig
2 vilada
3 mas
5 pou
9 roure
```

37 Repeat the previous exercise doing a left join. And then, again doing a full outer join.

38 Create the file cognoms2.txt and join noms.txt with cognoms2.txt using the id field as key.

```
puig pedroś 1
vilada vendrell 2
mas morell 3
pou prat 5
roure ribalta 9
```

39 Repeat the previous exercise doing a full outer join.
join -a1 -a2 -j1 1 -j2 3 noms.txt cognoms2.txt

103.8 (3) Basic file editing

103.8 (3) Basic file editing

Description: Candidates should be able to edit text files using vi. This objective includes vi navigation, vi modes, inserting, editing, deleting, copying and finding text. It also includes awareness of other common editors and setting the default editor.

Key Knowledge Areas:

- Navigate a document using vi.
- Understand and use vi modes.
- Insert, edit, delete, copy and find text in vi.
- Awareness of Emacs, nano and vim.
- Configure the standard editor.

Terms and Utilities:

- vi
- /, ?
- h,j,k,l
- i, o, a
- d, p, y, dd, yy
- ZZ, :w!, :q!
- EDITOR

Exercices

1. Crear un fitxer anomenat *carta.txt* al directori actiu, amb el següent contingut i desa'l.

```
aquest és un text qualsevol.
podem escriure múltiples línies
com per exemple questes.

llistat de noms:
pau pou
marta pou
anna pou
jordi mas
pere mas
admin sys
```

2. Edita el fitxer *carta.txt* del directori actiu i fes:
   - elimina amb *dd* el nom "pau pou"
   - elimina amb *3dd* les tres línies següents
   - amb *u* anul·la l'última acció.
   - torna a prémer *u* per anul·lar l'esborrat de "pau pou".

3. Desa una còpia anomenada *carta.bk* del contingut actual. Des d'una altra sessió fes un cat per mostrar el contingut de carta.bk

4. Edita el fitxer *carta.txt* i fes les accions següents:
    - Ves al final del document prement *G*.
    - Ves al principi del document prement *gg*.
    - Ves a la cinquena línia prement *5G*.
    - Insereix una línia nova a sota prement *o* i escriu el contingut: “julia mas”.

5. Edita el fitxer carta.txt i situa el cursor a la cinquena línia sobre la paraula “mas”, sense estar en mode inserció. Fes les accions:
    - Situa el cursor a “julia mas” (sobre mas).
    - Prem *dd* per esborrar una línia.
    - Prem el caràcter punt . dues vegades per repetir la última acció dos cops.
    - Surt del fitxer sense desar amb *q!*.
    - Què fan les següents accios: i, a, o, O?

6. Edita el text *carta.txt* i practica:
    - Situat al principi del document amb *gg*.
    - Busca la paraula pou fent escape /pou
    - Busca la següent aparició de la paraula pou.
    - Busca la anterior aparició de la paraula pou.
    - Situat a la última línia.
    - Busca cap a l’inici la paraula marta

7. Edita el text *carta.txt* i practica:
    - Situat a la línia amb “pau pou” i copia questa línia i les tres següents al porta papers.
    - Ves al final del document, insereix una línia en blanc i enganxa el contingut del portapapers.
    - Activa la numeració de les línies.
    - Ves a l’inici del document i enganxa el portpapers amb P.
    - Desfès la acció anterior (undo).
    - Retalla (elimina) les tres línies inicials.
    - Enganxa-les de nou al final del primer llistat de noms
    - Surt sense desar els canvis.

8. Edita el text *carta.txt* i practica:
    - Edita el fitxer mantenint-lo obert.
    - Des d’un altre shell afegeix la data (date >> carta.txt). Després el uname i després el uptime.
    - Torna al vi i observa que detecta que el fitxer ha estat modificat. Recarrega’l sense sortir.

Topic 104: (14) Devices, Linux Filesystems, Filesystem Hierarchy Standard

Topic 104: (14) Devices, Linux Filesystems, Filesystem Hierarchy Standard
- 104.1 (2) Create partitions and filesystems
- 104.2 (2) Maintain the integrity of filesystems
- 104.3 (3) Control mounting and unmounting of filesystems
- 104.5 (3) Manage file permissions and ownership
- 104.6 (2) Create and change hard and symbolic links
- 104.7 (2) Find system files and place files in the correct location

104.1 (2) Create partitions and filesystems

Description: Candidates should be able to configure disk partitions and then create filesystems on media such as hard disks. This includes the handling of swap partitions.

Key Knowledge Areas:
- Manage MBR and GPT partition tables
- Use various mkfs commands to create various filesystems such as:
- ext2/ext3/ext4
- XFS
- VFAT
- exFAT
- Basic feature knowledge of Btrfs, including multi-device filesystems, compression and subvolumes.

The following is a partial list of the used files, terms and utilities:
- fdisk
- gdisk
- parted
- mkfs
- mkswap

104.2 (2) Maintain the integrity of filesystems

Description: Candidates should be able to maintain a standard filesystem, as well as the extra data associated with a journaling filesystem.

Key Knowledge Areas:
- Verify the integrity of filesystems.
- Monitor free space and inodes.
- Repair simple filesystem problems.

The following is a partial list of the used files, terms and utilities:
- du
- df
- fsck
- e2fsck
- mke2fs
- tune2fs
- xfs_repair
- xfs_fsr
- xfs_db

104.3 (3) Control mounting and unmounting of filesystems

Description: Candidates should be able to configure the mounting of a filesystem.

Key Knowledge Areas:
- Manually mount and unmount filesystems.
- Configure filesystem mounting on bootup.
- Configure user mountable removable filesystems.
- Use of labels and UUIDs for identifying and mounting file systems.
- Awareness of systemd mount units.

The following is a partial list of the used files, terms and utilities:
- /etc/fstab
- /media/
- mount
- umount
- blkid
- lsblk

104.5 (3) Manage file permissions and ownership

Description: Candidates should be able to control file access through the proper use of permissions and ownerships.

Key Knowledge Areas:
- Manage access permissions on regular and special files as well as directories.
- Use access modes such as suid, sgid and the sticky bit to maintain security.
- Know how to change the file creation mask.
- Use the group field to grant file access to group members.

The following is a partial list of the used files, terms and utilities:
- chmod
- umask
- chown
- chgrp

104.6 (2) Create and change hard and symbolic links

Description: Candidates should be able to create and manage hard and symbolic links to a file.

Key Knowledge Areas:
- Create links.
- Identify hard and/or soft links.
- Copying versus linking files.
- Use links to support system administration tasks.

The following is a partial list of the used files, terms and utilities:
- ln
- ls

104.7 (2) Find system files and place files in the correct location

Description: Candidates should be thoroughly familiar with the Filesystem Hierarchy Standard (FHS), including typical file locations and directory classifications.

Key Knowledge Areas:
- Understand the correct locations of files under the FHS.
- Find files and commands on a Linux system.
- Know the location and purpose of important file and directories as defined in the FHS.

The following is a partial list of the used files, terms and utilities:
- find
- locate
- updatedb
- whereis
- which
- type
- /etc/updatedb.conf

104.1 (2) Create partitions and filesystems

104.1 (2) Create partitions and filesystems

Description: Candidates should be able to configure disk partitions and then create filesystems on media such as hard disks. This includes the handling of swap partitions.

Key Knowledge Areas:

- Manage MBR and GPT partition tables
- Use various mkfs commands to create various filesystems such as:
- ext2/ext3/ext4
- XFS
- VFAT
- exFAT
- Basic feature knowledge of Btrfs, including multi-device filesystems, compression and subvolumes.

The following is a partial list of the used files, terms and utilities:

- fdisk
- gdisk
- parted
- mkfs
- mkswap

104.2 (2) Maintain the integrity of filesystems

104.2 (2) Maintain the integrity of filesystems

Description: Candidates should be able to maintain a standard filesystem, as well as the extra data associated with a journaling filesystem.

Key Knowledge Areas:

- Verify the integrity of filesystems.
- Monitor free space and inodes.
- Repair simple filesystem problems.

The following is a partial list of the used files, terms and utilities:

- du
- df
- fsck
- e2fsck
- mke2fs
- tune2fs
- xfs_repair
- xfs_fsr
- xfs_db

104.3 (3) Control mounting and unmounting of filesystems

104.3 (3) Control mounting and unmounting of filesystems

Description: Candidates should be able to configure the mounting of a filesystem.

Key Knowledge Areas:

- Manually mount and unmount filesystems.
- Configure filesystem mounting on bootup.
- Configure user mountable removable filesystems.
- Use of labels and UUIDs for identifying and mounting file systems.
- Awareness of systemd mount units.

The following is a partial list of the used files, terms and utilities:

- /etc/fstab
- /media/
- mount
- umount
- blkid
- lsblk

104.5 (3) Manage file permissions and ownership

104.5 (3) Manage file permissions and ownership

Description: Candidates should be able to control file access through the proper use of permissions and ownerships.

Key Knowledge Areas:

- Manage access permissions on regular and special files as well as directories.
- Use access modes such as suid, sgid and the sticky bit to maintain security.
- Know how to change the file creation mask.
- Use the group field to grant file access to group members.

The following is a partial list of the used files, terms and utilities:

- chmod
- umask
- chown
- chgrp

104.6 (2) Create and change hard and symbolic links

104.6 (2) Create and change hard and symbolic links

Description: Candidates should be able to create and manage hard and symbolic links to a file.

Key Knowledge Areas:

- Create links.
- Identify hard and/or soft links.
- Copying versus linking files.
- Use links to support system administration tasks.

The following is a partial list of the used files, terms and utilities:

- ln
- ls

104.7 (2) Find system files and place files in the correct location

104.7 (2) Find system files and place files in the correct location

Description: Candidates should be thoroughly familiar with the Filesystem Hierarchy Standard (FHS), including typical file locations and directory classifications.

Key Knowledge Areas:

- Understand the correct locations of files under the FHS.
- Find files and commands on a Linux system.
- Know the location and purpose of important file and directories as defined in the FHS.

The following is a partial list of the used files, terms and utilities:

- find
- locate
- updatedb
- whereis
- which
- type
- /etc/updatedb.conf