



Practica 1 (GNU/Linux y Conceptos- Resolucion

1. Características de GNU/Linux:

a. Mencione y explique las características más relevantes de GNU/Linux.

- Todo es un archivo en linux.
- Puede ser usado, copiado, estudiado y redistribuido ya que mantiene una característica de software libre
- Correccion rapida a los errores

b. Mencione otros sistemas operativos y compárelos con GNU/Linux en cuanto a los puntos mencionados en el inciso a.

- Comparando Windows con Linux:
 - Nula distribucion libre
 - Siempre viene con restricciones y no permite su modificacion directa
 - Correccion de fallas a cargo del propietario y normalmente hay que esperar a actualizaciones futuras
 - Siempre viene con un costo asociado

c. ¿Qué es GNU?

- GNU (GNU NotUnix) es un sistema operativo similar a Unix con la diferencia de que este esta compuesto en su totalidad por software libre, ademas de contener paquetes y demas.

d. Indique una breve historia sobre la evolución del proyecto GNU.

e. Explique qué es la multitarea, e indique si GNU/Linux hace uso de ella.

- Si hace uso de multitarea, ademas de tener multiprocesador (ejemplo pueden ser 4 cada uno con su PC) y multiusuarios. La multitarea tiene como objetvo permitirle al usuario estar realizando varias tareas al mismo tiempo sin que ninguna de ellas sea interrumpida por la otra

f. ¿Qué es POSIX?

-Es una familia de estándares que tiene como objetivo la interoperabilidad en los sistemas operativos. Si se sigue un estándar POSIX es muy probable que ese sistema operativo se pueda utilizar en SO de UNIX, de lo contrario puede tener alguna falla y no funcionar en todos los sistemas

2. Distribuciones de GNU/Linux:

a. ¿Qué es una distribución de GNU/Linux? Nombre al menos 4 distribuciones de GNU/Linux y cite diferencias básicas entre ellas.

-Son un conjunto de aplicaciones que permiten brindar mejoras o distintas herramientas para instalar fácilmente un sistema operativo Linux/GNU (Diferentes interfaces, uso de HW, etc). Se diferencian entre sí justamente por esto mismo, depende de las necesidades del usuario y de lo que lo que esta distribución ofrezca para que el usuario tome la decisión por gusto personal de instalar una u otra.

-Distribuciones:

- Ubuntu, se centra en ser lo más amigable o friendly para la hora de la instalación

- Debian, se centra en servidores Linux

- SteamOS, se centra en los videojuegos

- Linux Mint, uso de HW más potente para competir con Windows

b. ¿En qué se diferencia una distribución de otra?

-Lo único que tienen en común es el Kernel, y lo que cambia es todo el resto de herramientas/componentes/funcionalidades ya que la shell, la GUI, herramientas de cada distribución son diferentes y se personalizan desde 0.

c. ¿Qué es Debian? Acceda al sitio e indique cuáles son los objetivos del proyecto y una breve cronología del mismo.

-Debian es un grupo de personas con una causa común que se juntaron para crear un SO libre

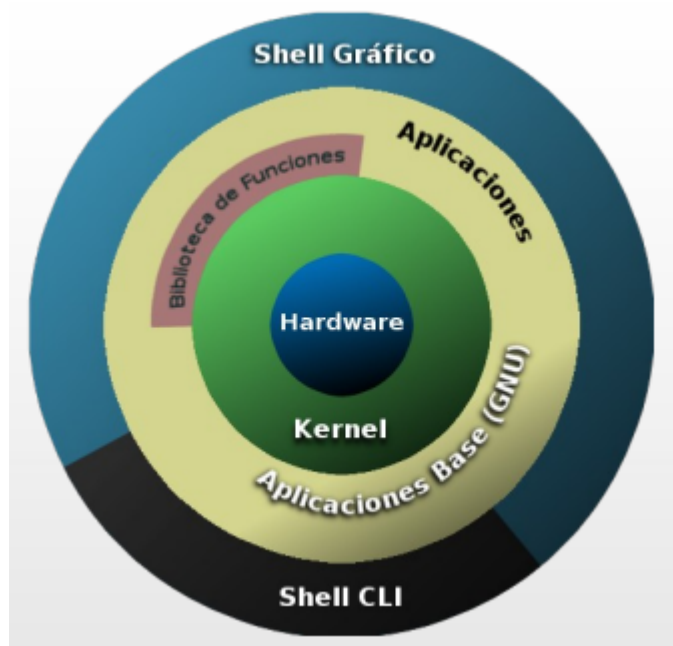
3. Estructura de GNU/Linux:

a. Nombre cuáles son los 3 componentes fundamentales de GNU/Linux.

-Kernel (núcleo), Shell (intérprete de comandos, intermediario entre usuario y Kernel) y FileSystem (sistema de archivos)

b. Mencione y explique la estructura básica del Sistema Operativo GNU/Linux.

-La estructura básica de un SO es el Kernel, el cual se encarga de ejecutar los programas y de toda la gestión de dispositivos HW ya que nosotros los usuarios no tenemos acceso directo a ello.



4. Kernel

a. ¿Cuáles son sus funciones principales?

-Su función principal es la administración de memoria, CPU y de dispositivos E/S

b. ¿Cuál es la versión actual? ¿Cómo se definía el esquema de versionado del Kernel en versiones anteriores a la 2.4? ¿Qué cambió en el versionado que se impuso a partir de la versión 2.6?

-5.6. Anteriormente los números pares indicaban que la versión lanzada era estable (1.2, 2.4, ..) y los números impares indicaban que eran versiones en desarrollo y podían contener fallos (2.5, 2.7)

c. ¿Es posible tener más de un Kernel de GNU/Linux instalado en la misma máquina?

-Si es posible tener más de uno instalado pero al momento de levantar el SO solo funciona uno

d. ¿Dónde se encuentra ubicado dentro del File System?

-Se encuentra en /boot en el primer sector del disco y nos indica información general de donde se almacena el Kernel y como arranca

5. Intérprete de comandos (Shell):

a. ¿Qué es?

-Es un programa que recibe lo que se escriba en la terminal y lo transforma en instrucciones para el Kernel. Es un intermediario entre el usuario y el Kernel

b. ¿Cuáles son sus funciones?

-Interfaz mas "amigable" para comunicar al usuario con el Kernel. Cual tipo de mensaje los espera mediante una ventana que espera comandos textuales y los interpreta, decodifica y entrega al SO para que se ejecuten

c. Mencione al menos 3 intérpretes de comandos que posee GNU/Linux y compárelos entre ellos.

d. ¿Dónde se ubican (path) los comandos propios y externos al Shell?

-Los comandos externos (ls, cat, mkdir) se buscan en los directions indicados en la variable PATH

-Los comandos internos (cd, bg, alias, eval) se ejecutan en el mismo proceso del shell

e. ¿Por qué considera que el Shell no es parte del Kernel de GNU/Linux?

-Ya que actua de intermediario o de mensajero con el Kernel, si estuviera en el mismo, el shell podria interpretar ordenes de usuario y ejectarlas al instante sobre el HW y no queremos eso, tendria acceso a instrucciones propias del SO de manera inmedianta.

f. ¿Es posible definir un intérprete de comandos distinto para cada usuario?

¿Desde dónde se define? ¿Cualquier usuario puede realizar dicha tarea?

-Si porque cada usuario tiene asignado un shell que se inicia con el proceso login, el cual hay que realizar y se logea a una terminal

6. El sistema de Archivos (File System) en Linux:

a. ¿Qué es?

-Es la forma en la que los archivos se organizan y se administran en un SO

b. ¿Cuál es la estructura básica de los File System en GNU/Linux? Mencione los directorios más importantes e indique qué tipo de información se encuentra en ellos. ¿A qué hace referencia la sigla FHS?

-Cuenta con paquetes de software para diferenciar distribuciones, editores de texto, herramientas de networking , paquete de oficina e interfaces graficas

-Directorios mas importantes segun FHS (Filesystem Hierarchy Standard):

- / → tope de la estructura (C:/ en windows)

-/home → almacena archivos de usuarios

-/var → informacion que varia de tamaño

-/etc → archivos de configuracion

-/bin → archivos binarios y ejecutables

-/dev → enlace a dispositivos

-/usr → aplicaciones de usuarios

al instalar un archivo las diferentes "partes" o características se encuentran en sus debidos directorios

c. Mencione sistemas de archivos soportados por GNU/Linux.

-ext2, ext3, FAT, FAT3, NTFS

d. ¿Es posible visualizar particiones del tipo FAT y NTFS (que son de Windows) en GNU/Linux?

-Linux no permite ver contenido de particiones de Windows

7. Particiones:

a. Definición. Tipos de particiones. Ventajas y Desventajas.

-Cada particion es una division presente en una sola unidad fisica de almacenamiento de datos (disco).

-Ventajas y Desventajas:

-Separa los datos del usuario de las aplicaciones y del SO

-Puede ubicar el Kernel en una particion read only

-Backup general del sistema

-Muchas particiones puede generar fragmentacion (archivos grandes no entren en el tamaño de una de las particiones)

-Tipos:

-Particion Primaria: Divisiones primarias (maximo 4) del disco que dependen de una tabla de particiones detectada por el SO al arrancar.

-Particion Secundaria o extendida: Se creo para obtener mas de 4 particiones en el disco duro y se utilizan unicamente para el almacenamiento de datos ya que no se puede instalar un SO en ellas

-Particion Logica: Se hacen dentro de la particion secundaria o extendida y se le asigna un tamaño y tipo de sistema de archivos

b. ¿Cómo se identifican las particiones en GNU/Linux? (Considere discos IDE, SCSI y SATA).

-Disqueteras:

-Primera : /dev/fd0 ; Segunda: /dev/fd1

-Discos:

-Primer disco (todo): /dev/hda

-Particiones primarias:

-Primera: /dev/hda1 ; Segunda: /dev/hda2 ; Tercera: /dev/hda3 ;
Cuarta: /dev/hda4

-Particiones logicas:

-Primera: /dev/hda5 ; Todas las demas: /dev/hda#

-Segundo disco: /dev/hdb

-Particiones primarias:

-Primera: /dev/hdb1 ; Segunda: /dev/hdb2 ; Tercera: /dev/hdb3 ;
Cuarta: /dev/hdb4

-Particiones logicas:

-Primera: /dev/hdb5 ; Todas las demas: /dev/hdb#

-Discos SCSI:

-Primer disco SCSI: /dev/sda

-Segundo disco SCSI: /dev/sdb

-Tercero y sucesivamente: /dev/sd#

c. ¿Cuántas particiones son necesarias como mínimo para instalar GNU/Linux? Nómbralas indicando tipo de partición, identificación, tipo de File System y punto de montaje.

-Mínimo una para instalar el (/) aunque se recomiendan dos para también instalar el SWAP

d. Dar ejemplos de diversos casos de particionamiento dependiendo del tipo de tarea que se deba realizar en su sistema operativo.

e. ¿Qué tipo de software para particionar existe? Menciónelos y compare.

-Destructivos (permiten crear y eliminar particiones → fdisk) y No destructivos (permiten crear, eliminar y modificar particiones generalmente desde la interfaz → fips, gparted)

8. Arranque (bootstrap) de un Sistema Operativo:

a. ¿Qué es el BIOS? ¿Qué tarea realiza?

-La BIOS es el Basic Input Output System y se encarga del manejo y configuración de la placa base y sus componentes. Inicia los componentes del HW y lanza el SO cuando prendes la computadora ya que la BIOS es lo primero que se carga en él

b. ¿Qué es UEFI? ¿Cuál es su función?

-UEFI es Unified Extensible Firmware Interface) y es un nexo entre el sistema operativo y el firmware que fue desarrollado por Intel. Es una alternativa para reemplazar la BIOS

c. ¿Qué es el MBR? ¿Qué es el MBC?

-El MBR es el Master Boot Record y se encuentra en el primer sector del disco (cilindro 0, cabeza 0, sector 1). Es un registro de arranque principal que se carga en memoria y se ejecuta. Contiene un código de arranque llamado MBC (Master Boot Code) y una marca de 2 bytes que indica su presencia o también puede tener una tabla de particiones

d. ¿A qué hacen referencia las siglas GPT? ¿Qué sustituye? Indique cuál es su formato.

-GPT es GUID Partition Table, y es un solucionador para las limitaciones del MBR, las cuales pueden ser cantidad de particiones, o tamaño de las mismas. GPT establece una ubicación y formato para la tabla de particiones en el disco duro

e. ¿Cuál es la funcionalidad de un "Gestor de Arranque"? ¿Qué tipos existen? ¿Dónde se instalan? Cite gestores de arranque conocidos.

-Es un bootloader que se encarga exclusivamente de preparar todo lo necesario para que el SO funcione. Se instala normalmente en el MBR y nos permitira elegir el sistema operativo a arrancar. En Linux existen distintos tipos de gestor de arranque: LILO (Linux Loader), GAG (Gestor Arranque Grafico)

f. ¿Cuáles son los pasos que se suceden desde que se prende una computadora hasta que el Sistema Operativo es cargado (proceso de bootstrap)?

-Lo primero que sucede es que se arranca la BIOS y se encarga de iniciar la carga del SO a traves del MBC ubicado en el MBR. Carga el programa de booteo y el gestor de arranque toma presencia y carga el Kernel, el cual prueba y hace disponibles los dispositivos para darle el control al init

g. Analice el proceso de arranque en GNU/Linux y describa sus principales pasos.

-Cuando se arranca la computadora, el BIOS se ejecuta realizando el POST (Power-on self-test), que incluye rutinas que, entre otras actividades, fijan valores de las señales internas, y ejecutan test internos (RAM, el teclado, y otros dispositivos a través de los buses ISA y PCI). Luego se lee el primer sector del disco llamado MBR que se carga en memoria y se ejecuta el MBC. Este puede ser de varios tipos, en el caso de Linux, el más frecuentemente usado era LILO, pero ya hace tiempo que se usa en bastantes distribuciones un cargador alternativo, llamado GRUB. Otros Sistemas Operativos tienen su propio programa cargador. Usaremos LILO en la descripción, pues es más ilustrativo.

En el caso concreto de LILO, lo que se carga en el sector de arranque es una parte de éste, denominada "first stage boot loader" (primer paso del cargador de inicio). Su misión es cargar y ejecutar el segundo paso del cargador de inicio.

Esta segunda parte suele mostrar una selección de Sistemas Operativos a cargar, procediendo a cargar a continuación el sistema escogido por el usuario (o bien el que se haya predeterminado como sistema por defecto, tras un tiempo de espera, si no escogemos nada). Esta información está

incluida dentro del cargador de inicio y, para introducirla, se usa la orden 'lilo' que a su vez usa el contenido de '/etc/lilo.conf'. Todo ello sucede, por supuesto, con el ordenador ya en marcha.

Una vez LILO ha cargado el "kernel" (núcleo) de Linux, le pasa el control a éste. Al cargarlo, le ha pasado algunos parámetros. De éstos, el más importante es el que le dice al núcleo qué dispositivo usar como sistema de ficheros raíz, es decir, lo que en UNIX se denomina '/'. En un ordenador de sobremesa, la raíz sería típicamente una partición de un disco duro, pero en sistemas incrustados es frecuente usar como raíz una partición virtual basada en memoria (Flash, RAM,...). Si el núcleo ha conseguido montar el sistema de ficheros raíz, lo siguiente a ejecutar es el programa 'init'. Sólo si dicho programa es estático (es decir, no usa librerías de funciones externas), no será necesario tener acceso a dichas librerías en la raíz. La librería básica en todo sistema GNU/Linux es la librería estándar C, "glibc". En un sistema mínimo, es decir, con una funcionalidad muy concreta, inmutable y sencilla, con tener solamente el programa 'init' enlazado estáticamente sería suficiente (y el núcleo, claro). En ese caso, init sería en realidad nuestro programa de aplicación al completo.

En general, 'init' es sólo el programa que se encarga de arrancar el resto de procesos que la máquina debe ejecutar. Entre sus tareas está el comprobar y montar sistemas de archivos, así como iniciar programas servidores (daemons) para cada función necesaria. Otra tarea importante es la de arrancar procesos 'getty' cuya misión es proporcionar consolas donde poder registrarse y entrar en el sistema. Las órdenes a seguir por 'init' están en el fichero '/etc/inittab'. A partir de ese punto, y en función del sistema de inicialización utilizado (el más frecuente es el denominado "System V") el proceso seguido por 'init' es distinto, pero en el fondo obedece más a un factor de forma, es decir, a una estrategia de ordenamiento de los "scripts" de inicialización de los distintos procesos que a un factor de fondo. Una vez iniciados todos los servidores y procesos de entrada de usuario, o bien estamos delante de una consola de texto en la que el ordenador nos pide que nos identifiquemos, o bien estamos ante una consola gráfica que nos pide lo mismo, o bien estamos ante una pantalla llena de opciones sobre qué ejecutar (escuchar música, ver películas, por ejemplo) si el sistema arranca bajo un usuario predeterminado y no nos pide registrarnos. Esto es, si es que hablamos de un ordenador de sobremesa que, típicamente, nos

ofrece una interfaz basada en dispositivos de entrada (teclado, ratón, mando a distancia) y de salida (monitor, TV, audio) para interactuar con él. Pero si el ordenador que se ha iniciado es un dispositivo con una funcionalidad concreta y su misión es controlar una serie de procesos y accedemos a él a través de medios indirectos (como pueda ser un navegador Web), el ordenador se inicia cuando está en disposición de prestar sus servicios, aun cuando no haya una indicación visual de dicho estado.

h. ¿Cuáles son los pasos que se suceden en el proceso de parada (shutdown) de GNU/Linux?

-Una vez se ejecuta el comando shutdown pueden ocurrir 2 escenarios:

- 1) Usuario unico del sistema, debe terminar todos los programas en ejecucion y hacer un log out de tdoas las sesiones de todos lados o mantener la sesion si ya existe una pero cambiando de directorio de trabajo para evitar problemas de demontajed el fyle system. Se ejecuta el comando shutdown -h now y se para
- 2) Muchos usuario, se utiliza el comando shutdown -h +time mensaje, donde time es el numero de minutos en el que la detenida del programa se posterga y el mensaje explica por que se esta haciendo el shutdown. Advierte a todos los usuarios que el sistema se apagara en X tiempo y lo mejor seria que hagan una desconexion o logout para no perder informacion

i. ¿Es posible tener en una PC GNU/Linux y otro Sistema Operativo instalado? Justifique.

-Si , particionando el disco y en cada particion tener un file system diferente

9. Archivos y editores:

a. ¿Cómo se identifican los archivos en GNU/Linux?

-El sistema de archivos nos permite que Linux maneje los archivos, los cuales tienen un nombre entre 1 y 255 caracteres y es case sensitive. La base del sistema de archivos de linux es una estructura empleada por el SO para almacenar informacion en un dispositivo fisico

b. Investigue el funcionamiento de los editores vim, nano y mcedit, y los comandos cat, more y less.

-Algunas pruebas de vim escribiendo en el txt y dps mostrarlo de diferentes modos:

```
~
"archivo2.txt" [New] 2L, 24B written
agus@agus:~$ cat archivo2.txt
hola escribo en el txt
agus@agus:~$ _
```

```
agus@agus:~$ cat archivo2.txt
hola escribo en el txt

agus@agus:~$ more archivo2.txt
hola escribo en el txt

agus@agus:~$ less archivo2.txt
hola escribo en el txt

archivo2.txt (END)
```

c. Cree un archivo llamado "prueba.exe" en su directorio personal usando el vim. El mismo debe contener su número de alumno y su nombre.

-vim prueba.exe y escribo 23005/2 - Agustin Gonzalez , lo muestro:

```
~
"prueba.exe" [New] 2L, 28B written
agus@agus:~$ cat prueba.exe
23005/2 - AGUSTIN GONZALEZ
agus@agus:~$ _
```

d. Investigue el funcionamiento del comando file. Pruébalo con diferentes archivos. ¿Qué diferencia nota?

-Diferentes archivos muestra diferentes tipos, si mostras un ejecutable tambien cambia:

```
agus@agus:~$ file prueba.exe
prueba.exe: ASCII text
agus@agus:~$ file archivo2.txt
archivo2.txt: ASCII text
agus@agus:~$ file /bin/ls
/bin/ls: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked
a574bbc8ef951ddde9b21, for GNU/Linux 3.2.0, stripped
agus@agus:~$ _
```

e. Investigue la funcionalidad y parámetros de los siguientes comandos relacionados con el uso de archivos:

i. cd → cambias de carpeta en la que estas parado

ii. mkdir → creas carpeta

iii. rmdir → borras carpeta

```
mkdir carpeta1
rmdir carpeta1
```

iv. ln → crea enlaces

```
ln prueba.exe enlace
ln -s prueba.exe enlace_simbolico
```

v. tail → muestra ultimas lineas del archivo

```
agus@agus:~$ tail prueba.exe
23005/2 - AGUSTIN GONZALEZ
```

vi. locate → busca archivos

vii. ls → lista archivos

```
agus@agus:~$ locate prueba.exe
/var/lib/locate/locate.db: No such file or directory
agus@agus:~$ ls -l
total 12
-rw-rw-r-- 1 agus agus 24 Aug 26 22:19 archivo2.txt
-rw-rw-r-- 1 agus agus  0 Aug 26 22:18 archivo.txt
-rw-rw-r-- 1 agus agus  0 Aug 26 22:18 Archivo.txt
-rw-rw-r-- 2 agus agus 28 Aug 26 22:23 enlace_duro
lrwxrwxrwx 1 agus agus 10 Aug 26 22:28 enlace_simbolico -> prueba.exe
-rw-rw-r-- 2 agus agus 28 Aug 26 22:23 prueba.exe
```

viii. pwd → muestra direccion del directorio actual

ix. cp → copia el primer archivo en el 2do que lo crea

```
cp prueba.exe copiaprueba.exe
```

x. mv → mueve o renombra (si le pasas direccion lo mueve, si le pasas un archivo lo copia)

```
mv prueba.exe carpeta1
```

```
mv copiaprueba.exe nuevonombre.exe
```

xi. find → *busqueda avanzada (find ~ -name "prueba.exe")*

10. Indique qué comando es necesario utilizar para realizar cada una de las siguientes acciones. Investigue su funcionamiento y parámetros más importantes:

a. Cree la carpeta ISOCSO

b. Acceda a la carpeta

c. Cree dos archivos con los nombres isocso.txt e isocso.csv

d. Liste el contenido del directorio actual

e. Visualizar la ruta donde estoy situado

f. Busque todos los archivos en los que su nombre contiene la cadena "iso*"

g. Informar la cantidad de espacio libre en disco

h. Verifique los usuarios conectados al sistema

i. Editar a el archivo isocso.txt e ingresar Nombre y Apellido

j. Mostrar en pantalla las últimas líneas de un archivo.

11. Investigue el funcionamiento, parámetros y ubicación (directorio) de los siguientes comandos:

→ man → dmesg → shutdown → lspci → reboot → at → halt → netstat →
uname → head → tail

12. Procesos:

**a. ¿Qué es un proceso? ¿A que hacen referencia las siglas PID y PPID?
¿Todos los procesos tienen estos atributos en GNU/Linux? Justifique.
Indique qué otros atributos tiene un proceso.**

b. Investigue el funcionamiento, parámetros y ubicación (directorio) de los siguientes comandos relacionados a procesos. En caso de que algún comando no venga por defecto en la distribución que utiliza deberá proceder a instalarlo:

**i.top vii. pkill ii.htop viii. killall iii.ps ix. renice iv.pstree x. xkillv.kill xi. atop
vi.pgrep xii. nice**

13. Proceso de Arranque SystemV (<https://github.com/systeminit/si>):

a. Enumere los pasos del proceso de inicio de un sistema GNU/Linux, desde que se prende la PC hasta que se logra obtener el login en el sistema.

b. Proceso INIT. ¿Quién lo ejecuta? ¿Cuál es su objetivo?

c. RunLevels. ¿Qué son? ¿Cuál es su objetivo?

**d. ¿A qué hace referencia cada nivel de ejecución según el estándar?
¿Dónde se define qué Runlevel ejecutar al iniciar el sistema operativo?
¿Todas las distribuciones respetan estos estándares?**

e. Archivo /etc/inittab. ¿Cuál es su finalidad? ¿Qué tipo de información se almacena en él? ¿Cuál es la estructura de la información que en él se almacena?

f. Suponga que se encuentra en el runlevel <X>. Indique qué comando(s) deberá ejecutar para cambiar al runlevel <Y>. ¿Este cambio es permanente? ¿Por qué?

g. Scripts RC. ¿Cuál es su finalidad? ¿Dónde se almacenan? Cuando un sistema GNU/Linux arranca o se detiene se ejecutan scripts, indique cómo determina qué script ejecutar ante cada acción. ¿Existe un orden para llamarlos? Justifique.

14. SystemD (<https://github.com/systemd/systemd>):

a. ¿Qué es systemd?

b. ¿A qué hace referencia el concepto de Unit en SystemD?

c. ¿Para qué sirve el comando systemctl en SystemD?

d. ¿A qué hace referencia el concepto de target en SystemD?

e. Ejecutar el comando pstree. ¿Qué es lo que se puede observar a partir de la ejecución de este comando?

15. Usuarios:

a. ¿Qué archivos son utilizados en un sistema GNU/Linux para guardar la información de los usuarios?

b. ¿A qué hacen referencia las siglas UID y GID? ¿Pueden coexistir UIDs iguales en un sistema GNU/Linux? Justifique.

c. ¿Qué es el usuario root? ¿Puede existir más de un usuario con este perfil en GNU/Linux? ¿Cuál es la UID de root?

d. Agregue un nuevo usuario llamado isocso a su instalación de GNU/Linux, especifique que su home sea creada en /home/isocso, y hágalo miembro del grupo informatica (si no existe, deberá crearlo). Luego, sin iniciar sesión como este usuario cree un archivo en su home personal que le pertenezca. Luego de todo esto, borre el usuario y verifique que no queden registros de él en los archivos de información de los usuarios y grupos.

e. Investigue la funcionalidad y parámetros de los siguientes comandos:

i. useradd y adduser v. groupadd ii. usermod vi. who iii. userdel vii. groupdel iv. su viii. passwd

16. FileSystem y permisos:

a. ¿Cómo son definidos los permisos sobre archivos en un sistema GNU/Linux?

b. Investigue la funcionalidad y parámetros de los siguientes comandos relacionados con los permisos en GNU/Linux:

i. chmod

ii. chown

iii. chgrp

c. Al utilizar el comando chmod generalmente se utiliza una notación octal asociada para definir permisos. ¿Qué significa esto? ¿A qué hace referencia cada valor?

d. ¿Existe la posibilidad de que algún usuario del sistema pueda acceder a determinado archivo para el cual no posee permisos? Indíquelo y realice las

pruebas correspondientes.

e. Explique los conceptos de "full path name" (path absoluto) y "relative path name" (path relativo). De ejemplos claros de cada uno de ellos.

f. ¿Con qué comando puede determinar en qué directorio se encuentra actualmente?

¿Existe alguna forma de ingresar a su directorio personal sin necesidad de escribir todo el path completo? ¿Podría utilizar la misma idea para acceder a otros directorios? ¿Cómo? Explique con un ejemplo.

g. Investigue la funcionalidad y parámetros de los siguientes comandos relacionados con el uso del FileSystem:

i. umount

ii. du

iii. df

iv. mount

v. mkfs

vi. fdisk (con cuidado)

vii. write

viii. losetup

ix. stat

17. Procesos:

a. ¿Qué significa que un proceso se está ejecutando en Background? ¿Y en Foreground? b. ¿Cómo puedo hacer para ejecutar un proceso en Background? ¿Como puedo hacer para pasar un proceso de background a foreground y viceversa?

c. Pipe (|). ¿Cuál es su finalidad? Cite ejemplos de su utilización.

d. Redirección. ¿Qué tipo de redirecciones existen? ¿Cuál es su finalidad? Cite ejemplos de utilización.

18. Otros comandos de Linux (Indique funcionalidad y parámetros):

a. ¿A qué hace referencia el concepto de empaquetar archivos en GNU/Linux?

b. Seleccione 4 archivos dentro de algún directorio al que tenga permiso y sume el tamaño de cada uno de estos archivos. Cree un archivo empaquetado conteniendo estos 4 archivos y compare los tamaños de los

mismos. ¿Qué característica nota? c. ¿Qué acciones debe llevar a cabo para comprimir 4 archivos en uno solo? Indique la secuencia de comandos ejecutados.

d. ¿Pueden comprimirse un conjunto de archivos utilizando un único comando?

e. Investigue la funcionalidad de los siguientes comandos:

i. tar

ii. grep

iii. gzip

iv. zgrep

v. wc

19. Indique qué acción realiza cada uno de los comandos indicados a continuación considerando su orden. Suponga que se ejecutan desde un usuario que no es root ni pertenece al grupo de root. (Asuma que se encuentra posicionado en el directorio de trabajo del usuario con el que se logueó). En caso de no poder ejecutarse el comando, indique la razón:

ls -l > prueba

ps > PRUEBA

chmod 710 prueba

chown root:root PRUEBA

chmod 777 PRUEBA

chmod 700 /etc/passwd

passwd root

rm PRUEBA

man /etc/shadow

find / -name *.conf

usermod root -d /home/newroot -L

cd / root

rm *

cd / etc

cp */home -R

shutdown

20. Indique qué comando sería necesario ejecutar para realizar cada una de las siguientes acciones:

a. Terminar el proceso con PID 23.

b. Terminar el proceso llamado init o systemd. ¿Qué resultados obtuvo?

c. Buscar todos los archivos de usuarios en los que su nombre contiene la cadena ".conf"

d. Guardar una lista de procesos en ejecución el archivo /home/<su nombre de usuario>/procesos

e. Cambiar los permisos del archivo /home/<su nombre de usuario>/xxxx a:

i. Usuario: Lectura, escritura, ejecución

ii. Grupo: Lectura, ejecución

iii. Otros: ejecución

f. Cambiar los permisos del archivo /home/<su nombre de usuario>/yyyy a:

i. Usuario: Lectura, escritura.

ii. Grupo: Lectura, ejecución

iii. Otros: Ninguno

g. Borrar todos los archivos del directorio /tmp

h. Cambiar el propietario del archivo /opt/isodata al usuario isocso

i. Guardar en el archivo /home/<su nombre de usuario>/donde el directorio donde me encuentro en este momento, en caso de que el archivo exista no se debe eliminar su contenido anterior.

21. Indique qué comando sería necesario ejecutar para realizar cada una de las siguientes acciones:

a. Ingrese al sistema como usuario "root"

b. Cree un usuario. Elija como nombre, por convención, la primera letra de su nombre seguida de su apellido. Asígnele una contraseña de acceso.

- c. ¿Qué archivos fueron modificados luego de crear el usuario y qué directorios se crearon?**
- d. Crear un directorio en /tmp llamado miCursada**
- e. Copiar todos los archivos de /var/log al directorio antes creado.**
- f. Para el directorio antes creado (y los archivos y subdirectorios contenidos en él) cambiar el propietario y grupo al usuario creado y grupo users.**
- g. Agregue permiso total al dueño, de escritura al grupo y escritura y ejecución a todos los demás usuarios para todos los archivos dentro de un directorio en forma recursiva.**
- h. Acceda a otra terminal para loguearse con el usuario antes creado.**
- i. Una vez logueado con el usuario antes creado, averigüe cuál es el nombre de su terminal.**
- j. Verifique la cantidad de procesos activos que hay en el sistema.**
- k. Verifique la cantidad de usuarios conectados al sistema.**
- l. Vuelva a la terminal del usuario root y envíele un mensaje al usuario anteriormente creado enviándole que el sistema va a ser apagado.**
- m. Apague el sistema.**

22. Indique qué comando sería necesario ejecutar para realizar cada una de las siguientes

acciones:

- n. Cree un directorio cuyo nombre sea su número de legajo e ingrese a él.**
- o. Cree un archivo utilizando el editor de textos vi, e introduzca su información personal: Nombre, Apellido, Número de alumno y dirección de correo electrónico. El archivo debe llamarse "LEAME".**
- p. Cambie los permisos del archivo LEAME, de manera que se puedan ver reflejados los siguientes permisos:**
 - Dueño: ningún permiso**
 - Grupo: permiso de ejecución**
 - Otros: todos los permisos**
- q. Vaya al directorio /etc y verifique su contenido. Cree un archivo dentro de su directorio personal cuyo nombre sea leame donde el contenido del mismo**

sea el listado de todos los archivos y directorios contenidos en /etc. ¿Cuál es la razón por la cuál puede crear este archivo si ya existe un archivo llamado "LEAME" en este directorio?

r. ¿Qué comando utilizaría y de qué manera si tuviera que localizar un archivo dentro del filesystem? ¿Y si tuviera que localizar varios archivos con características similares? Explique el concepto teórico y ejemplifique.

s. Utilizando los conceptos aprendidos en el punto anterior, busque todos los archivos cuya extensión sea .so y almacene el resultado de esta búsqueda en un archivo dentro del directorio creado en el primer inciso. El archivo deberá llamarse ejercicioF.

23. Indique qué acción realiza cada uno de los comandos indicados a continuación considerando su orden. Suponga que se ejecutan desde un usuario que no es root ni pertenece al grupo de root. (Asuma que se encuentra posicionado en el directorio de trabajo del usuario con el que se logueó). En caso de no poder ejecutarse el comando indique la razón:

01. mkdir iso

02. cd ./iso; ps > f0

03. ls > f1

04. cd /

05. echo \$HOME

06. ls -l \$> \$HOME/iso/ls

07. cd \$HOME; mkdir f2

08. ls -ld f2

09. chmod 341 f2

10. touch dir

11. cd f2

12. cd ~/iso

13. pwd > f3

14. ps | grep 'ps' | wc -l >> ../f2/f3

15. chmod 700 ../f2 ; cd ..

16. find . -name etc/passwd

17. find / -name etc/passwd

18. mkdir ejercicio5

19.

20.

a. Inicie 2 sesiones utilizando su nombre de usuario y contraseña. En una sesión vaya siguiendo paso a paso las órdenes que se encuentran escritas en el cuadro superior. En la otra sesión, cree utilizando algún editor de textos un archivo que se llame "explicacion_de_ejercicio" dentro del directorio creado en el ejercicio 22 y, para cada una de los comandos que ejecute en la otra sesión, realice una breve explicación de los resultados obtenidos.

b. Complete los comandos 19 y 20, de manera tal que realicen la siguiente acción:

19: Copiar el directorio iso y todo su contenido al directorio creado en 24.a

20: Copiar el resto de los archivos y directorios que se crearon en este ejercicio al directorio creado en el ejercicio 24.a

c. Ejecute las órdenes 19 y 20 y coméntelas en el archivo creado en el inciso a).

24. Cree una estructura desde el directorio /home que incluya varios directorios, subdirectorios y archivos, según el esquema siguiente.



Asuma que "usuario" indica cuál es su nombre de usuario. Además deberá tener en cuenta que dirX hace referencia a directorios y fX hace referencia a archivos. Utilizando la estructura de directorios anteriormente creada, indique qué comandos son necesarios para realizar las siguientes acciones:

a. Mueva el archivo "f3" al directorio de trabajo /home/usuario.

b. Copie el archivo "f4" en el directorio "dir11".

- c. Haga lo mismo que en el inciso anterior pero el archivo de destino, se debe llamar "f7".**
- d. Cree el directorio copia dentro del directorio usuario y copie en él, el contenido de "dir1".**
- e. Renombre el archivo "f1" por el nombre archivo y vea los permisos del mismo.**
- f. Cambie los permisos del archivo llamado archivo de manera de reflejar lo siguiente:**
 - Usuario: Permisos de lectura y escritura**
 - Grupo: Permisos de ejecución**
 - Otros: Todos los permisos**
- g. Renombre los archivos "f3" y "f4" de manera que se llamen "f3.exe" y "f4.exe" respectivamente.**
- h. Utilizando un único comando cambie los permisos de los dos archivos renombrados en el inciso anterior, de manera de reflejar lo siguiente:**
 - Usuario: Ningún permiso**
 - Grupo: Permisos de escritura**
 - Otros: Permisos de escritura y ejecución**

25. Indique qué comando/s es necesario para realizar cada una de las acciones de la siguiente secuencia de pasos (considerando su orden de aparición):

- a. Cree un directorio llamado logs en el directorio /tmp.**
- b. Copie todo el contenido del directorio /var/log en el directorio creado en el punto anterior.**
- c. Empaquete el directorio creado en a), el archivo resultante se debe llamar "misLogs.tar".**
- d. Empaquete y comprima el directorio creado en a), el archivo resultante se debe llamar "misLogs.tar.gz".**
- e. Copie los archivos creados en c) y d) al directorio de trabajo de su usuario.**
- f. Elimine el directorio creado en a), logs.**
- g. Desempaquete los archivos creados en c y d en 2 directorios diferentes.**

