**1. Editor de textos:**

**(a) Nombre al menos 3 editores de texto que puede utilizar desde la línea de comandos.**

Vim, Vi, Emacs, Nano

**(b) ¿En qué se diferencia un editor de texto de los comandos cat, more o less? Enumere los modos de operación que posee el editor de textos vi.**

**(c) Nombre los comandos más comunes que se le pueden enviar al editor de textos vi**

**2. Proceso de Arranque SystemV :**

**(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.**

Se empieza a ejecutar el código de la BIOS

El BIOS ejecuta el POST

El BIOS lee el sector de arranque (MBR)

Se carga el gestor de arranque (MBC)

El bootloader carga el kernel y el initrd

Se monta el initrd como sistema de archivos raíz y se inicializan componentes esenciales (ej:scheduler)

El kernel ejecuta el proceso init y se desmonta el initrd

Se lee el /etc/inittab

Se ejecutan los scripts apuntados por el runlevel 1

El final del runlevel 1 le indica que vaya al runlevel por defecto

Se ejecutan los scripts del runlevel por defecto

El sistema esta listo para usarse

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

Es ejecutado por el kernel. Su objetivo es el de cargar todos los subprocesos necesarios para el correcto funcionamiento del SO

Se encuentra en /sbin/init

No tiene padre y es el padre de todos los procesos

**(c) Ejecute el comando pstree. ¿Qué es lo que se puede observar a partir de la ejecución de este comando?**

Se observa el árbol con todos los procesos ejecutándose en ese momento

**(d) RunLevels. ¿Qué son? ¿Cuál es su objetivo?**

Son el modo en el que arranca Linux, se divide en niveles, cada Runlevel es responsable de levantar o bajar una serie de servicios

**(e) ¿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?**

Existen 7 según el estándar:

0: halt

1: single user mode

2: multiuser, without NFS (Sin soporte de red)

3: full multiuser mode console

4: no se usa

5: X11 (multiususario completo con login grafico basado en X)

6: reboot

**(f) Archivo /etc/inittab. ¿Cuál es su finalidad? ¿Qué tipo de información se almacena en el? ¿Cuál es la estructura de la información que en él se almacena?**

Su finalidad es la de definir los runlevels. En el se almacena la ID, el nivel de ejecución, la acción a realizar y el proceso que será ejecutado

**(g) Suponga que se encuentra en el runlevel X. Indique qué comando/s ejecutaría para cambiar al runlevel Y. ¿Este cambio es permanente? ¿Por qué?**

Init X (X = 0..6)

Depende de que runlevel ejecutes se puede apagar la maquina jaja

**(h) 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.**

Se almacenan en /etc/init.d

Para llamar a cada script se utiliza un link desde el /etc/rcX.d (X = 0..6)

[S/K]<orden><nombreScript> (S=Start, K=Kill)

**(i) ¿Qué es insserv? ¿Para qué se utiliza? ¿Qué ventajas provee respecto de un arranque tradicional?**

Se utiliza para administrar el orden de los enlaces simbólicos del /etc/rcX.d.

Usa cabeceras en los scripts del /etc/init.d que permiten especificar la relación con otros scripts rc

**(j) ¿Cómo maneja Upstart el proceso de arranque del sistema?**

En Upstart el proceso de arranque del sistema se maneja a través de Jobs, los cuales son eventos asincrónicos.

El principal objetivo de cada job es el de definir servicios o tareas a ser ejecutadas por init

Cada job esta definido en el /etc/init (.conf)

Hay dos tipos: Task (ejecución finita 🡪 not respawning 🡪 exit 0 o stop), Service (ejecución indeterminada 🡪 respawning)

Los Jobs se ejecutan ante eventos (arranque del equipo, inserción de un USB, etc)

Es compatible con SystemV

Cada job posee un objetivo (goal start/stop) y un estado (state)

No existe mas /etc/inittab

**(k) Cite las principales diferencias entre SystemV y Upstart.**

**(l) Qué reemplaza a los scripts rc de SystemV en Upstart? ¿En que ubicación del filesystem se encuentran?**

Los scripts rc se reemplazan por Jobs. Estos se encuentran definidos en el /etc/init

**(m) Dado el siguiente job de upstart perteneciente al servicio de base de datos del mysql indique a qué hace referencia cada línea del mismo:**

**# MySQL Service**

**description "MySQL Server"**

**author "info autor "**

**start on ( net−device−up**

**and local−file systems**

**and runlevel [ 2345 ] )**

**stop on runlevel [ 016 ]**

**[ . . . ]**

**exec /usr/sbin/mysqld**

**[ . . . ]**

**(n) ¿Qué es systemd?**

Es un sistema que centraliza la administración de demonios y librerías del sistema

Mejora el paralelismo de booteo. Puede ser controlado por systemctl

Compatible con SysV (si se lo llama init)

El demonio SystemD reemplaza a init

Los runlevels se reemplazan por targets

Tampoco existe el archivo /etc/inittab

Las unidades de trabajo se denominan units y existen diferentes tipos:

Service: controla un servicio particular (.service)

Socket: encapsula IPC, un socket del sistema o filesystem FIFO (.socket)

Target: agrupa units o establece puntos de sincronización durante el booteo (.target)

Snapshot: almacena el estado de un conjunto de unidades que puede ser reestablecido mas tarde (.snapshot)

Pueden tener dos estados (active o inactive)

**(ñ) ¿A qué hace referencia el concepto de activación de socket en systemd?**

Este es un mecanismo que se utiliza para iniciar los servicios por demanda, así no se activan de manera innecesaria.

Cuando un socket recibe una conexión spawnea al servicio y le pasa el socket

No se necesita definir dependencias entre servicios

**(o) ¿A qué hace referencia el concepto de cgroup?**

El concepto cgroup hace referencia a agrupar procesos relacionados entre si, para poder organizarlos de forma jerárquica.

Realiza Tracking (mediante el subsistema cgroups 🡪 no se utiliza el PID)

Limita el uso de recursos

**3. Usuarios:**

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

/etc/passw se usa para guardar configuración

/etc/group “ ”

/etc/shadow “ ”

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

UID Unique IDentifier

GID Group IDentifier

Si pueden tener dos o mas UIDs iguales

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

El usuario root es el administrador del sistema

No puede existir mas de un root por sistema

Su UID es 0

**(d) Agregue un nuevo usuario llamado iso2017 a su instalación de GNU/Linux, especifique que su home sea creada en /home/iso\_2017, y hágalo miembro del grupo catedra (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:**

**useradd ó adduser**

**usermod**

**userdel**

**su**

**groupadd**

**who**

**groupdel**

**passwd**

**4. FileSystem:**

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

Permisos de Read, Write y eXecute

Se aplican sobre dUeño, Grupo, Otros

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

**chmod**

**chown**

**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?**

rwx = 100 = 4 solo read, rwx = 010 = 2 solo write, rwx = 001 = 1 solo execute

primer numero de User, segundo de Group y tercero de Others

EJ: 754 user=7=111 todos, group=5=101 solo read y execute, others=4=100 solo read

**(d) ¿Existe la posibilidad de que algún usuario del sistema pueda acceder a determinado archivo para el cual no posee permisos? Nombrelo, y realice las pruebas correspondientes.**

**(e) Explique los conceptos de “full path name” y “relative path name”. De ejemplos claros de cada uno de ellos.**

Full name path o absolute name path es el nombre del archivo desde la raíz (/home/joaco/CSO2022/cso2022-1.txt)

Relative name path usa . para referirse al directorio actual .. para referirse al padre (cd ../..) cambia el directorio actual a dos directorios mas arribaq

**(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:**

**cd**

**umount**

**mkdir**

**du**

**rmdir**

**df**

**mount**

**ln**

**ls**

**pwd**

**cp**

**mv**

5. 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) Indique qué comandos se podrían utilizar para ver qué procesos están en ejecución en un sistema GNU/Linux.

(c) ¿Qué significa que un proceso se está ejecutando en Background? ¿Y en Foreground?

(d) ¿Cómo puedo hacer para ejecutar un proceso en Background? ¿Como puedo hacer para pasar un proceso de background a foreground y viceversa?

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

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

(g) Comando kill. ¿Cuál es su funcionalidad? Cite ejemplos.

(h) Investigue la funcionalidad y parámetros de los siguientes comandos relacionados con el manejo de procesos en GNU/Linux. Además, compárelos entre ellos:

ps

kill

pstree

killall

top

nice

6. 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:

tar

grep

gzip

zgrep

wc