



## SISTEMAS OPERATIVOS Y DISTRIBUIDOS

Trabajo Práctico N° 2

**Intérprete de comandos e Interacción con el Sistema Operativo**

Segundo Cuatrimestre de 2015

### Ejercicios

Resuelva los siguientes incisos utilizando el Sistema Operativo GNU/Linux.

1. Explique el concepto de *shell*. Utilice al menos dos formas alternativas para conocer el intérprete de comandos que está asociado a su usuario. ¿Existe alguna forma de modificar el intérprete de comandos utilizado?

**Pista:** explore la opción *intro* de las páginas de manual.

2. ¿Es posible que una cuenta existente en el sistema no posea un intérprete de comandos asociado? ¿Tendría sentido?
3. ¿Resulta importante contar con ambientes CLI dada la importancia de las interfaces gráficas de usuario en la actualidad?
4. Determine la versión del kernel que está utilizando y el tipo de arquitectura subyacente. ¿Posee soporte de virtualización a nivel de hardware?
5. En lo que respecta a memoria determine:
  - a) Los distintos niveles de *cache* que posee la arquitectura y el tamaño asociado a cada nivel.
  - b) Los módulos de memoria RAM instalados, detallando el tipo y frecuencia de operación.
6. Para cada uno de los siguientes casos, suponga que desea encontrar las páginas de manual (junto con una breve descripción) que posean información relacionada a:
  - a) Un comando que genera claves de autenticación para el servicio SSH, cuyo nombre comienza con las letras “*ssh*”.
  - b) Todas las ocurrencias de “*printf*”, sin importar si es un comando o una función de librería.
  - c) Todos los archivos de configuración que cuyo nombre finaliza en “*tab*”.
7. Muestre una forma directa de encontrar la ubicación del **binario** y la página de manual asociada al comando del ejercicio 6a.
8. Determine su directorio **home**, cree en el mismo un subdirectorío denominado “*tp2*” y resuelva los siguientes incisos dentro del mismo:

- a) Cree un archivo denominado `file1`, verifique que el tamaño del mismo corresponde a 0 bytes.
- b) Utilizando una sentencia desde el intérprete de comandos,
  - 1) Agregue al archivo `file1` la palabra “Hola”.
  - 2) Cree un archivo denominado `file2` con el contenido “Mundo”.
  - 3) Utilizando el comando `cat`, cree un archivo `file3` a partir de los contenidos de los archivos `file1` y `file2`.
- c) Utilizando los comandos `stat` y `grep`, almacene en un archivo de texto el nombre de cada archivo y su estampilla de tiempo de acceso.

**Pista:** considere la utilización del operador “pipe” y la redirección de la salida estándar.

- d) A partir del desarrollo del inciso anterior, cree un *script* denominado `almacenar_accesos.sh` de forma tal que el mismo durante su ejecución cree un archivo `taccesos.txt`, en el cual se almacene el identificador y tiempo de acceso de cada archivo considerado previamente. El nombre del archivo debe pasarse como parámetro al momento de invocar al *script*.
- e) Modifique el tiempo de acceso de `file1` de forma tal que coincida con el de `file3`.

**Pista:** busque en las páginas de manual una descripción de un **comando** para identificar estampillas de tiempo cuyo identificador comienza con la letra ‘t’.
- f) Cree el archivo `taccesos_new.txt` invocando nuevamente al *script*.
- g) Compruebe la actualización en la estampilla de tiempo observando las diferencias entre los archivos generados por el *script* mediante la utilización del comando `diff`.

9. ¿Cuál es la finalidad del directorio `/etc`? Listar su contenido cumpliendo con las siguientes restricciones:

- a) Ordenar el listado por fecha de modificación (primero los más recientes)
- b) Ordenar el listado por fecha de modificación (primero los más antiguos)
- c) Mostrar los tamaños de archivo en unidades amigables (KB, MB, GB)
- d) Listar primero los directorios y luego los archivos
- e) Ordenar por tamaño de archivo (de mayor a menor)
- f) Ordenar por tamaño de archivo (de menor a mayor)
- g) Crear un alias ‘l’ del comando `ls`, de forma tal que la invocación al mismo muestre una línea por cada ítem junto con los permisos, usuario, grupo, tamaño (en unidades amigables), y que además muestre primero los directorios y luego los archivos.

10. Buscar recursivamente dentro del directorio `/etc` todos los archivos cuyo nombre comienza con “net”.

11. Utilizando el intérprete de comandos determinar:

- a) La cantidad de elementos que posee un directorio sin tener en cuenta ‘.’ o ‘..’
- b) La cantidad de palabras, separadas por espacio, que contiene un archivo de texto

- c) El espacio en disco ocupado por el directorio `/home`
12. Cree un script en `BASH` denominado `“soyd.sh”`, de forma tal que el mismo se ejecute cada vez que el intérprete de comandos sea invocado. En cada invocación, el script debe crear y sobrescribir el archivo `/var/log/soyd` el cual debe contener la fecha y hora de la última ejecución del script.
13. Determine los alcances del usuario `root`. ¿Por qué es diferente del resto de los usuarios del sistema?
14. ¿Cuál es el objetivo de los comandos `su` y `sudo`?
15. Realice la siguiente gestión de usuarios:
- a) Crear el usuario `“pepe”` cuyo directorio `$HOME` sea `/home/pepe`
  - b) Asignar una nueva contraseña al usuario `“pepe”`
  - c) Agregar al usuario `“pepe”` al grupo `wheel`
  - d) Cambiar el nombre del usuario `“pepe”` a `“pedro”`
  - e) Determinar a qué grupos pertenece el usuario `“pedro”`
  - f) Permitir el uso del comando `sudo` al usuario `“pedro”`
  - g) Denegar el uso del comando `sudo` al usuario `“pedro”`
  - h) Eliminar al usuario `“pedro”` junto con su directorio `$HOME`
16. Listar los últimos comandos ejecutados en la sesión actual.
17. Administración y gestión de procesos.
- a) Determinar el `userid` efectivo de la sesión actual
  - b) Ejecutar en segundo plano el comando `$ping 200.49.226.11`
  - c) Ejecutar el comando `top`, una vez iniciado enviarlo a segundo plano
  - d) Mostrar todos los procesos que se están ejecutando con el mismo `userid` efectivo que la sesión actual
  - e) Mostrar todos los procesos ejecutando como usuario `root` (ID efectivo y real)
  - f) Determinar el PID del proceso que se encuentra ejecutando el comando `$ping 200.49.226.11`
  - g) Eliminar el proceso que se encuentra ejecutando el comando `$ping 200.49.226.11`
  - h) Determinar qué proceso está consumiendo más ciclos de CPU
  - i) Determinar qué proceso está consumiendo más memoria RAM
  - j) Determinar la cantidad de memoria RAM disponible
18. Liste todos los procesos que poseen archivos abiertos en el directorio `/tmp`. Repita el procedimiento para el directorio `/var/log`, considerando esta vez todos sus subdirectorios.
19. ¿Cuál es la relación que existe entre un comando y un proceso?
20. Explique el concepto de *información de estado*. ¿Existe algún mecanismo para obtenerla desde el intérprete de comandos?