



Facultad de Ingeniería

# GUÍA PRÁCTICA DE LABORATORIO N°2 GNU-LINUX



## Consignas:

1. Inicie sesión con el comando `ssh` en el servidor LINUX cuya dirección IP es 192.168.1.105 y cuyo hostname es Pitagoras2.

```
$ ssh usuario@hostname  
$ ssh usuario@ip
```

2. Tipee **`ls`** | **`tee cosas2`**. ¿Obtuvo la lista de sus archivos por pantalla? Ahora tiene un nuevo archivo llamado cosas2. Liste su contenido tipeando **`cat cosas2`**. ¿Es esta la misma lista? El comando **`tee`** le da la posibilidad de copiar la entrada estándar de un comando a un archivo y así mismo seguir teniendo salida estándar por pantalla. Nos da la posibilidad de obtener salida doble.

3. Supongamos que Ud. necesite un archivo que contenga una lista de todos sus archivos por orden alfabético, pero al revés (o sea, comenzando por la z) y además, la cantidad de archivos. Podría utilizar la **opción -r**

---

del comando **ls**, pero en este caso utilice el comando **sort**, para familiarizarse con más comandos. Tipee **ls |sort -r |tee cosas2 |wc -w**. ¿Qué número obtuvo en la pantalla? Tipee **ls**. Es este el número de archivos que Ud. tiene?

4. Tipee **cat cosas2**. ¿Cuál es el orden del contenido? Cuando hicimos un **pipe** hacia el comando **sort**, el standard input del **sort** fue el output del comando anterior (**ls**), y el standard output del **sort** va a ser el standard input del comando **tee** (por medio del **pipe**).

5. El comando **tee** creó dos copias del archivo, una la almacenó en el archivo **cosas2**, y la otra la mandó como standard input al comando **wc**, para que este la contara.

6. Tipee "**who**" para saber cuál es el nombre del archivo especial que representa su terminal. La respuesta será su **userid** seguido del nombre del archivo. Este archivo se encuentra en el directorio **/dev** y es un archivo especial para direccionar los datos desde y hacia su terminal. Este archivo se convierte en el standard input y standard output para sus filtros, siempre que haya iniciado sesión (salvo que Ud. haya cambiado sus standard in/out). Dado que es un archivo, Ud. puede redireccionar datos hacia él. Así es como obtendremos la lista por pantalla.

7. Ahora tipee **ls |tee /dev/ptsX |wc -w** sustituyendo **ptsX** por el nombre de su archivo de terminal ¿Cuál es la respuesta?

8. Tipee **cat cosas | sed -e 's/Linux/Windows/g' > otrascosas**.

9. Revise el archivo **otrascosas**. **sed** viene de stream editor, es un editor de flujos. Lee de la entrada estandar y escribe en la salida estandar. Aquí lo hemos usado para buscar una expresión y reemplazarla por otra. Su forma es **sed [opciones] 's/<expresión-buscada>/<expresión-reemplazante>/<opciones de s>'**. Encontrará otros usos de **sed** en la página 79 de **rute.pdf**

10. Cada vez que Ud. ingresa algo al sistema, el sistema primero le tuvo que haber dado el prompt **\$**. Si Ud. quisiera cambiar el prompt por cualquier otra cosa (su nombre por ej.), debería cambiar una variable del sistema.

- 
11. Típe `set` (enter). Este comando nos da la lista de variables del sistema. Observe las variables `PS1` y `PS2`. Estos son los valores de los prompts. Típe `PS1 =+`  (enter). ¿Qué obtiene ahora como prompt?
  12. Cambie nuevamente el prompt ahora por su nombre. Típe `PS1=nombre` (enter). Si su nombre es compuesto, debe ponerlo entre comillas, si es solo una palabra, no es necesario.
  13. Típe `echo hola \` (enter). ¿Cuál es el prompt a continuación? Típe `a todos` (enter) para completar esta línea y obtener el primer prompt.
  14. Típe `PS2=2` (enter). Ahora típe `cat \` (enter). ¿Qué es lo que obtuvo como segundo prompt?  
Típe cualquier cosa y de enter antes de continuar.
  15. Típe `echo $HOME` (enter) ¿Qué sucedió? El uso del metacarácter `$` le permite hacer referencia a un nombre del sistema y, de esta manera, ver su contenido por medio del comando `echo`. El contenido de la variable `HOME` es su directorio de login.
  16. Típe `cd /; pwd`. ¿En qué directorio está Ud. ahora?
  17. Típe `cd;pwd` (enter). ¿En qué directorio está Ud ahora? ¿Es el mismo nombre que obtuvo al ver el contenido de la variable `HOME`? El comando `cd`, sin ninguna especificación de path, lo lleva a su directorio home.
  18. Típe `HOME=/bin` (enter). Ahora Ud. cambió su directorio default a `/bin`, aunque su directorio actual siga siendo el mismo.
  19. Típe `cd` (enter), luego típe `pwd` (enter). ¿En qué directorio está y por qué?
  20. Salga del sistema y luego vuelva a entrar. ¿Cuáles son sus prompts primario y secundario?.
  21. Típe `cd;pwd` (enter). ¿En qué directorio está ahora y por qué? Cuando Ud. hizo `logout` y luego volvió a ingresar, las variables del sistema se resetearon. Estas pueden ser cambiadas en forma permanente si esto se realiza en el archivo `.profile`.

---

22. Tipee `ps` (enter) para ver los procesos activos. Verá solamente los que le pertenecen.

23. Si usa `ps -e` verá los demás procesos que están en el sistema.

24. Cree un pequeño script llamado `proc.sh` “`vim proc.sh`” con las siguientes líneas:

```
#!/bin/sh
echo "proc.sh se está ejecutando"
sleep 1000
```

25. Ahora ejecútelo, para ello deberá fijarse si tiene permiso de ejecución. Para darle permiso deberá escribir “`chmod +x proc.sh`” y para ejecutarlo puede escribir “`./proc.sh`”.

26. No verá el prompt del shell porque se está ejecutando `proc.sh`. Para suspenderlo presione `ctrl-z`. ¿Qué pasa cuando ejecuta `ps`, en qué estado está `proc.sh`?

27. Escriba `bg` para mandar el proceso a background. El script ahora está activado y se ejecutará, sin embargo, usted dispone del prompt para ejecutar otra tarea. El programa está en background, para volver ejecute `fg` (foreground). Presione `ctrl-c` para terminarlo.

28. Pruebe a ejecutarlo con la siguiente orden `./proc.sh &`. ¿Qué pasó? El símbolo `&` sirve para ejecutar procesos en background.

29. El comando de terminación de un proceso es `kill`. En realidad `kill` envía una señal de terminación al proceso. Para terminar un proceso incondicionalmente la señal que debe usar es `SIGKILL` o su equivalente numérico 9. Por lo tanto `kill -9 <número del proceso a matar>` es el último recurso que usted debería usar.

30. Desconéctese del sistema, usando `exit` o `ctrl-d`. Esto no significa shutdown, aunque sea root y lo haga desde la consola, ya que Ud. verá aparecer el prompt del login (si está usando putty también cerrará ese programa y volverá a windows).