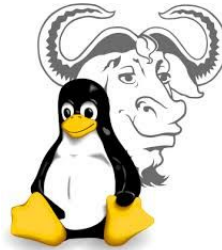


GUÍA PRÁCTICA DE LABORATORIO N°1

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
```

- El nombre de usuario para este curso es el usuario de las cátedras virtuales/correo electrónico.
- La contraseña por el momento es igual a su nombre de usuario y su legajo. Ej pepe.honguito1001.
- Después de comprobar el Id de usuario y la contraseña, Linux visualiza algunos mensajes y coloca el indicador del shell (shell prompt) para que usted ingrese los comandos u órdenes que desee.
- Normalmente este indicador está visualizado con el signo \$ o %.

2. Como medida de seguridad, cambie su contraseña. Ingrese el comando **passwd** (enter) y siga las instrucciones.

3. Después de haber cambiado satisfactoriamente su contraseña, desconéctese del sistema (usando el comando **exit** o **ctrl-d**) y vuelva a conectarse utilizando la nueva clave.

4. Utilice el comando **who** para determinar quién está conectado al sistema.

8. Ingrese la orden **date** para ver la fecha y la hora actuales.

9. Utilice el comando **cal** para marcar el día en que comenzaron las clases del primer semestre de este año.



10. Ubique el día del cumpleaños de alguno de los integrantes de su grupo en el año en que esperan recibirse utilizando el comando **cal**.

11. Utilice el comando **ls** para listar archivos. Puede ser que **ls** no haga nada y vuelva al shell. Esto es porque probablemente usted no tenga archivos todavía. Muchos comandos UNIX no dan ningún tipo de mensajes a no ser que haya algún error. Si hay archivos debería ver los nombres listados sin otra indicación.

12. Como usted sabe los archivos se organizan en directorios. Para crear un directorio utilice el comando **mkdir**. Por ejemplo, escriba "**mkdir sistemas_operativos**". Ahora escriba **ls** y verá "saludo" y sistemas_operativos.

13. Para ir al directorio sistemas_operativos debe usar el comando **cd** (change directory) y el argumento será el nombre del directorio. En este ejemplo "**cd sistemas_operativos**". Como este directorio recién ha sido creado no debe tener ningún archivo. Compruébelo con **ls**. Cree otro archivo prueba2.

14. Para volver al directorio anterior escriba "**cd ..**", esto significa el directorio anterior o padre. Un solo punto significa el directorio donde se encuentra.

15. Use el comando **ls** con la opción de formato largo **l**, o sea "**ls -l**". Aquí puede ver muchos detalles acerca de los archivos. Algunos de ellos son el nombre del archivo a la derecha, el tamaño del archivo en la quinta columna desde la izquierda, el tipo de archivo a la izquierda. El tipo puede ser **-** que significa archivo común o **d** que significa directorio.

16. Vaya a los distintos directorios usando la orden **cd**. El directorio raíz se simboliza con **/**. Pruebe a ir al directorio raíz usando "**cd /**". Mire los archivos o directorios que se encuentran ahí o en los otros directorios. Vuelva a su directorio (home directory) con la orden **cd** sin argumentos.

17. Verifique en el directorio en el que se encuentra utilizando el comando **pwd** (print working directory).

18. Cuando un directorio tiene demasiados archivos que no los puede ver, puede recurrir a la orden "**ls -l | less**". Luego use AvPag o RePag y presione **q** cuando termine).

19. Otra variación posible para el comando **ls** es la opción **a** (all) y sirve para listar todos los archivos incluidos los ocultos. La estructura de los comandos en Unix o Linux es:
Las opciones se pueden colocar de cualquier manera. Por ejemplo "**ls -a -l**" es igual a **ls -l -a** y es igual a "**ls -al**". Este comando mostrará todos los archivos (inclusive los ocultos) con el formato largo.

20. Escriba cualquier comando con la opción **--help** y obtendrá un resumen de su uso. Es una pequeña ayuda que resumirá opciones que puede haber olvidado si usted está familiarizado con ese comando.

21. Para una descripción exhaustiva del uso de un comando le conviene usar las páginas del manual del sistema mediante el comando **man**. Pruebe "**man ls**". Para salir use **q**.



22. Otra forma de crear archivos es con los editores. El editor que va a encontrar en cualquier versión de Unix o Linux es el **vi**. Cree un archivo llamado **prueba2** utilizando la orden "**vim prueba2**". El editor abre un buffer vacío. Si lo que usted ve por pantalla no es un archivo vacío, eso quiere decir que el archivo existe y Ud verá las líneas que lo componen. En este caso, salga del archivo ingresando **:q!**.

23. Borre el archivo usando el comando **rm** (**rm prueba2**) y vuelva a crear el archivo.

24. El editor **vim** tiene dos modos, el modo comando y el modo texto. Para cambiar a modo texto puede utilizar **i**. Ingrese el siguiente texto:

La historia de vim

El editor vi se desarrolló en la Universidad de California, Berkeley

Como parte del sistema unix de Berkeley.

25. Para salir del modo texto y volver al modo comando presione la tecla **ESC**. Una vez en modo comando grabe el archivo y salga del editor con los comandos **:wq**

26. Use el **vim** para corregir los errores y vuélvalo a guardar.

27. Vea el contenido del archivo **prueba2** mediante el uso de la orden **cat**.

28. Los comodines generan nombre de archivos que coinciden con máscaras. Los comodines pueden ser utilizados en cualquier comando que procese listas de nombres de archivos.

29. Ud ha listado el contenido de su directorio de trabajo varias veces, utilizando el comando **ls**.

Ahora tipee **ls *** (enter). ¿Cuál es la diferencia?

El comando **ls** sin ningún argumento lista el nombre de los archivos y subdirectorios del directorio actual. Sin embargo, si Ud. ingresa **ls ***, el Shell expande el ***** al contenido de todos los subdirectorios y lo pasa al comando **ls**. El efecto es mostrar por pantalla no sólo los nombres de los archivos y subdirectorios, sino también el contenido de los subdirectorios del directorio actual.

30. Tipee **cd /bin** (enter), y luego **ls** (enter). Ud encontrará este output bastante extenso.

31. Ahora liste sólo los archivos que comiencen con la letra **a**. Tipee "**ls a***".

El **a*** le dice al Shell que se fije en los archivos que comiencen con la letra **a** seguida de cualquier número de caracteres.

32. Ahora queremos listar todos los archivos que tengan un nombre de dos caracteres.

Para hacer esto, necesitamos otro comodín: el **?**.



Este comodín sustituye cualquier carácter, pero por uno solo. De esta manera, ingresando **ls a?** obtendremos una lista de todos los archivos cuyos nombres son de dos caracteres y comiencen con la letra **a**. Ahora vamos a obtener una lista de todos los archivos cuyos nombres sean de dos letras. Ingrese **ls ??**.

33. Ahora vamos a listar todos los archivos cuyos nombres comiencen con las letras **a**, **b** o **c**.

Para hacer esto, ingrese **ls [abc]***. Esto nos lista todos los archivos que comiencen con **a**, **b** o **c**.

34. Ahora ingrese **ls [a-c]*** ¿Nos da la misma lista?

35. Tipee **ls [a-cm]*** ¿Qué sucedió?

36. Tipee **ls [a-cm- r]*** ¿Cuál fue el resultado ahora?

37. Tipee **ls ?[a-c]?** ¿Qué es único respecto de los nombres de estos archivos?.

38. Para listar todos los archivos excepto los que comiencen con las letras del intervalo **c-t**, tipee **ls [!c-t]***.

39. Ahora tipee **ls [!c-t!w]*** (enter) ¿Qué cambia entre esta lista y la anterior?

40. Tipee el comando **cat** y de enter.

Ahora tipee algunas líneas con las frases que Ud. desee.

Ahora pulse las teclas **Ctrl-d** . Las líneas que usted tipeó ¿se volvieron a mostrar en pantalla?. Lo que hicimos fue utilizar el comando **cat** sin especificar el nombre de ningún archivo.

Dado que **cat** es un filtro, su input por default es el standard input, o sea el teclado. Esto significa que cualquier cosa que Ud. tipee es aceptada como input para **cat**. Como tampoco le especificamos un output a **cat**, este va a tomar el standard output, que es la pantalla. Tipee **Ctrl-d** para volver a tener el prompt.

Ahora vamos a redireccionar el output hacia un archivo. O sea, que el standard output ahora será un archivo en lugar de ser la pantalla.

41. Vuelva a su directorio home. Tipee **cat > cosas**.

Ahora vuelva a tipear algunas líneas, las que Ud. desee. Cuando haya terminado pulse enter seguido de **Ctrl-d**.

Ahora tipee **cat cosas**. ¿El archivo contiene lo que Ud. tipeó?

42. Vuelva a tipear el comando **cat > cosas**, pero esta vez tipee como contenido:



“El sistema operativo que más me gusta, más copado y más lindo del mundo es Linux (enter), Ctrl-d.

Vuelva a ver por pantalla el contenido de cosas tipeando `cat cosas`. ¿Qué sucedió?

Si el standard output es redireccionado a un archivo que ya existe, el contenido del archivo será reemplazado, sin ninguna advertencia.

43. La forma de agregar a un archivo existente es utilizando la función `>>`.

44. Tipee `cat >> cosas` (enter) y luego: Ustedes están de acuerdo? (enter), **Ctrl-d**.

45. Ahora tipee `cat cosas` (enter) ¿Cuál es el contenido del archivo ahora?

46. Tipee `cat cosas3` (enter) ¿Qué pasó? Obtuvo un error porque el archivo no existe.

47. Pruebe `cat cosas3 > errores`.

¿Esperaba usted ver el error en pantalla?

Tipee `ls -l errores` y verá que está vacío.

En realidad los errores usan otro flujo llamado `stderr` que por defecto también se escribe en la pantalla. Cuando nosotros redireccionamos con el `>` sin aclarar nada, en realidad lo único que redireccionamos es la salida. Para redireccionar los errores tenemos que especificar su descriptor de archivo que es el **2**.

48. Pruebe ahora `cat cosas3 2> errores` y vea qué pasa.

49. Ahora tipee `ls` para obtener la lista de archivos de su directorio de trabajo. ¿cuántos hay?

50. Ahora tipee `ls > temp` (enter)

luego `wc -w temp` (enter).

¿Es la misma cantidad de archivos? (debería ser de uno más, debido al archivo `temp` recién creado).

Tipee `rm temp` (enter) para borrar el archivo `temp`.

51. Ahora tipee `ls | wc -c` (enter)

¿Es la misma cantidad que la del punto 50?

Hicimos esto tomando el output del comando `ls` y, por medio del pipe, lo mandamos como standard input al comando `wc` y, de esta forma, eliminamos el paso intermedio de crear un archivo temporario (en nuestro caso `temp`) que luego tendría que ser borrado.