1. Todavia no lo hacemos
2. ls lista, como siempre. Tee copia la entrada estandar a un archivo y seguir mostrando la entrada en pantalla. Con ls | tee <nombre\_archivo> guarda todos los archivos y directorios listados en un archivo.
3. La opcion -r ordena la salida en orden alfabetico inverso, el comando sort sirve para ordenar.
4. Analizamos el funcionamiento del pipe.
5. Mas analisis.
6. Who sirve para ver el nombre del archivo especial que representa mi terminal. Es el standard in/out de todos mis comandos y esta en el directorio /dev.
7. La respuesta a “ls |tee /dev/ptsX |wc -w” fue:

tee: /dev/lucas: Permission denied

18

1. Sed viene de strem editor (editor de flujos). Este lee la entrada estandar y escribe en la salida estandar. Se usa de la siguiente manera
2. sed [opciones] ‘s/<expresión-buscada>/<expresión-reemplazante>/<opciones de s>’
3. Cada vez que se ingresa algo al sistema, primero el sistema le tiene que dar el prompt $. Si queres cambiar el prompt hay que cambiar la variable del sistema.
4. Con vim .bashrc podemos entrar a modificar el prompt. Si ponemos PS1 = “algo” cambiamos el nombre del prompt.
5. Lo mismo
6. provamos el prompt
7. cambiamos el segundo prompt
8. 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.
9. Con cd /; pwd estoy en el directorio raiz
10. con cd;pwd estoy en el directorio /home/lucas. Es decir, mi directorio HOME
11. HOME=/bin cambio mi directorio home a /bin
12. con cd y pwd descubro que el cambio es oficial.
13. Saliendo y volivendo a entrar me doy cuenta que todos mis cambios ahora ya no estan.
14. Tampoco se guardo el nuevo directorio HOME
15. ps muestra pocos procesos, solo los que me pertenecen.
16. Con ps -e veo todos los procesos del sistema.
17. Creamos el script proc.sh como un archivo de texto cualquiera. Le dimos permiso de ejecucion con “chmod +x proc.sh” y lo ejecutamos haciendo “./proc.sh”
18. lo anterior
19. con ctrl-z se para el proceso pero no se termina.
20. Con bg lo podes mandar al fondo y usar la terminal sin que joda. Con fg lo traes de vuelta y con ctrl-c lo terminas.
21. Con el simbolo & ejecutas las cosas en el background. ./proc.sh &
22. kill <numero de proceso> para matar procesos. Ultimo recurso kill -9 <PID>
23. con ctrl-d hace un shutdown.