

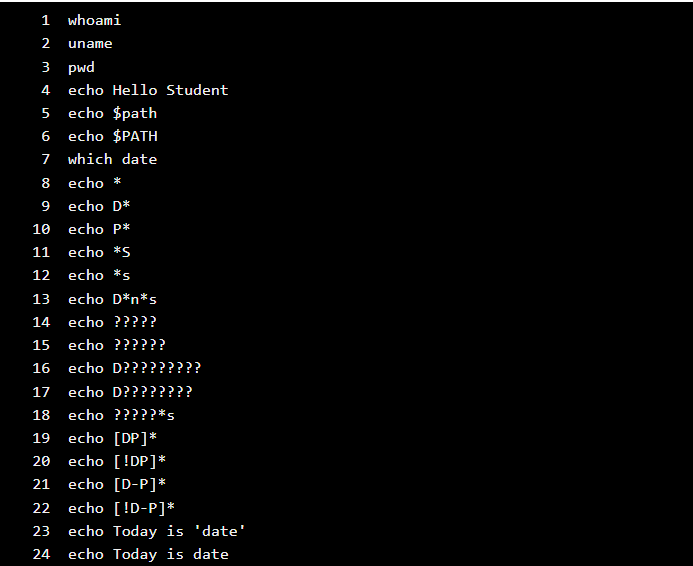
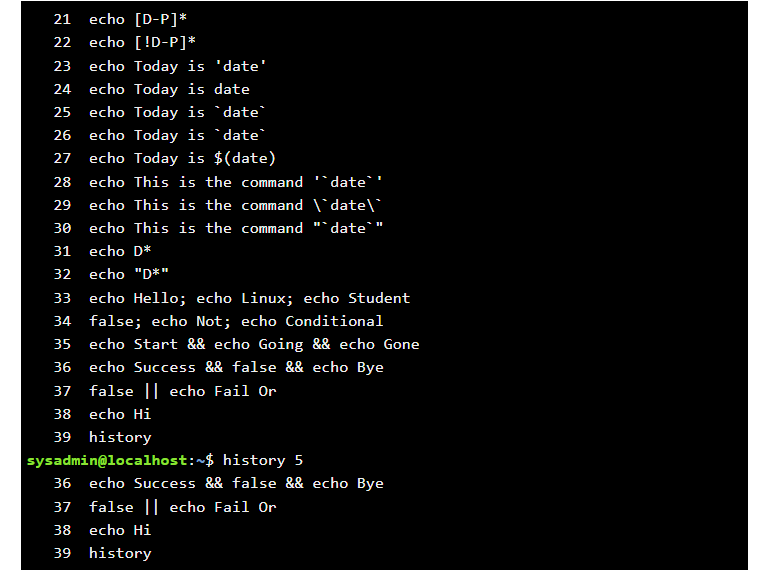
Reporte de comandos en linux

231011 Kevin Antonio Andrade Lopez



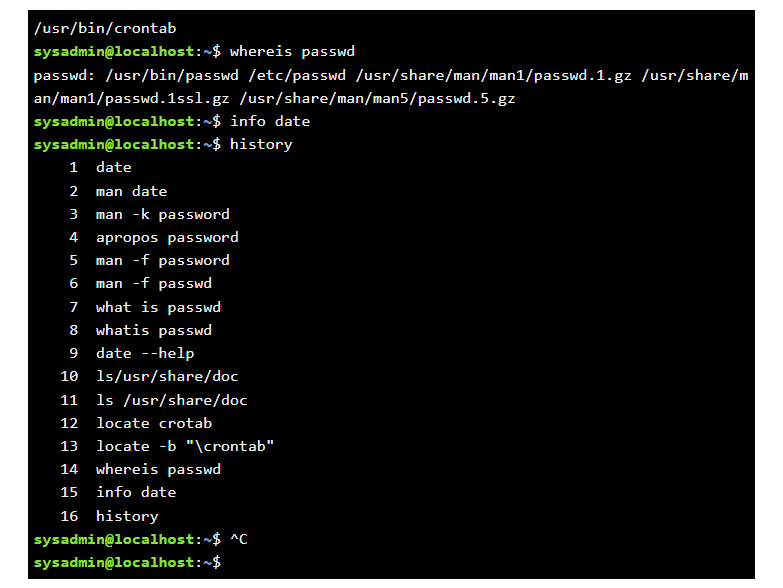
universidad tecnologica de aguascalientes

Capitulo 4:



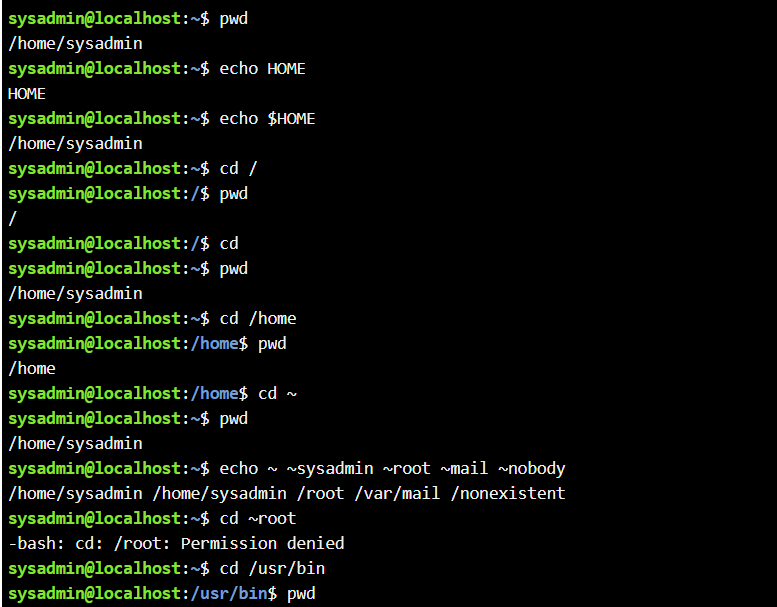
1. whoami: Muestra el nombre de usuario del usuario actual.
2. uname: Muestra información del sistema.
3. pwd: Imprime el directorio de trabajo actual.
4. echo Hello Student: Imprime el texto “Hello Student”.
5. echo $path: Intenta mostrar una variable llamada ‘path’, pero parece incorrecto ya que las variables distinguen entre mayúsculas y minúsculas y probablemente debería ser $PATH.
6. echo $PATH: Muestra el valor de la variable de entorno $PATH, que almacena las rutas a los directorios donde se buscan los archivos ejecutables.
7. which date: Busca el archivo ejecutable para el comando ‘date’ en los directorios especificados en la variable de entorno $PATH.
8. echo \*: Muestra todos los archivos en el directorio actual.
9. echo D\*: Muestra todos los archivos en el directorio actual que comienzan con ‘D’.
10. echo P\*: Muestra todos los archivos en el directorio actual que comienzan con ‘P’.
11. echo \*S: Muestra todos los archivos en el directorio actual que terminan con ‘S’.
12. echo \*s: Muestra todos los archivos en el directorio actual que terminan con ‘s’.
13. echo D\*n\*s: Muestra todos los archivos en el directorio actual que comienzan con ‘D’, tienen ‘n’ en algún lugar del medio, y terminan con ‘s’.
14. echo ?????: Muestra todos los archivos en el directorio actual que tienen exactamente 5 caracteres.
15. echo ??????: Muestra todos los archivos en el directorio actual que tienen exactamente 6 caracteres.
16. echo D ?????????: Muestra todos los archivos en el directorio actual que comienzan con ‘D’ y tienen exactamente 10 caracteres.
17. echo D ????????: Muestra todos los archivos en el directorio actual que comienzan con ‘D’ y tienen exactamente 9 caracteres.
18. echo ?????\*s: Muestra los archivos con 5 caracteres y que terminen en S-
19. echo [DP]\*: Muestra todos los archivos en el directorio actual que comienzan con ‘D’ o ‘P’.
20. echo [!DP]\*: Muestra todos los archivos en el directorio actual que no comienzan con ‘D’ ni ‘P’.
21. echo [D-P]\*: Muestra todos los archivos en el directorio actual que comienzan con cualquier letra entre ‘D’ y ‘P’ (inclusive).
22. echo [!D-P]\*: Muestra todos los archivos en el directorio actual que no comienzan con ninguna letra entre ‘D’ y ‘P’.
23. echo "Today is $(date)" : Imprime el texto “Today is” seguido de la fecha actual.
24. echo Today is date: Imprime el texto “Today is date”.
25. echo [D-P]\*: Muestra todos los archivos en el directorio actual que comienzan con cualquier letra entre ‘D’ y ‘P’ (inclusive).
26. echo [!D-P]\*: Muestra todos los archivos en el directorio actual que no comienzan con ninguna letra entre ‘D’ y ‘P’.
27. echo Today is 'date': Imprime el texto “Today is” seguido de la fecha actual. Sin embargo, parece que hay un error. Debería ser echo "Today is $(date)" para que funcione correctamente.
28. echo Today is date: Imprime el texto “Today is date”.
29. echo Today is date: Imprime el texto “Today is date”.
30. echo Today is date: Imprime el texto “Today is date”.
31. echo Today is $(date): Imprime el texto “Today is” seguido de la fecha actual.
32. echo This is the command 'date'': Imprime el texto “This is the command ‘date’'”. Parece que hay un error con las comillas al final.
33. echo This is the command \'date\': Imprime el texto “This is the command ‘date’”.
34. echo This is the command ""date ": Imprime el texto “This is the command "date "”. Parece que hay un error con las comillas dobles alrededor de “date”.
35. echo D\*: Muestra todos los archivos en el directorio actual que comienzan con ‘D’.
36. echo "D\*": Imprime el texto “D\*”. Las comillas hacen que se trate como una cadena literal en lugar de un comodín.
37. echo Hello; echo Linux; echo Student: Imprime tres líneas de texto: “Hello”, “Linux”, y “Student”. El punto y coma (;) se usa para separar múltiples comandos en una sola línea.
38. false; echo Not; echo Conditional: Ejecuta tres comandos en secuencia. false es un comando que siempre devuelve un estado de salida no exitoso. Luego imprime “Not” y “Conditional” en líneas separadas.
39. echo Start && echo Going && echo Gone: Ejecuta tres comandos en secuencia, pero solo si el comando anterior fue exitoso. Imprime “Start”, “Going”, y “Gone” en líneas separadas.
40. echo Success && false && echo Bye: Intenta ejecutar tres comandos en secuencia, pero solo si el comando anterior fue exitoso. Sin embargo, false siempre devuelve un estado de salida no exitoso, por lo que “Bye” no se imprimirá.
41. false || echo Fail Or: Ejecuta el comando false, y si falla (lo cual siempre hace), entonces ejecuta echo Fail Or.
42. echo Hi: Imprime el texto “Hi”.
43. history: Muestra el historial de comandos.
44. sysadmin@localhost: ~ $ history 5: Muestra los últimos 5 comandos del historial.
45. 36 echo Success && false && echo Bye: Este es el comando número 36 del historial, que intenta ejecutar tres comandos en secuencia, pero solo si el comando anterior fue exitoso. Sin embargo, false siempre devuelve un estado de salida no exitoso, por lo que “Bye” no se imprimirá.
46. 37 false || echo Fail Or: Este es el comando número 37 del historial, que ejecuta el comando false, y si falla (lo cual siempre hace), entonces ejecuta echo Fail Or.
47. 38 echo Hi: Este es el comando número 38 del historial, que imprime el texto “Hi”.
48. 39 history: Este es el comando número 39 del historial, que muestra el historial de comandos.

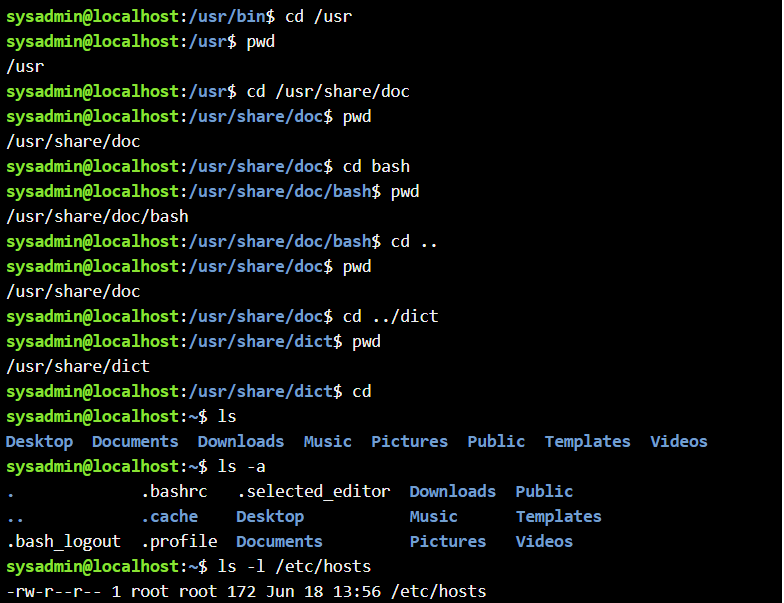
# Capítulo 5:

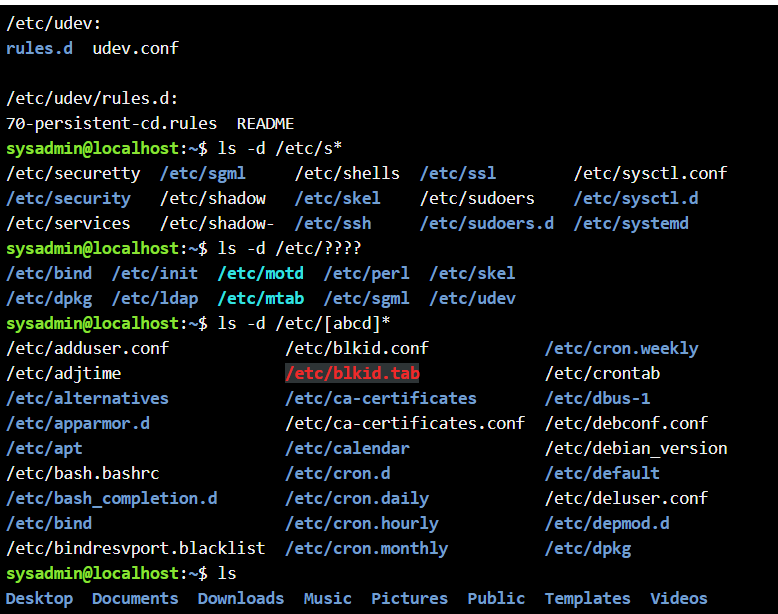


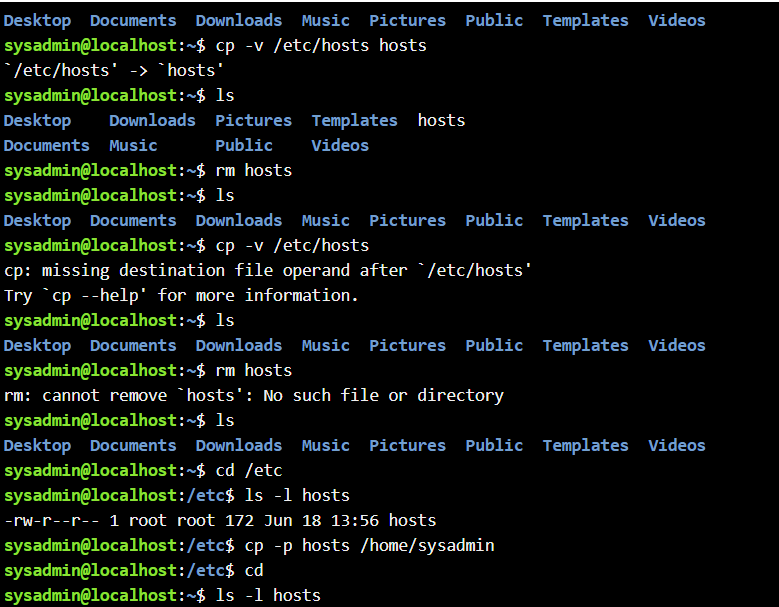
1. date: Este comando muestra la fecha y hora actuales.
2. man date: Este comando muestra la página de manual para el comando date.
3. man -k password: Este comando busca en las páginas de manual todos los comandos relacionados con la palabra clave “password”.
4. apropos password: Este comando es equivalente a man -k. Busca en las páginas de manual todos los comandos relacionados con la palabra clave “password”.
5. man -f password: Este comando muestra una breve descripción del comando password tomada de la página de manual.
6. whatis passwd: Este comando muestra una breve descripción del comando passwd tomada de la página de manual.
7. man -f passwd: Este comando muestra una breve descripción del comando passwd tomada de la página de manual.
8. whatis passwd: Este comando muestra una breve descripción del comando passwD tomada de la página de manual.
9. date --help: Este comando muestra la ayuda para el comando date.
10. ls /usr/share/doc: Este comando lista los archivos en el directorio /usr/share/doc.
11. locate share: Este comando busca en el sistema archivos que contienen la palabra “share”.
12. whereis passwd: Este comando busca el archivo ejecutable y las páginas de manual para el comando passwd en los directorios especificados en la variable de entorno $PATH.
13. info date: Este comando muestra la página de información para el comando date.
14. whereis date: Este comando busca el archivo ejecutable y las páginas de manual para el comando date en los directorios especificados en la variable de entorno $PATH.
15. history: Este comando muestra el historial de comandos que se han introducido en la terminal.

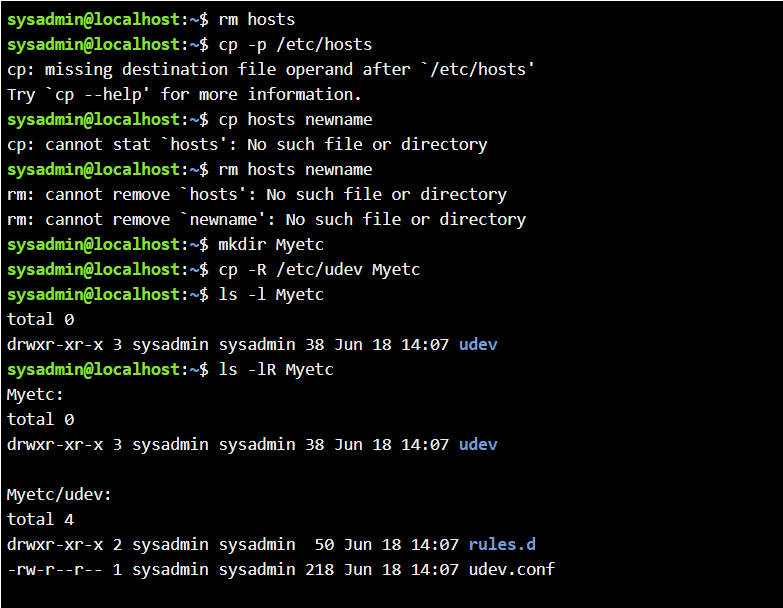
# Capítulo 6:

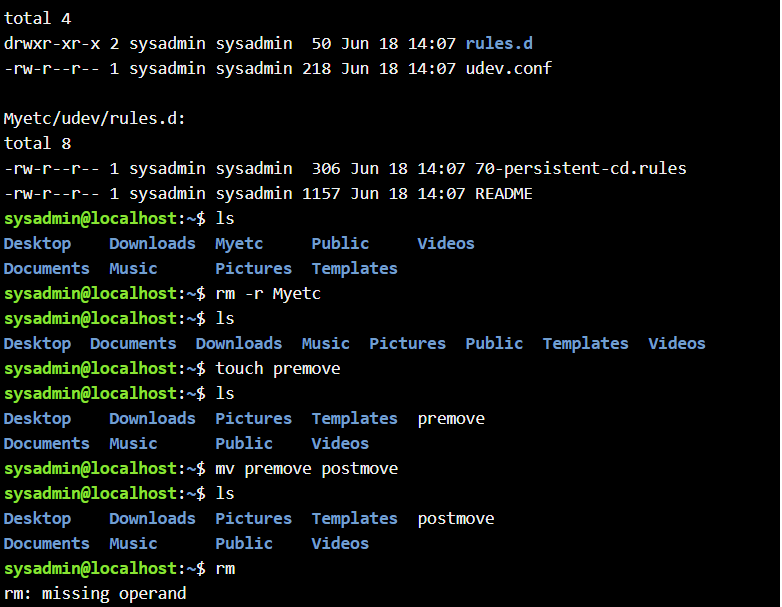






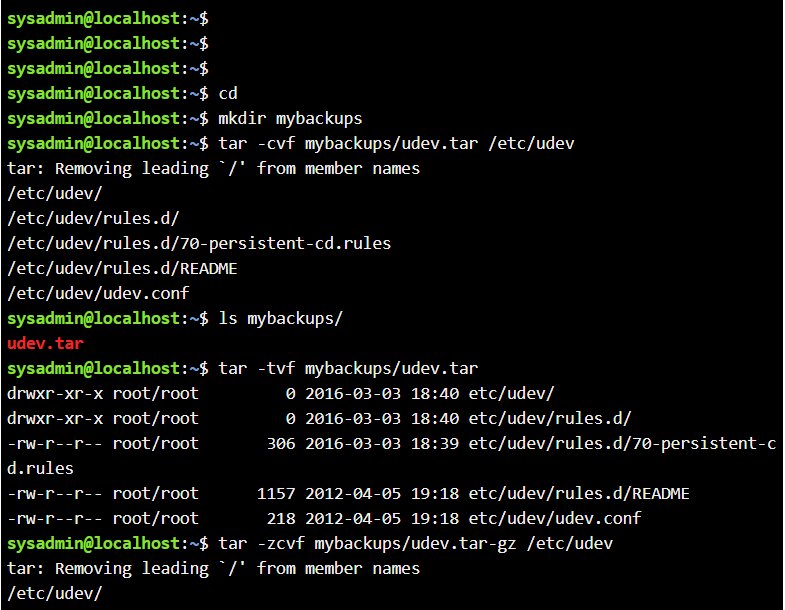


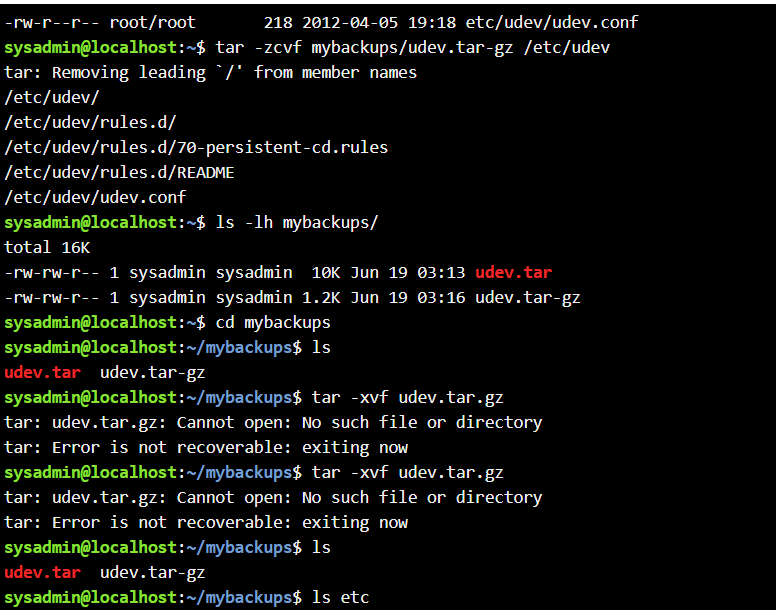


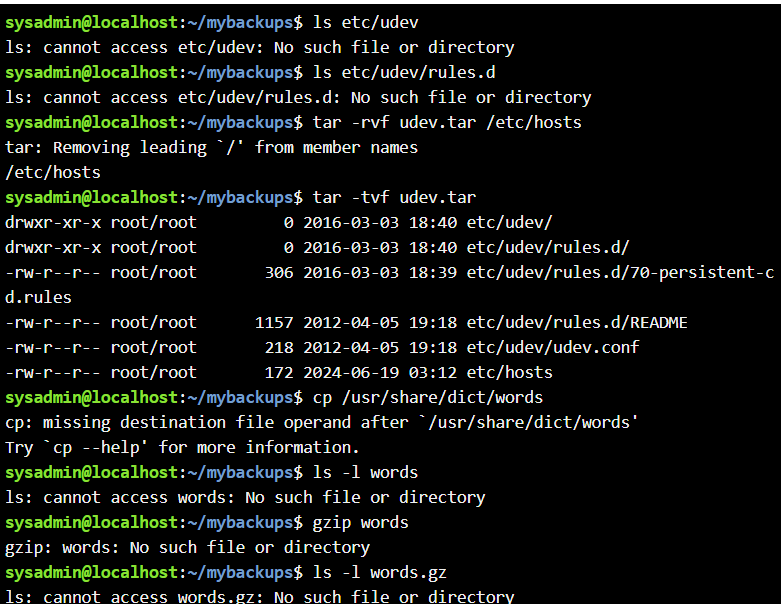


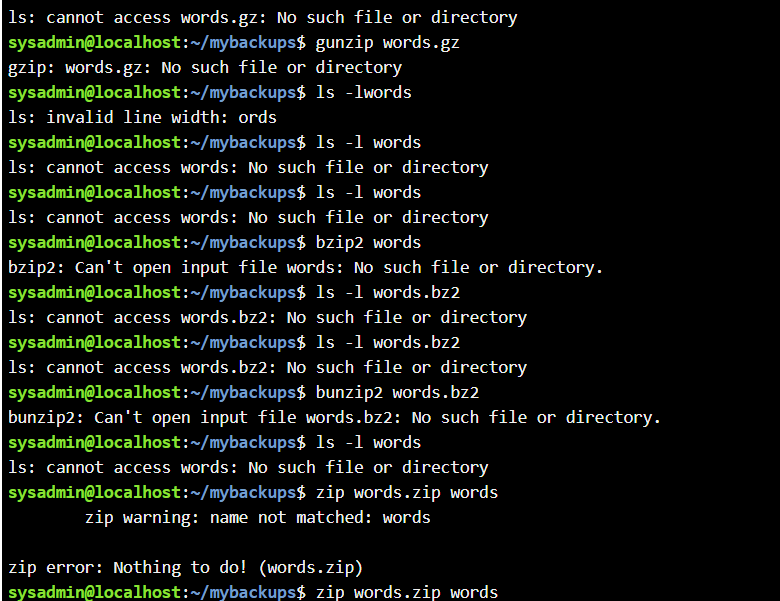
1. `pwd`: Muestra el directorio de trabajo actual.
2. `echo $HOME`: Muestra la ruta al directorio principal del usuario.
3. `cd /`: Cambia el directorio actual al directorio raíz.
4. `pwd`: Muestra el directorio de trabajo actual (ahora raíz).
5. `cd`: Cambia el directorio actual al directorio principal del usuario.
6. `pwd`: Muestra el directorio de trabajo actual (ahora principal).
7. `cd /home`: Cambia el directorio actual al directorio /home.
8. `pwd`: Muestra el directorio de trabajo actual (ahora /home).
9. `cd ~`: Cambia el directorio actual al directorio principal del usuario.
10. `pwd`: Muestra el directorio de trabajo actual (ahora principal).
11. `echo ~ ~sysadmin ~root ~mail ~nobody`: Muestra los directorios principales para los usuarios listados.
12. `cd ~root`: Intenta cambiar al directorio principal de root (requiere privilegios de root).
13. `cd /usr/bin`: Cambia el directorio actual a /usr/bin.
14. `pwd`: Muestra el directorio de trabajo actual (ahora /usr/bin).
15. `cd /usr`: Cambia el directorio actual a /usr.
16. `/usr$ pwd`: Es una sintaxis incorrecta; parece un intento de mostrar el directorio de trabajo pero debería ser solo `pwd`.
17. `cd /usr/share/doc`: Cambia el directorio actual a /usr/share/doc.
18. `pwd`: Muestra el directorio de trabajo actual (ahora /usr/share/doc).
19. `cd bash`: Cambia el directorio actual a bash, que debería ser un subdirectorio de /usr/share/doc.
20. `pwd`: Muestra el directorio de trabajo actual (ahora dentro del subdirectorio bash).
21. `cd ..`: Sube un nivel en la jerarquía de directorios.
22. `pwd`: Muestra el directorio de trabajo actual (de vuelta a /usr/share/doc).
23. `cd ../dict`: Cambia al subdirectorio dict, que es un hermano de doc en /usr/share.
24. `pwd`: Muestra el directorio de trabajo actual (ahora dentro del subdirectorio dict).
25. `cd`: Cambia al directorio principal del usuario.
26. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
27. `ls -a`: Lista todos los archivos y directorios, incluyendo los ocultos (aquellos que comienzan con un punto).
28. `ls -l /etc/hosts`: Lista información detallada sobre el archivo /etc/hosts.
29. `ls -R /etc/udev`: Lista todos los archivos y directorios recursivamente bajo /etc/udev.
30. `ls -d /etc/s\*`: Lista los directorios que comienzan con 's' bajo /etc/.
31. `ls -d /etc/????`: Lista los directorios con exactamente cuatro caracteres bajo /etc/.
32. `ls -d /etc/[abcd]\*`: Lista los directorios bajo /etc/ que comienzan con 'a', 'b', 'c' o 'd'.
33. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
34. `cp /etc/hosts hosts`: Copia el archivo /etc/hosts al directorio de trabajo actual con nombre 'hosts'.
35. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
36. `rm hosts`: Elimina el archivo 'hosts' del directorio de trabajo actual.
37. `ls4` probablemente es un error tipográfico; debería ser solo 'ls'.
38. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
39. `cp -v /etc/hosts hosts` Copia el archivo /etc/hosts a 'hosts' con salida detallada, mostrando lo que se está haciendo.
40. `ls` Lista archivos y direct
41. `rm hosts`: Elimina el archivo 'hosts' del directorio de trabajo actual.
42. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
43. `cp -v /etc/hosts .`: Copia el archivo /etc/hosts al directorio de trabajo actual con salida detallada.
44. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
45. `rm hosts`: Elimina el archivo 'hosts' del directorio de trabajo actual.
46. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
47. `cd /etc`: Cambia el directorio actual a /etc.
48. `ls -l hosts`: Lista información detallada sobre el archivo hosts en /etc.
49. `cp -p hosts /home/sysadmin`: Copia el archivo hosts a /home/sysadmin preservando los permisos.
50. `cd`: Cambia al directorio principal del usuario.
51. `ls -l hosts`: Lista información detallada sobre el archivo hosts en el directorio principal (si existe).
52. `rm hosts`: Elimina el archivo 'hosts' del directorio de trabajo actual (si existe).
53. `cp -p /etc/ hosts ~`: Es una sintaxis incorrecta; debería ser solo `cp -p /etc/hosts ~`.
54. `cp -p /etc/hosts ~`: Copia el archivo /etc/hosts al directorio principal del usuario preservando los permisos.
55. `cp hosts newname`: Copia el archivo 'hosts' a un nuevo archivo llamado 'newname' (si 'hosts' existe).
56. `ls -l hosts newname`: Lista información detallada sobre los archivos 'hosts' y 'newname'.
57. `rm hosts newname`: Elimina los archivos 'hosts' y 'newname'.
58. `mkdir Myetc`: Crea un nuevo directorio llamado Myetc.
59. `cp -R /etc/udev Mytec`: Es un error tipográfico; debería ser 'Myetc'. Copia recursivamente el contenido de /etc/udev a Myetc.
60. `ls -l Mytec`: Lista información detallada sobre el contenido de Mytec (si se creó correctamente).
61. `ls -lR Mytec`: Lista recursivamente información detallada sobre el contenido de Mytec.
62. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
63. `rm -r Mytec`: Elimina recursivamente el directorio Mytec y su contenido.
64. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
65. `touche premove`: Es un error tipográfico; debería ser 'touch premove'.
66. `touch premove`: Crea un nuevo archivo vacío llamado 'premove'.
67. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
68. `mv premove postmove`: Renombra o mueve el archivo 'premove' a 'postmove'.
69. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
70. `rm postmove`: Elimina el archivo 'postmove'.
71. `ls`: Lista archivos y directorios en el directorio de trabajo actual.
72. `history`: Muestra la historia reciente de comandos utilizados en la terminal.

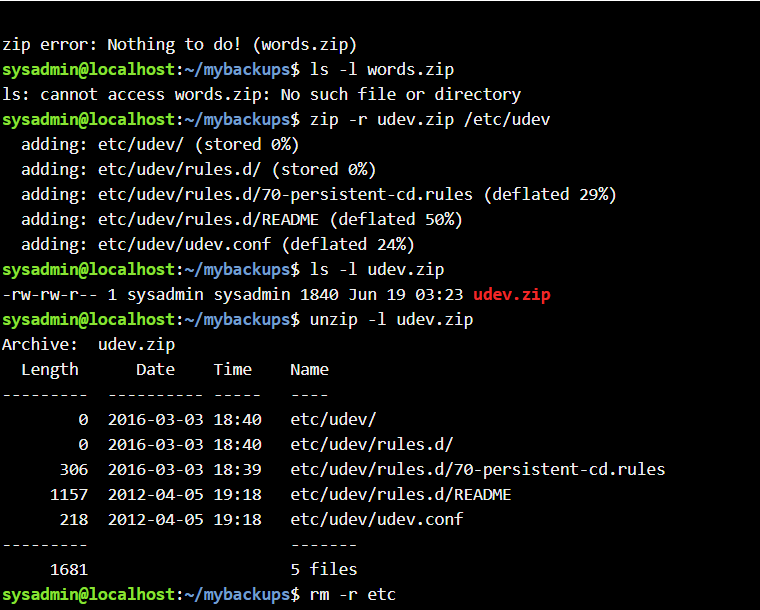
# Capitulo 7:

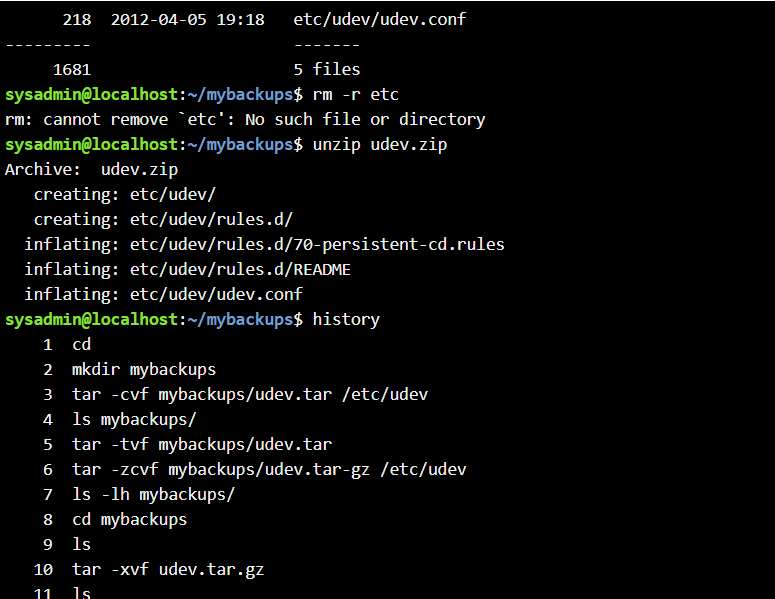






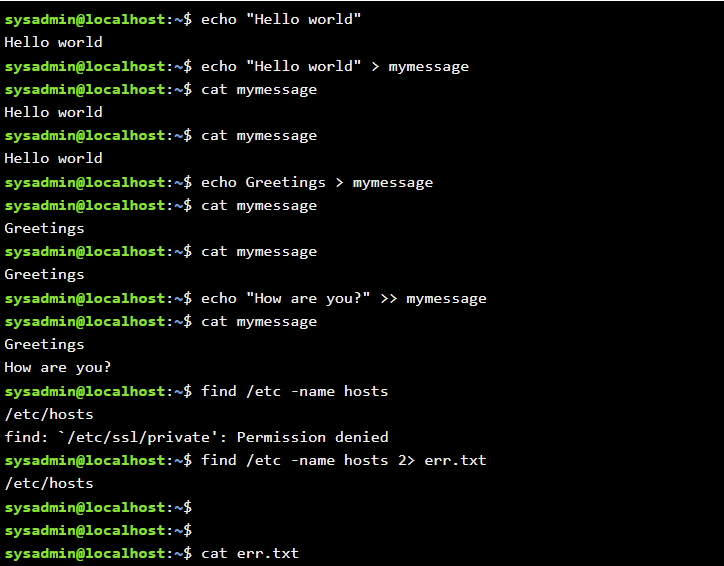


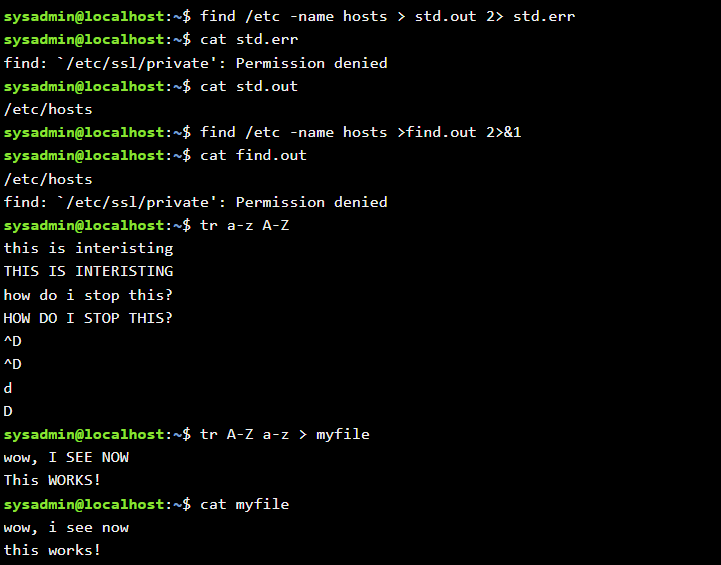


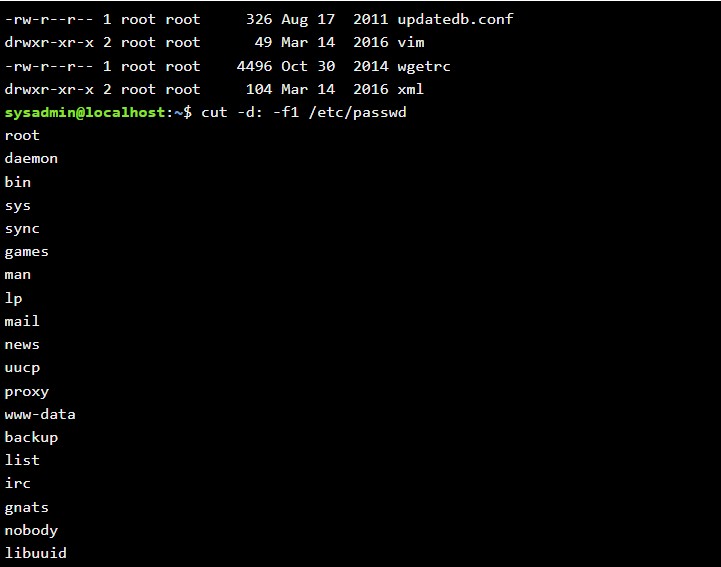


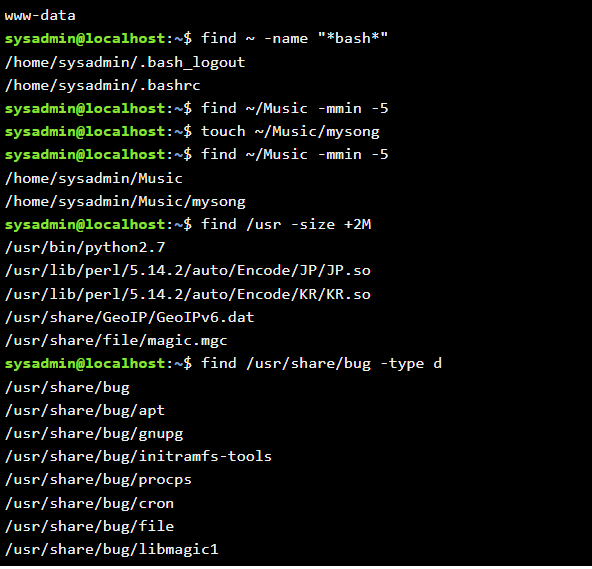
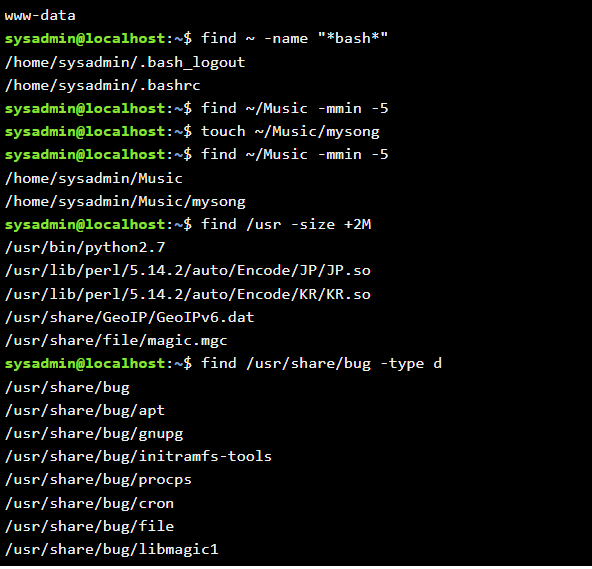
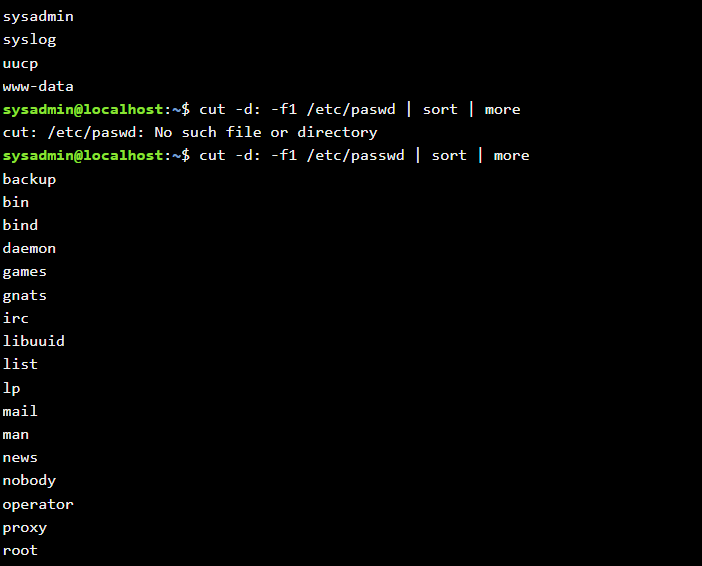
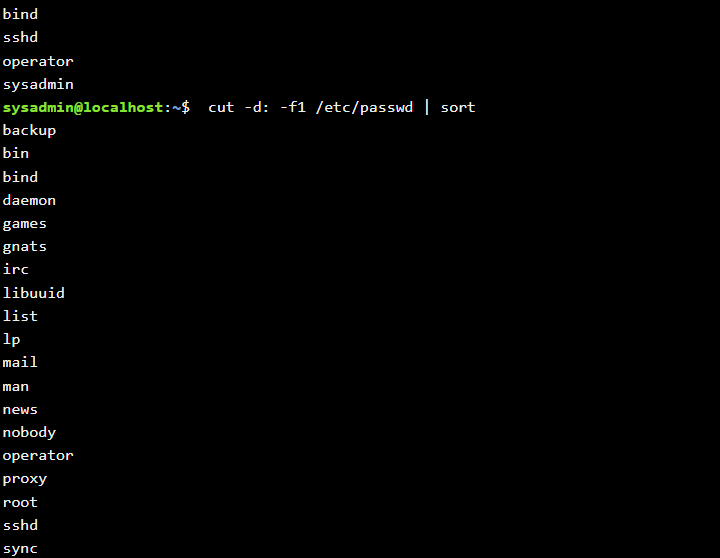
1. `cd`: Cambia el directorio actual al directorio de inicio del usuario.
2. `mkdir mybackups`: Crea un nuevo directorio llamado `mybackups`.
3. `tar -cvf mybackups/udev.tar /etc/udev`: Crea un archivo tar llamado `udev.tar` del directorio `/etc/udev` y lo guarda en el directorio `mybackups`.
4. `ls mybackups/`: Lista los archivos dentro del directorio `mybackups`.
5. `tar -tvf mybackups/udev.tar`: Muestra el contenido del archivo tar `udev.tar` sin extraerlo.
6. `tar -zcvf mybackups/udev.tar-gz /etc/udev`: Crea un archivo tar comprimido con gzip llamado `udev.tar-gz` del directorio `/etc/udev` y lo guarda en el directorio `mybackups`.
7. `ls -lh mybackups/`: Lista los archivos dentro de `mybackups` mostrando su tamaño en un formato legible por humanos.
8. `cd mybackups`: Cambia el directorio actual al directorio `mybackups`.
9. `ls`: Lista los archivos en el directorio actual.
10. `tar -xvf udev.tar.gz`: Intenta extraer el contenido del archivo tar comprimido con gzip llamado `udev.tar.gz`, pero hay un error en el comando ya que debería ser `.tar.gz` al final del nombre del archivo.
11. `ls`: Lista los archivos en el directorio actual.
12. `ls etc`: Lista los archivos dentro del directorio `etc`, pero este comando fallará si no existe un subdirectorio llamado `etc` en el directorio actual.
13. `ls etc/udev`: Lista los archivos dentro del subdirectorio `/etc/udev`, pero este comando fallará si no estás en la raíz o si no existe tal ruta relativa desde el directorio actual.
14. `ls etc/udev/rules.d`: Lista los archivos dentro del subdirectorio `/etc/udev/rules.d`, pero este comando también fallará si no estás en la raíz o si no existe tal ruta relativa desde el directorio actual.
15. `tar -rvf udev.tar /etc/hosts`: Añade el archivo `/etc/hosts` al archivo tar existente llamado `udev.tar`.
16. `tar -tvf udev.tar`: Muestra el contenido actualizado del archivo tar `udev.tar` sin extraerlo.
17. `cp /usr/share/dict/words`: Este comando está incompleto, falta especificar dónde se quiere copiar el archivo `words`.
18. `ls -l words`: Lista los detalles del archivo llamado `words`, pero este comando fallará si no hay un archivo con ese nombre en el directorio actual.
19. `gzip words`: Comprime el archivo `words` usando la compresión gzip y lo renombra a `words.gz`.
20. `ls -l words.gz`: Lista los detalles del archivo comprimido `words.gz`.
21. `gunzip words.gz`: Descomprime el archivo `words.gz` y lo restaura a su nombre original `words`.
22. `ls -lwords`: Parece haber un error tipográfico, debería ser `ls -l words` para listar los detalles del archivo `words`.
23. `ls -l words`: Lista los detalles del archivo `words`.
24. `bzip2 words`: Comprime el archivo `words` usando la compresión bzip2 y lo renombra a `words.bz2`.
25. `ls -l words.bz2`: Lista los detalles del archivo comprimido `words.bz2`.
26. `bunzip2 words.bz2`: Descomprime el archivo `words.bz2` y lo restaura a su nombre original `words`.
27. `ls -l words`: Lista los detalles del archivo `words`.
28. `zip words.zip words`: Crea un archivo zip llamado `words.zip` que contiene el archivo `words`.
29. `ls -l words.zip`: Lista los detalles del archivo zip `words.zip`.
30. `zip -r udev.zip /etc/udev`: Crea un archivo zip llamado `udev.zip` que contiene de manera recursiva todo el contenido del directorio `/etc/udev`.
31. `ls -l udev.zip`: Lista los detalles del archivo zip `udev.zip`.
32. `unzip -l udev.zip`: Muestra el contenido del archivo zip `udev.zip` sin extraerlo.
33. `rm -r etc`: Elimina de manera recursiva el directorio llamado 'etc' y todo su contenido.
34. `unzip udev.zip`: Extrae el contenido del archivo zip 'udev.zip' en el directorio actual.

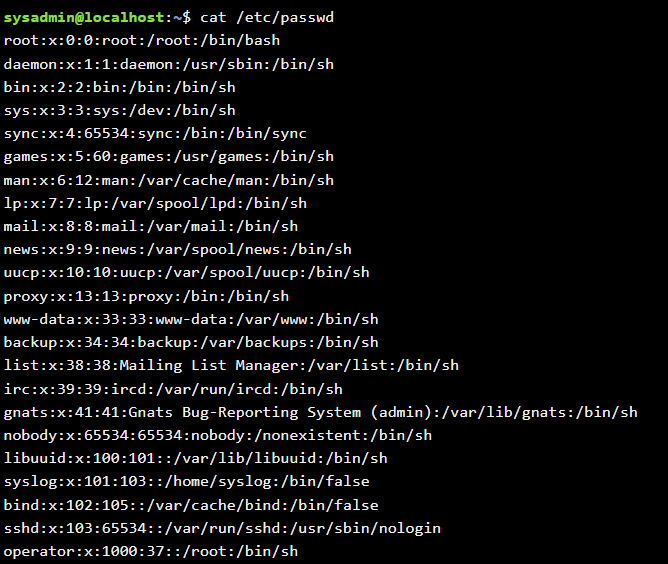
# Capitulo 8:

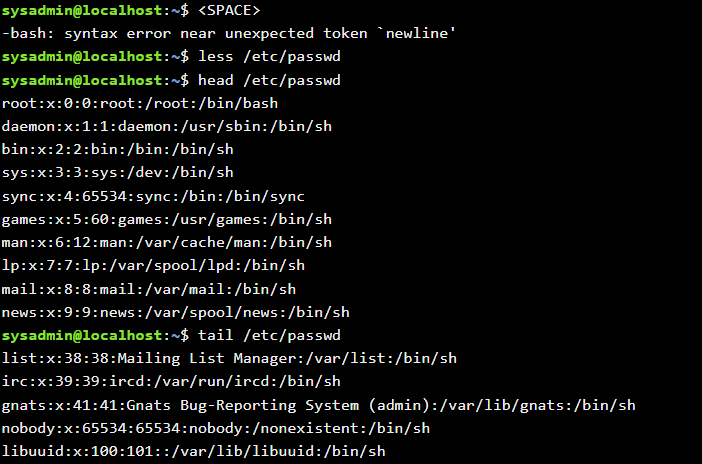
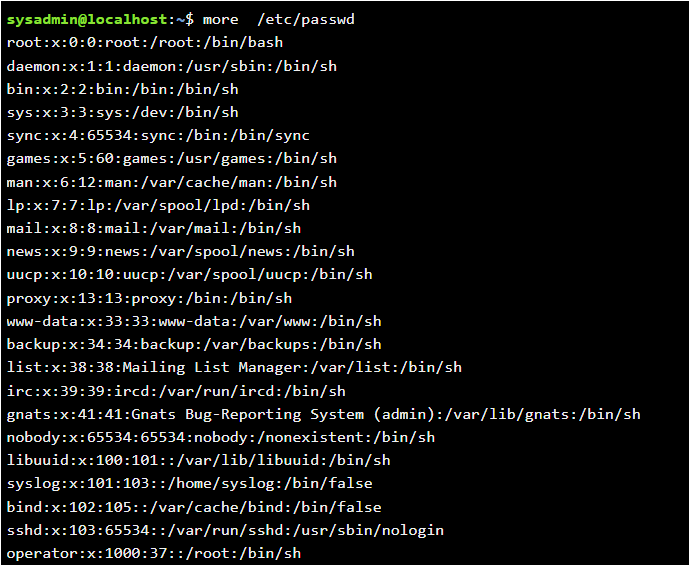


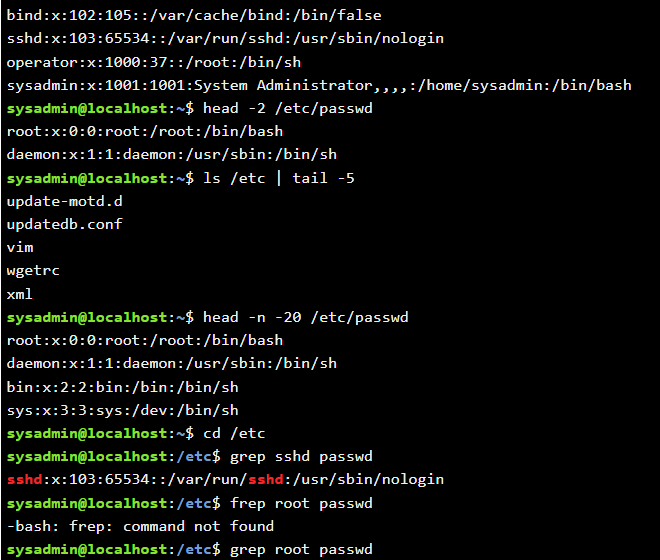


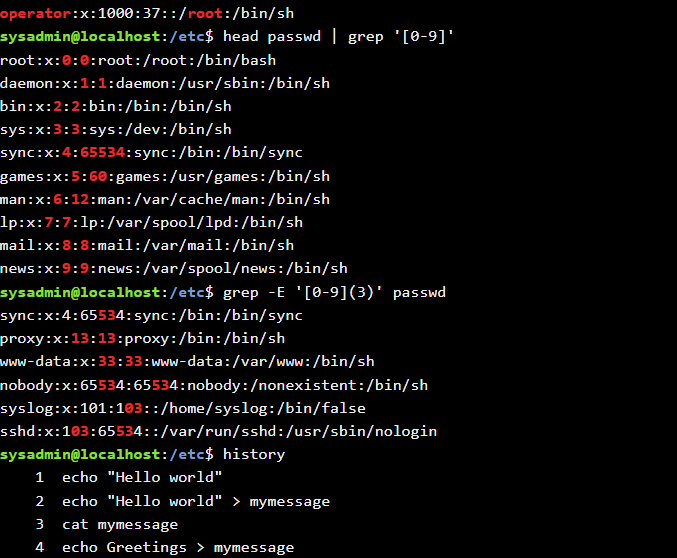
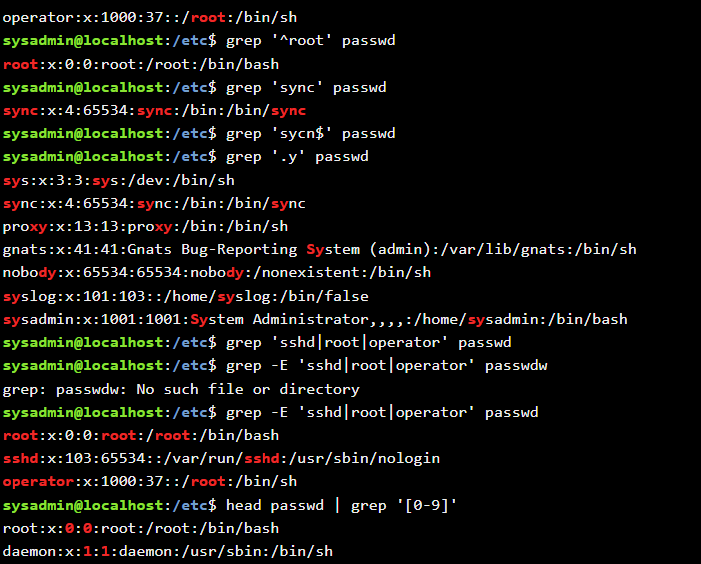












**echo**: Este comando muestra en la terminal el texto o variables que le proporcionas como argumentos. Por ejemplo, echo "¡Hola, mundo!" imprimirá la frase “¡Hola, mundo!” en la terminal

**cat**: El comando cat se utiliza para mostrar el contenido de un archivo. En tu ejemplo, cat mymessage mostrará el contenido del archivo llamado “mymessage”.

**find**: Este comando busca archivos o directorios en una ubicación específica. Por ejemplo, find /etc -name hosts buscará archivos llamados “hosts” dentro del directorio “/etc”. Si agregas 2> err.txt, redirigirá los errores a un archivo llamado “err.txt”.

**tr**: El comando tr se utiliza para transformar o reemplazar caracteres en un flujo de texto. Por ejemplo, tr a-z A-Z cambiará todas las letras minúsculas a mayúsculas. Si haces tr A-Z a-z > myfile, se creará un archivo llamado “myfile” con el texto transformado.

\*\*`tr a-z A-Z < myfile`\*\*: Este comando transforma el contenido del archivo "myfile" convirtiendo todas las letras minúsculas a mayúsculas.

\*\*`ls -l /etc | more`\*\*: El comando `ls` lista los archivos y directorios en el directorio "/etc". La opción `-l` muestra los detalles en formato largo, y el operador `|` (pipe) envía la salida al comando `more`, que permite ver la lista de forma paginada.

\*\*`cut -d: -f1 /etc/passwd`\*\*: El comando `cut` extrae campos de un archivo de texto. En este caso, se separan los campos del archivo "/etc/passwd" utilizando ":" como delimitador, y se muestra la primera columna (nombres de usuario).

\*\*`find ~ -name "\*bash\*"`\*\*: Busca archivos o directorios en tu directorio de inicio (~) que contengan la palabra "bash" en su nombre.

\*\*`find ~/Music -mmin -5`\*\*: Encuentra archivos en el directorio "~/Music" modificados en los últimos 5 minutos.

\*\*`touch ~/Music/mysong`\*\*: Crea un archivo llamado "mysong" en el directorio "~/Music".

\*\*`find ~/Music -mmin -5`\*\*: Busca nuevamente archivos en el directorio "~/Music" modificados en los últimos 5 minutos (incluyendo el archivo recién creado).

\*\*`find /usr -size +2M`\*\*: Encuentra archivos en el directorio "/usr" que tengan un tamaño mayor a 2 megabytes.

\*\*`find /usr/share/bug -type d`\*\*: Busca directorios en "/usr/share/bug".

\*\*`find /usr/share/bug -type d -ls`\*\*: Muestra detalles de los directorios encontrados en "/usr/share/bug".

\*\*`cat /etc/passwd`\*\*: Muestra el contenido del archivo "/etc/passwd".

\*\*`more /etc/passwd`\*\*: Permite ver el contenido del archivo "/etc/passwd" de forma paginada.

\*\*`h`\*\*: No es un comando válido. Puede ser un error tipográfico o una referencia a algo específico en tu sistema.

\*\*`more h`\*\*: Intenta mostrar el contenido del archivo "h" de forma paginada.

\*\*`<SPACE>`\*\*: Esto no es un comando válido. Puede ser un espacio en blanco o un error tipográfico.

\*\*`less /etc/passwd`\*\*: Similar a `more`, muestra el contenido del archivo "/etc/passwd" de forma paginada, pero con más opciones de navegación.

\*\*`head /etc/passwd`\*\*: Muestra las primeras líneas del archivo "/etc/passwd".

\*\*`tail /etc/passwd`\*\*: Muestra las últimas líneas del archivo "/etc/passwd".

\*\*`head -2 /etc/passwd`\*\*: Este comando muestra las primeras dos líneas del archivo "/etc/passwd". En este caso, probablemente verás información sobre usuarios del sistema.

\*\*`ls /etc | tail -5`\*\*: El comando `ls` lista los archivos y directorios en el directorio "/etc". Luego, el operador `|` (pipe) envía la salida al comando `tail -5`, que muestra las últimas 5 líneas de esa lista. Esto te dará una vista rápida de los últimos archivos y directorios en "/etc".

\*\*`head -n -20 /etc/passwd`\*\*: Muestra todas las líneas del archivo "/etc/passwd" excepto las últimas 20. Es decir, muestra todas las líneas excepto las 20 últimas.

\*\*`cd /etc`\*\*: Cambia el directorio actual al directorio "/etc". Ahora estarás trabajando dentro de ese directorio.

\*\*`grep sshd passwd`\*\*: Busca la palabra "sshd" en el archivo "/etc/passwd". Esto podría ser útil para encontrar información relacionada con el servicio SSH.

\*\*`frep root passwd`\*\*: Parece que hay un error tipográfico aquí. Debería ser `grep` en lugar de `frep`. Si corriges esto, el comando buscará la palabra "root" en el archivo "/etc/passwd".

\*\*`grep root passwd`\*\*: Similar al comando anterior, busca la palabra "root" en el archivo "/etc/passwd".

\*\*`grep '^root' passwd`\*\*: Busca las líneas que comienzan con la palabra "root" en el archivo "/etc/passwd". El símbolo `^` indica el inicio de la línea.

\*\*`grep 'sync' passwd`\*\*: Busca la palabra "sync" en el archivo "/etc/passwd". Esto podría estar relacionado con algún usuario o proceso del sistema.

\*\*`grep 'sycn$' passwd`\*\*: Aquí también parece haber un error tipográfico. Debería ser `grep 'sync$' passwd`. Este comando busca las líneas que terminan con la palabra "sync" en el archivo "/etc/passwd".

\*\*`grep '.y' passwd`\*\*: Busca las líneas que contienen cualquier caracter seguido de la letra "y" en el archivo "/etc/passwd".

\*\*`grep 'sshd|root|operator' passwd`\*\*: Busca las líneas que contienen cualquiera de las palabras "sshd", "root" o "operator" en el archivo "/etc/passwd".

\*\*`grep -E '[0-9](3)' passwd`\*\*: Utiliza expresiones regulares extendidas (-E) para buscar patrones. En este caso, busca líneas que contengan un dígito seguido de la cifra "3" en el archivo "/etc/passwd".

# Capítulo 10:

