

Sistemas Operativos I

Comandos útiles del Shell

Edwin Salvador

10 de diciembre de 2015

Sesión 11

Ejercicios

- En su directorio `home` crear el directorio `lab10`
- Dentro de `lab10` crear dos directorios llamados: `uno` y `dos` (hacerlo con un solo comando)
- Escribir **UN** comando que cree un archivo **oculto** llamado `secreto` dentro de `lab10` con el contenido `"Este es un secreto"`.
- Estando dentro de `lab10`, escribir **un** comando que cree un archivo llamado `ficheros_bin_descendente` que contenga el listado de todos los ficheros que se encuentran en el directorio `/bin` ordenados de manera descendente (de `z - a`).
- Ingresar al directorio `uno`
- Estando en `uno`, escribir **un** comando que utilice **tuberias y filtros** para crear, dentro del directorio `dos`, un archivo llamado `ficheros_bin_ascendente` que contenga el texto del archivo `ficheros_bin_descendente` pero ordenado de manera ascendente.

Ejercicios

- Estando en el directorio `uno`, escribir un comando que mueva el archivo `ficheros_bin_descendente` al directorio `dos`
- Estando en el directorio `uno`, escribir un comando que nos permita mover hacia el directorio `dos`
- Estando en `dos`, escribir un comando que cree un archivo oculto llamado `tmp`
- Estando en `dos`, escribir un comando que elimine el directorio `uno`
- Estando en `dos`, escribir un comando que mueva todos los archivos en el directorio actual hacia la carpeta `lab10`
- Estando en `dos`, eliminar el directorio `dos` (si, el directorio actual)
- Sin moverse del directorio actual, crear un archivo llamado `tmp`. ¿Qué sucede?
- Moverse al directorio `lab10`.

Ejercicios

- Escribir un comando que cree un archivo llamado `ficheros` que contenga el listado de todos los archivos (incluidos los archivos ocultos) que se encuentren en el directorio actual.
- Escribir un comando que cree un archivo llamado `historial_mk` que contenga el historial de comandos ejecutados que contienen la palabra `mkdir`.

- Utilizado para seleccionar los permisos de acceso a un archivo.
- Tres clases de permisos: Leer (r), Escribir (w), Ejecutar (x).
- Tres tipos de usuarios que pueden manejar el archivo: Dueño (u), Grupo (g), Otros (o)
- Su sintaxis es: `chmod tipos_usuarios[+|-]permisos fichero`
- Ejemplo: `chmod o+x ejemplo.sh`
- Ejemplo: `chmod go-w mydata.dat`

Formato octal

- $r=4$ $w=2$ $x=1$
- $rwX = 7$ ($4+2+1$)
- $rw- = 6$ ($4+2$)
- $r-x = 5$ ($4+1$)
- Ejemplo: $rw-r-r- = 644$
- Ejemplo: $rw---- = 600$
- Ejemplo: $rw-r-xr-x = 755$
- Para que el solo el dueño pueda leer y escribir en el fichero: `chmod 600 <fichero>`
- Para que sea ejecutable por todos: `chmod 755 <fichero>`

Comando grep

- Recibe una expresión regular y el nombre de uno o más ficheros
- Presenta por pantalla las líneas de los ficheros que coinciden con el patron de la expresión regular
- Ejemplo: `grep ^W`

Metacarácter	Significado
.	Representa a cualquier carácter
[lista de caracteres] o [carácter ₁ – carácter _n]	Representa a uno cualquiera de los caracteres de la lista, o a cualquier carácter comprendido entre carácter ₁ y carácter _n según el orden ASCII. Si el primer carácter tras el corchete [es el carácter '^', el significado se invierte, es decir, representa a todos los caracteres que no están en la lista o en el intervalo. Dentro de los corchetes, los metacaracteres '\$', '*', y '/' pierden su significado especial.
*	Pospuesta a cualquier expresión y significa cero o más ocurrencias de dicha expresión.
^	Antepuesta a cualquier expresión regular, indica que la expresión debe aparecer al comienzo de la línea solamente.
\$	Pospuesta a cualquier expresión regular, indica que la expresión debe aparecer al final de la línea solamente.
\	El significado de cualquier metacarácter puede ser ignorado antecediéndole por la barra inversa ('\'), en cuyo caso el metacarácter se interpreta de forma literal.

Figura : RegEX

Ejercicios

- Crear un directorio `lab11` en el home
- Quitarle todos los permisos para todos los usuarios
- Ejecutar `ls -l lab11`. ¿Qué sucede? ¿Por qué sucede esto?
- ¿Cómo puedo hacer para poder ver los archivos dentro de `lab11`?
- Cambiar los permisos del directorio para que el dueño pueda únicamente leer los archivos
- Crear un archivo llamado `ejemplo.sh`
- ¿Se pudo crear el fichero? ¿Si no se pudo crear el fichero ejecutar el comando para permitir que el dueño pueda escribir en el directorio `lab11`.
- Crear el fichero `ejemplo.sh`.
- Ejecutar el comando `history` y guardar su resultado en un archivo llamado `historial.txt`
- Filtrar las líneas del fichero `historial.txt` para que incluya solo las líneas que empiezan con `c` y guardarlo en un fichero `historial_c.txt`