

Fichero Aclaratorio Práctica 4: Expresiones regulares en Bash

Pablo Rodríguez Guillén
Profesor: Enrique García Salcines

26 de mayo de 2019

Índice

1. ejercicio1.sh	2
2. ejercicio2.sh	3
3. ejercicio3.sh	4
4. ejercicio4.sh	5

1. ejercicio1.sh

El ejercicio hace uso principalmente del comando *grep* con algunas de sus opciones, principalmente la opción *-colour* que hace que el patrón que coincida con la expresión regular se resalte con color en la salida de pantalla. El comando *sed* también es utilizado en el último apartado del ejercicio. A la salida de este se le aplica un *grep* para resaltar las modificaciones.

Cada uno de los apartados está codificado en el fichero *ejercicio1.sh* y la explicación concreta de cada uno está disponible a través de comentarios de bash. Al inicio del script se encuentra la comprobación de errores al pasar el argumento, que debe ser un fichero. A continuación se muestra la primera parte de la ejecución del script, no se mostrará entera ya que la salida es extensa.

```
➔ Practice_4 git:(master) ✕ ejercicio1.sh peliculas.txt
*****
1) Líneas con la duración de la película:
1hr 59min
2hr 17min
1hr 33min
2hr 09min
1hr 48min
1hr 25min
*****
2) Líneas con el país de las películas:
(10/03/2017) -Estados Unidos, Vietnam-
(03/03/2017) -Estados Unidos-
(24/02/2017) -España-
(03/03/2017) -España-
(22/12/2016) -Estados Unidos-
(29/01/2015) -Rusia-
*****
3) Solo el país de las películas:
Estados Unidos, Vietnam
Estados Unidos
España
España
Estados Unidos
Rusia
*****
4) Hay 1 películas del 2016 y 4 películas del 2017
*****
5) Eliminar las líneas vacías:
Kong: La Isla Calavera
=====
(10/03/2017) -Estados Unidos, Vietnam-
Dirigida por Jordan Vogt-Roberts
Reparto: Tom Hiddleston, Samuel L. Jackson, Brie Larson...
    Un grupo de viajeros, exploradores y soldados de lo más dispar se reúne para viajar a una misteriosa isla del
pas. Entre ellos están el Capitán James Conrad (Tom Hiddleston), el Teniente Coronel Packard (Samuel L. Jackson)
) amante de la naturaleza. Pero al adentrarse en esta bella y también traicionera isla, los exploradores encontr
te. Sin saberlo, estarán invadiendo los dominios del mítico Kong, el gigante gorila rey de esta isla. Será Marlo
bitante del lugar, quien les enseñe los secretos de Isla Calavera, además del resto de seres monstruosos que la
1hr 59min
```

Figura 1: Salida de los 4 primeros apartados del ejercicio 1

2. ejercicio2.sh

Este ejercicio comprueba que se le pasa un solo argumento por línea de comandos y que ese argumento, es la ruta de un fichero regular que existe en la máquina en la que se está ejecutando el script.

En primer lugar se hace un *grep -v -E* que hace que se impriman las líneas que no coinciden con la expresión regular especificada. A esta salida se le aplica un *sed* con varios comandos. El primero de ellos es un *d*, que elimina las líneas que emparejan con la expresión regular precedente. El resto son comandos *s* que tiene dos partes separadas por el carácter */*. La primera de ellas es una expresión regular en la que podemos crear una referencia para repetir el contenido de esta en la segunda parte. En esta segunda parte, especificamos la cadena que va a sustituir al texto que empareje con la expresión regular de la primera parte, recalcar que sustituye a la expresión entera no a la referencia. Esta referencia también puede llamarse argumento *t* se utiliza para repetir una parte de la cadena en la sustitución.

```
→ Practice_4 git:(master) x ejercicio2.sh peliculas.txt
Titulo: Kong: La Isla Calavera
|-> Fecha de estreno: 10/03/2017
|-> Director: Jordan Vogt-Roberts
|-> Reparto: Tom Hiddleston, Samuel L. Jackson, Brie Larson...
|-> Duración: 1hr 59min
Titulo: Logan
|-> Fecha de estreno: 03/03/2017
|-> Director: James Mangold
|-> Reparto: Hugh Jackman, Patrick Stewart, Dafne Keen...
|-> Duración: 2hr 17min
Titulo: Es por tu bien
|-> Fecha de estreno: 24/02/2017
|-> Director: Carlos Therón
|-> Reparto: José Coronado, Javier Cámara, Roberto Álamo...
|-> Duración: 1hr 33min
Titulo: El guardián Invisible
|-> Fecha de estreno: 03/03/2017
|-> Director: Fernando González Molina
|-> Reparto: Marta Etura, Elvira Minguez, Nene...
|-> Duración: 2hr 09min
Titulo: ¡Canta!
|-> Fecha de estreno: 22/12/2016
|-> Director: Garth Jennings
|-> Reparto: Matthew McConaughey, Reese Witherspoon, Seth MacFarlane...
|-> Duración: 1hr 48min
Titulo: Red Army
|-> Fecha de estreno: 29/01/2015
|-> Director: Gabe Polsky
|-> Reparto: Documentary, Gabe Polsky...
|-> Duración: 1hr 25min
→ Practice_4 git:(master) x
```

Todos los emparejamientos de las expresiones regulares están explicados en el script a través de comentarios.

3. ejercicio3.sh

Este script permite un argumento opcional, si recibe más de un argumento se imprime por pantalla un mensaje de error y se abandona la ejecución del script. El primer apartado consiste en aplicar un *grep* que empareje con todos los archivos ocultos de la salida de un *ls -a* en el directorio *home* del usuario que ejecute el script.

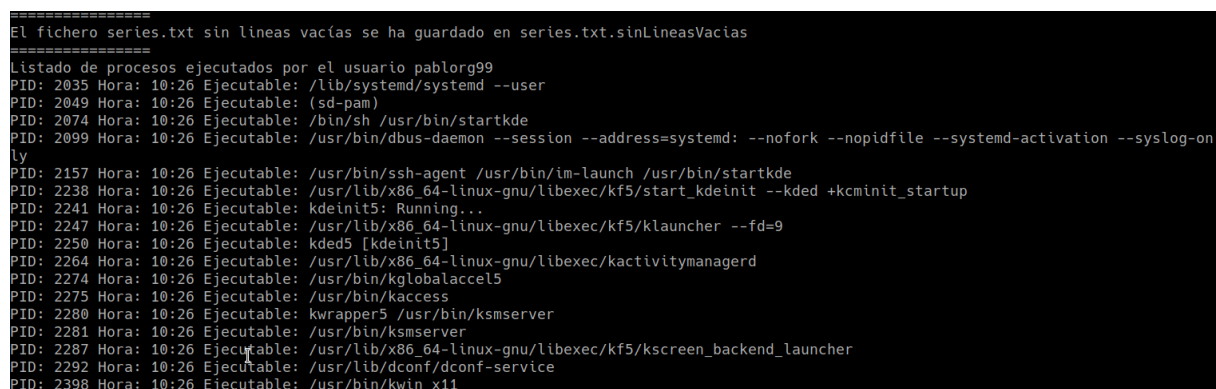
La ejecución del segundo apartado varía según si se incluye el argumento opcional. Si se especifica un fichero de texto por línea de comandos. Se hace una copia del mismo eliminando las líneas vacías, haciendo uso de *grep* y el operador *>*

El tercer apartado utiliza el comando *sed* precedido de *grep* (por razones especificadas en el script) para mostrar algunos campos de los procesos ejecutados por el usuario actual. La expresión regular de este *sed* está detallada en el script a través de comentarios.



```
→ Practice_4 git:(master) x ejercicio3.sh series.txt
=====
Listado de los archivos ocultos del directorio /home/pablor99
.
..
.apt-vim
.bash_history
.bash_logout
.bashrc
.cache
.CLion2019.1
.config
.dic
.directory
.dropbox
.dropbox-dist
.face
.face.icon
```

(a) Comienzo del apartado 1



```
=====
El fichero series.txt sin líneas vacías se ha guardado en series.txt.sinLineasVacías
=====
Listado de procesos ejecutados por el usuario pablor99
PID: 2035 Hora: 10:26 Ejecutable: /lib/systemd/systemd --user
PID: 2049 Hora: 10:26 Ejecutable: (sd-pam)
PID: 2074 Hora: 10:26 Ejecutable: /bin/sh /usr/bin/startkde
PID: 2099 Hora: 10:26 Ejecutable: /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation --syslog-on
ly
PID: 2157 Hora: 10:26 Ejecutable: /usr/bin/ssh-agent /usr/bin/im-launch /usr/bin/startkde
PID: 2238 Hora: 10:26 Ejecutable: /usr/lib/x86_64-linux-gnu/libexec/kf5/start_kdeinit --kded +kcmnit_startup
PID: 2241 Hora: 10:26 Ejecutable: kdeinit5: Running...
PID: 2247 Hora: 10:26 Ejecutable: /usr/lib/x86_64-linux-gnu/libexec/kf5/klauncher --fd=9
PID: 2250 Hora: 10:26 Ejecutable: kded5 [kdeinit5]
PID: 2264 Hora: 10:26 Ejecutable: /usr/lib/x86_64-linux-gnu/libexec/kactivitymanagerd
PID: 2274 Hora: 10:26 Ejecutable: /usr/bin/kglobalaccel5
PID: 2275 Hora: 10:26 Ejecutable: /usr/bin/kaccess
PID: 2280 Hora: 10:26 Ejecutable: kwrapper5 /usr/bin/ksmsserver
PID: 2281 Hora: 10:26 Ejecutable: /usr/bin/ksmsserver
PID: 2287 Hora: 10:26 Ejecutable: /usr/lib/x86_64-linux-gnu/libexec/kf5/kscreen_backend_launcher
PID: 2292 Hora: 10:26 Ejecutable: /usr/lib/dconf/dconf-service
PID: 2398 Hora: 10:26 Ejecutable: /usr/bin/kwin_x11
```

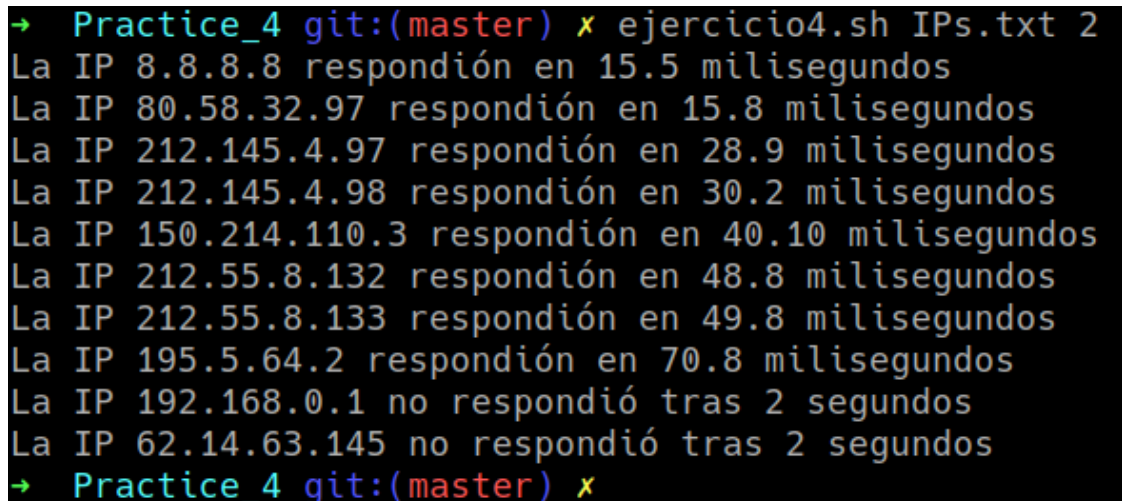
(b) Apartado 2 y parte del apartado 3

Figura 2: Ejecución del script ejercicio3.sh

4. ejercicio4.sh

Este script requiere dos argumentos, si alguno de estos no se especifican o se introducen de manera incorrecta (el fichero no existe o el número de segundos es un número menor o igual a 0), se termina la ejecución del script. Respecto a estos argumentos, decir que el fichero pasado como argumento será uno que contenga una dirección IP válida en cada línea y que el número de segundos será el que el script usará para determinar que un servidor no ha respondido.

El script hace un bucle en el que en cada iteración, la variable *\$ip* es una línea del fichero pasado como argumento y se utiliza como destino en el comando *ping*. Este se ejecuta con dos opciones: *-c 1* que hace que el ping solo envíe un paquete y *-W*, que como se indica en el propio script, devuelve un error cuando el número de segundos especificado como argumento de esta opción transcurre sin que haya habido respuesta por el servidor. Por último destacar que la sentencia condicional *if* no usa *[expression]* porque no evalúa una expresión, sino una variable que se trata como un booleano.



```
→ Practice_4 git:(master) x ejercicio4.sh IPs.txt 2
La IP 8.8.8.8 respondió en 15.5 milisegundos
La IP 80.58.32.97 respondió en 15.8 milisegundos
La IP 212.145.4.97 respondió en 28.9 milisegundos
La IP 212.145.4.98 respondió en 30.2 milisegundos
La IP 150.214.110.3 respondió en 40.10 milisegundos
La IP 212.55.8.132 respondió en 48.8 milisegundos
La IP 212.55.8.133 respondió en 49.8 milisegundos
La IP 195.5.64.2 respondió en 70.8 milisegundos
La IP 192.168.0.1 no respondió tras 2 segundos
La IP 62.14.63.145 no respondió tras 2 segundos
→ Practice_4 git:(master) x
```

Figura 3: Salida de la ejecución del ejercicio 4

El resto de explicaciones concretas del script están documentadas en el código del mismo a través de comentarios.