

SISTEMAS OPERATIVOS

Práctica opcional: bash scripts

Grado de Ingeniería en Informática

Curso 2017/2018



Autores:

Juan José Garzón Luján (100363861), 100363861@alumnos.uc3m.es

Carlos García Cañibano (100363813), 100363813@alumnos.uc3m.es

Andrés Javier Estévez Fernández (100358857), 100358857@alumnos.uc3m.es

Introducción

La práctica ha sido realizada en un entorno Unix personalizado. Hemos probado los scripts para los casos de uso disponibles en la máquina donde ha tenido lugar el desarrollo. Sin embargo, han sido evaluados y su funcionamiento ha sido el correcto para los casos de uso descritos en la memoria de la práctica. Dicho esto, pasamos a la resolución de las cuestiones propuestas en el enunciado.

Ejercicio 2

a) El comando `tr -c [:alnum:] [\n]*`:

En esta parte del comando lo que hacemos es transcribir/truncar los espacios de entre todas las letras y números que aparecen en el documento. De esta forma recibimos una lista que tiene línea por línea toda las palabras que conforman el texto.

La siguiente instrucción `sort`:

Tiene como objetivo ordenar alfabéticamente las palabras que resultan de aplicar el comando anterior. Es decir, ordena nuestra lista de palabras.

Después `uniq -c`:

En este caso lo que hacemos es agrupar las palabras repetidas en una sola ocurrencia dentro de nuestra lista. El comando también nos indica el número de veces que han aparecido.

Para `sort -nr`:

Cambia el orden de clasificación de las palabras de alfabético a lexicográfico según el juego de caracteres que maneja el sistema.

Por último `head -$2`:

Según el segundo parámetro, nos muestra por orden lexicográfico el número de coincidencias que ha tenido una palabra 'x' dentro de nuestro texto. Cuanto mayor sea el número introducido, más palabras nos muestra en cada línea. `Head` selecciona el número de líneas empezando por el principio del documento.

Ejercicio 5:

a) 409 bytes antes de la ejecución del comando del enunciado. 77,1 MB después de la ejecución del comando del enunciado. El tamaño del archivo ha aumentado considerablemente.

b) El script muestra una lista de una serie de directorios que tiene comprimidos y codificados en base64 incluidos en el código del fichero. De esta forma, cuando seleccionamos elementos de la lista, se descomprime y decodifica los directorios seleccionados mientras muestra un mensaje descriptivo de la operación realizada actualmente.

Exit 0 es necesario para que el intérprete de comandos no entienda que todos los datos codificados y comprimidos que se encuentran a continuación del fuente son código con el que debe trabajar.

La expresión `2>&1` combina la salida de error del programa con la salida estándar del mismo. De esta forma, el programa es capaz de mostrar por salida estándar mensajes relacionados con la ejecución normal del programa y con errores que puedan ocurrir.

Toda la secuencia cuyo resultado va a parar a la variable WHAT codifica una caja de texto de tipo checklist donde el usuario es capaz de seleccionar cual/es de los directorios listados en la orden quiere decodificar y descomprimir después.

La variable SIZE se encarga de obtener los metadatos relacionados con el tamaño del fichero con el servicio stat para imprimirlos con la directiva `--printf`.

Después, si WHAT no contiene un valor nulo, es decir, el usuario ha marcado alguna casilla del panel explicado anteriormente, se ejecuta la secuencia descrita:

Primero se eligen las últimas 8 líneas del valor de la variable con `tail`.

Después `pv` es el encargado de monitorizar la transferencia de datos a través del pipe, con los argumentos especifica que se trata de un tamaño de fichero, que la representación no será gráfica, si no numérica y que el intervalo de refresco será de 0.25 segundos.

Más adelante, `base64 -d` realiza la decodificación del bloque de datos pertinente.

`Tar` es el encargado de descomprimir el contenido después de ser decodificado.

La salida de error se "concatena" a la estándar y por último zenity muestra una ventana interactiva de progreso con un texto informativo que se cerrará automáticamente al terminar.