

Trabajo Práctico 1

Sistemas operativos

Segundo cuatrimestre 2012

Software para procesar archivos de logging

Apellido y nombre	Padrón
Alejandro Garcia Marra	
Miguel Torres	
Sebastián Bogado	
Ariel Barreiro	78648
Ignacio Garay	
Javier Choque	

16 de octubre de 2012

Índice

1. Hipótesis y aclaraciones globales	3
2. Problemas relevantes	3
2.1. BuscarV5: Lectura de expresiones regulares del archivo de patrones	3
2.2. ListarV5: Filtros para la consulta de archivos de resultado . . .	3
3. Archivo README	3
4. Listado de comandos y funciones	4
4.1. BuscarV5.sh	4
4.2. DetectaV5.sh	4
4.3. InstalaV5.sh	5
4.4. MirarV5.sh	5
4.5. StartD.sh	5
4.6. StopD.sh	5
4.7. LoguearV5.sh	5
4.8. MoverV5.sh	6
4.9. IniciarV5.sh	6
4.10. ListarV5.pl	6
5. Archivos	7
5.1. Archivo maestro de patrones	7
5.2. Archivos de resultados detallados	7
5.3. Archivos de resultados globales	7
5.4. Archivos de reporte	7
5.5. Archivos de posibles mensajes de logueo	7

1. Hipótesis y aclaraciones globales

2. Problemas relevantes

2.1. BuscarV5: Lectura de expresiones regulares del archivo de patrones

Según el formato del archivo de patrones, la expresión regular a aplicar se ubica en el segundo campo de cada registro del archivo, pero encerrado entre comillas simples. Al leer esta expresión regular, es necesario quitar las comillas simples antes de ingresarlas al comando grep, sino el comportamiento era incorrecto: se interpretaban las comillas simples como caracteres buscados. Se solucionó con una función en el script BuscarV5.sh que recorta el primer y último carácter de un string.

Otro problema relevante fue la lectura de los registros del archivo de patrones. Es necesario usar el comando read con el flag “-r” para que no interpretara caracteres de escape en las expresiones regulares.

2.2. ListarV5: Filtros para la consulta de archivos de resultado

El enunciado requería la opción de poder filtrar los resultados seleccionando los distintos patrones, ciclos y archivos. Quisimos mantener una forma flexible de filtrado pero sin requerir cargar todos los archivos en memoria en el script. Tomando eso en cuenta, se hace una primera iteración sobre los archivos de resultado para obtener las diferentes opciones de filtrados, que pueden ser combinadas de cualquier manera, y luego una segunda iteración para mostrar los resultados aplicando los filtros seleccionados.

3. Archivo README

Este es el README del trabajo práctico correspondiente al segundo cuatrimestre del año 2012 de la materia 75.08 Sistemas Operativos, grupo 7.

ARCHIVOS DEL PROYECTO

El proyecto consiste en un software para procesar archivos que contienen información de logueo provenientes de diferentes sistemas. El programa valida los archivos, aplica patrones de búsqueda y genera archivos de resultados con las coincidencias. También mueve los archivos procesados a otras carpetas de almacenamiento de archivos procesados con éxito. Permite efectuar consultas sobre lo procesado, y generar reportes.

Los archivos correspondientes al proyecto son los siguientes:

README

INSTALACION

Insertar el dispositivo de almacenamiento con el contenido del trabajo práctico.

Crear en el directorio corriente un directorio de trabajo.

Copiar el archivo *.tgz en esa carpeta.

Descomprimir el *.tgz de manera de generar un *.tar.

Extraer los archivos del tar.

REQUERIMIENTOS

Para la correcta instalación, se deberá tener instalado Bash versión 3 o superior, y Perl versión o superior.

INTRODUCCION

Para frenar la ejecución del demonio, se debe ejecutar el script StopD.sh ubicado en

4. Listado de comandos y funciones

4.1. BuscarV5.sh

Archivos de input: Los archivos que sean aceptados y que se encuentren en el directorio de aceptados, el archivo de configuración producido por el modulo de inicialización, y el archivo de patrones.

Archivos intermedios: No produce.

Archivos de output: los archivos de resultados detallados y globales para los patrones. Se guardaran en la carpeta de procesados.

Parámetros y opciones: No posee argumentos de ejecución ni opciones. Cuando se ejecuta, comienza a procesar los archivos.

Ejemplo de invocación: es automática por el módulo DetectarV5. No lleva parámetros, así que tan sólo es necesaria su ejecución de la forma:

\$BINDIR/BuscarV5.sh

4.2. DetectaV5.sh

Archivos de input:

Archivos intermedios:

Archivos de output:

Parámetros y opciones:

Ejemplo de invocación:

4.3. InstalarV5.sh

Archivos de input:
Archivos intermedios:
Archivos de output:
Parámetros y opciones:
Ejemplo de invocación:

4.4. MirarV5.sh

Archivos de input:
Archivos intermedios:
Archivos de output:
Parámetros y opciones:
Ejemplo de invocación:

4.5. StartD.sh

Archivos de input:
Archivos intermedios:
Archivos de output:
Parámetros y opciones:
Ejemplo de invocación:

4.6. StopD.sh

Archivos de input:
Archivos intermedios:
Archivos de output:
Parámetros y opciones:
Ejemplo de invocación:

4.7. LoguearV5.sh

Archivos de input:
Archivos intermedios:
Archivos de output:
Parámetros y opciones:
Ejemplo de invocación:

4.8. MoverV5.sh

Archivos de input:

Archivos intermedios:

Archivos de output:

Parámetros y opciones:

Ejemplo de invocación:

4.9. IniciarV5.sh

Archivos de input:

Archivos intermedios:

Archivos de output:

Parámetros y opciones:

Ejemplo de invocación:

4.10. ListarV5.pl

Archivos de input: Los archivos de resultados detallados y globales generados por el comando BuscarV5.sh que se encuentran en el directorio de procesados.

Archivos intermedios: No Produce

Archivos de output: Dependiendo de un parametro (-x), genera el informe de los archivos de resultado aplicando los filtros seleccionados en un archivo de salida en el directorio de reportes.

Parámetros y opciones:

Sin parámetros, la aplicación muestra información y filtros sobre los resultados globales.

```
# $BINDIR/ListarV5.pl -h
```

Uso: ListarV5 [opciones]

-g, -global

(default) Consultar los resultados globales.

-r, -resultado

Consultar los resultados detallados.

Implica no global.

-x, -salida

Grabar el informe en un archivo en lugar de imprimirlo en pantalla.

Ejemplo de invocación:

```
# $BINDIR/ListarV5.pl -r
```

5. Archivos

5.1. Archivo maestro de patrones

Nombre: patrones

Directorio: \$MAEDIR

Estructura: registros de la forma ID_PATRON (numérico), EXPR_REG (caracteres), ID_SISTEMA (caracteres), CONTEXTO (línea ó carácter), DESDE (numérico), HASTA (numérico), separados por comas.

5.2. Archivos de resultados detallados

Nombre: resultados.ID_PATRON

Directorio: \$PROCDIR

Estructura: registros de la forma CICLO_BUSQUEDA (numérico), NOMBRE_ARCHIVO (alfanumérico), NUM_REGISTRO (numérico), RESULTADO (alfanumérico), separados por la secuencia de caracteres “+ - # - +”.

5.3. Archivos de resultados globales

Nombre: rglobales.ID_PATRON

Directorio: \$PROCDIR

Estructura: registros de la forma CICLO_BUSQUEDA (numérico), NOMBRE_ARCHIVO (alfanumérico), EXPR_REG (caracteres), CONTEXTO (línea ó carácter), DESDE (numérico), HASTA (numérico), CANT_HALLAZGOS (numérico) separados por comas.

5.4. Archivos de reporte

Nombre: salida_YYYYMMDDHHMMSS

Directorio: \$REPODIR

Estructura: Es un archivo de texto libre, en donde la salida está dividida por patrón y los filtros aplicados en cada patrón. Por cada patrón hay un listado de 3 columnas de campo fijo, donde la primera es el ciclo (10 caracteres), la segunda es el archivo (25 caracteres) y la tercera es el registro RESULTADO del archivo de resultados detallados (sin límite).

5.5. Archivos de posibles mensajes de logueo

No especificado en el enunciado. Desarrollado por el equipo para almacenar los mensajes de logueo que usan los módulos, y su formato.

Nombre: ListaErrores

Tipo de archivo: permanente.

Directorio: \$BINDIR

Justificación: este archivo contiene los posibles mensajes de logueo, de forma que el log sea coherente al registrar mensajes con el mismo propósito provenientes de distintos módulos. Para usarlo, cada módulo debe referenciar el mensaje segun un identificador. Si el mensaje llevara nombres o variables, se debe escribir “%s” en los lugares del mensaje que correspondan, y luego pasar esas variables como parámetros extra al usar el módulo de logueo.

Por ejemplo, para el mensaje:

401 Inicio BuscarV5 - Ciclo Nro:%s - Cantidad de archivos:%s

se necesitan dos variables: número de ciclo y cantidad de archivos. Entonces, desde el módulo que lo usa, BuscarV5, se debe invocar al log de esta forma:

\$BINDIR/LoguearV5.sh -c 401 -f BuscarV5 -i I \$CICLO \$CANT_ARCHIVOS

Estructura: registros de la forma ID_MENSAJE (numerico), MENSAJE (alfanumérico), separados por un espacio.