

Linux

Comandos de Filtro, búsqueda y expresiones regulares

Introducción a Filtros

Un grupo de comandos que leen alguna entrada, realizan una transformación y escriben una salida.

» **find**

» **sed**

» **grep**

» **join**

» **egrep**

» **unique**

» **fgrep**

» **head, tail**

» **sort**

» **wc**

» **tr**

» **cut**

Expresiones Regulares

- **Las expresiones regulares son patrones que describen un conjunto de cadenas de caracteres.**
- **Se utilizan en:**
 - **operaciones de comparación.**
 - **búsquedas o sustituciones de gran complejidad.**
- **Las expresiones regulares se construyen como las expresiones aritméticas, utilizando operadores para combinar expresiones más pequeñas.**

Expresiones Regulares

El conjunto de metacaracteres para expresiones regulares es el siguiente:

\ ^ \$. [] { } | () * + ?

Expresiones Regulares

Operador Significado

C	un caracter no especial concuerda consigo mismo
\C	elimina significado especial de un caracter C. El \ escapa el significado especial
^	indica comienzo de línea
\$	indica final de la línea
.	(punto) un caracter, cualquiera menos fin de línea)

Expresiones Regulares

Operador Significado

[...]	un caracter cualquiera de los caracteres ... Acepta intervalos del tipo a-z, 0-9, A-Z
[^...]	un caracter distinto de ... Acepta intervalos del tipo a-z, 0-9, A-Z
R*	0, 1 o más ocurrencias de la E. R. R (repetición)
R1R2	la E. R. R1 seguida de la E. R. R2 (concatenación)

Expresiones Regulares

Exp. Reg.	Coincide con
a.b	aXb aab abb adb a9b ...
a..b	aXXb aaab abbb a6;b ...
[abc]	a b c (cadenas de un caracter)
[aA]	a A (cadenas de un caracter)
[aA][bB]	ab Ab aB AB (cadenas de dos caracteres)
[0123456789]	0 1 2 3 4 5 6 7 8 9

Expresiones Regulares

Exp. Reg.	Coincide con
[0-9]	0 1 2 3 4 5 6 7 8 9
[A-Za-z]	A B C ... Z a b c ... z
[0-9][0-9][0-9]	000 001 ... 009 010 ... 019 100 ... 999
[0-9]*	cadena vacía, 0 1 9 00 99 123 456 999 9999 ...
[0-9][0-9]*	0 1 9 00 99 123 456 999 9999 99999 99999999 ...
^.*\$	cualquier línea con más de un caracter

Expresiones Regulares Extendidas

- **Algunos comandos, como egrep o grep -E, aceptan Expresiones Regulares Extendidas.**
- **Comprenden las Expresiones Regulares Básicas más algunos operadores que permiten construcciones más complejas.**

Expresiones Regulares Extendidas

Operador	Significado
R+	1 o más ocurrencias de la Exp. Reg. R
R?	0 o una ocurrencia de la Exp. Reg. R, y no más
R{N}	N ocurrencias de la Exp. Reg. R
R{N,}	N o más ocurrencias de la Exp. Reg. R
R{,M}	0 o a lo sumo M ocurrencias de la Exp. Reg. R

Expresiones Regulares Extendidas

Operador	Significado
$R\{n,m\}$	N o más ocurrencias de la Exp. Reg. R, pero a lo sumo M
$R1 R2$	la Exp. Reg. R1 o la Exp. Reg. R2 (alternativa)
(R)	Exp. Reg. anidada
"R"	evita que los caracteres de la Exp. Reg. R sean interpretados por el shell

Expresiones Regulares Extendidas

- **Ejemplos**

- **[0-9]+**

- 0 1 9 00 99 123 456 999 9999 99999**
999999999 ..

- **[0-9]?**

- cadena vacía, 0 1 2 .. 9**

- **^a|b**

- a b**

Expresiones Regulares Extendidas

- **Ejemplos**

- **$(ab)^*$**

- cadena vacía, **ab abab ababab ...**

- **$^[0-9]?b$**

- b 0b 1b 2b .. 9b**

- **$([0-9]^+ab)^*$**

- cadena vacía, **1234ab 9ab9ab9ab
9876543210ab 99ab99ab ...**

Comando grep, egrep y fgrep

```
grep [ -e PATRON | -f ARCHIVO | PATRON  
] OPCIONES ARCHIVO ...
```

- recorre los archivos indicados extrayendo las líneas que coinciden con un patrón de cadena de caracteres.**

Comando grep, egrep y fgrep

- **G** el patrón es una expresión regular básica. Opción por defecto.
- **E** el patrón es una expresión regular extendida.
- **F** el patrón es una cadena fija.
- **Existen dos variantes:**
 - **egrep** similar a **grep -E**
 - **fgrep** igual a **grep -F**.

Comando grep, egrep y fgrep

- **Parámetros:**

- **c** sólo muestra la cantidad de líneas
- **e PATRON** usa el patrón indicado; útil cuando el patrón empieza con -
- **f ARCHIVO** obtiene los patrones del archivo, uno por línea
- **h** suprime indicación de archivo cuando se busca en varios archivos
- **i** ignora mayúsculas y minúsculas en el patrón y en los archivos

Comando grep, egrep y fgrep

- **Parámetros:**

- **n** **indicar número de línea**
- **v** **muestra las líneas que no coinciden con el patrón**
- **w** **hace coincidir el patrón como una palabra**
- **x** **el patrón coincide con toda la línea**

Comando find

- **find [DIRECTORIO...] [EXPRESION]**
 - Recorre un árbol de directorios buscando archivos según una expresión y muestra información sobre los archivos encontrados.

Comando find

- **Criterios de búsqueda, con patrones:**
 - **name PATRON** el nombre se ajusta con **PATRON**
 - **iname PATRON** idem, insensible a mayúsculas
 - **path PATRON** nombre con vía completa
 - **ipath PATRON** idem, insensible a mayúsculas
 - **regexp EXP** nombre se ajusta con expresión regular **EXP**
 - **iregexp EXP** idem, insensible a mayúsculas

Comando find

- **Criterios de búsqueda, con enlaces:**
 - **Iname PATRON** archivo es un enlace simbólico
 - **iname PATRON** idem, insensible a mayúsculas
- **Criterios de búsqueda, con tamaño:**
 - **size N[bckw]** de tamaño = N
 - **b** = bloques,
 - **c** = caracteres,
 - **k** = KB
 - **w** = palabras (2 bytes)
 - **empty** archivo vacío, regular o directorio

Comando find

- **Criterios de búsqueda, con fechas:**
 - **atime N** último acceso N*24 horas atrás
 - **ctime N** último cambio de estado N*24 horas atrás
 - **mtime N** última modificación N*24 horas atrás
 - **amin N** último acceso N minutos atrás
 - **cmin N** último cambio de estado N minutos horas atrás
 - **mmin N** última modificación N minutos atrás
 - **daystart** mide tiempos desde las 0 horas de hoy

Comando find

- **Criterios de búsqueda, con usuarios:**
 - **user UNAME** **archivo del usuario UNAME**
 - **group GNAME** **archivo del grupo GNAME**
 - **uid N** **archivo del usuario de número N**
 - **gid N** **archivo del grupo de número N**
 - **nouser** **archivo sin dueño asignado**
 - **nogroup** **archivo sin grupo asignado**

Comando find

- **Argumentos numéricos:**
 - **+N** **mayor que N**
 - **-N** **menor que N**
 - **N** **exactamente N**

Comando find

- **Acciones a realizar con archivos encontrados:**
 - **print**
 - imprime la ruta completa de los archivos encontrados.
 - **ls**
 - lista los archivos encontrados en formato **ls -dils**
 - **exec COMANDO { } \;**
 - ejecuta **COMANDO** sobre los archivos encontrados

Comando find

- **Ejemplos:**

- **find /etc -name '*.texto'**

- busca en el directorio indicado los archivos de nombre indicado.

- **find /usr/src -name '*.c' -size +100k -print**

- muestra nombres de archivos terminados en .c con tamaño mayor de 100 Kb.

- **find . -lname '*system.c'**

- busca enlaces simbólicos del nombre indicado.

Comando find

- **Ejemplos:**

- **find /bin -amin +2 -amin -6**

- **archivos que fueron leídos entre 2 y 6 minutos atrás.**
 - **La medición de tiempos se hace desde 24 horas atrás (1 día atrás), para medir tiempos desde las 0 horas del día de hoy, usar la opción -daystart.**

- **find ~ -daystart -type f -mtime 1**

- **archivos que fueron modificados ayer.**

- **find / -empty -exec rm -rf '{}' ';'**

- **busca archivos o directorios vacíos y los elimina.**

Comando sort

- **sort [OPCIONES] [ARCHIVO...]**
 - ordena, mezcla o compara todas las líneas de los archivos indicados o de la entrada estándar.
 - Tiene 3 modos de operación:
 - ordenar
 - verificar orden
 - mezclar archivos

Comando sort

- **Parámetros:**

- **c** **verifica si los archivos ya están ordenados**
- **m** **mezcla archivos ordenados en uno solo, ordenado**
- **b** **ignora blancos adelante en la ordenación**
- **d** **ordena considerando sólo letras, números y blancos**
- **f** **ordena como si todas fueran mayúsculas**
- **i** **ignora caracteres no imprimibles**
- **n** **ordena como valores numéricos y no como ASCII**
- **r** **invertir el sentido de la ordenación.**

Comando sort

- **Parámetros:**
 - **o arch** archivo de salida. puede ser el de entrada.
 - **t** usar el caracter indicado como separador de campos
 - **k POS1[,POS2]**
 - campos de ordenación, POS1 hasta POS2 inclusive; o hasta el final si no hay POS2
 - POS1 y POS2 son de la forma F.C
 - donde F es el número del campo y C es el primer caracter desde el comienzo del campo (para POS1) o desde el final del campo (para POS2).
 - si se omite C se ordena tomando en cuenta el primer caracter del campo.

Comando sort

Ejemplos

```
cat /etc/passwd | sort -t:
```

```
cat /etc/passwd | sort -t: +4 -f | cut -d: -f5
```

```
cat /etc/passwd | sort -t: +5 | cut -d: -f6
```

```
cat /etc/passwd | sort -t: -n -k3,5 | cut -d: -  
f4,5
```

Comando tr

- **tr [OPCION]... CONJ1 [CONJ2]**
 - **copia entrada en salida traduciendo, comprimiendo o borrando caracteres.**
 - **CONJ1 y CONJ2 definen un conjunto ordenado de caracteres a transformar.**

Comando tr

- **Parámetros:**
 - **c** reemplaza CONJ1 por su complemento, los caracteres no especificados en el conjunto.
 - **d** elimina los caracteres de CONJ1.
 - **s** comprime caracteres repetidos en CONJ1 en una sola ocurrencia.
 - **ds** primero borra según CONJ1 y después comprime según CONJ2.

Comando tr

- **CONJ1 y CONJ2**
 - **M-N** intervalo de caracteres desde el M al N (M anterior al N)
 - **0-9** equivale a 0123456789
- **La notación [:CLASE:] expande a los caracteres predefinidos en las clases:**
 - **[:alnum:]** letras y números.
 - **[:alpha:]** letras.
 - **[:blank:]** espacio.
 - **[:cntrl:]** caracteres de control.

Comando tr

- **Notación [CLASE:], cont.**
 - **[digit:]** números
 - **[graph:]** caracteres imprimibles, excluido el espacio.
 - **[lower:]** minúsculas
 - **[print:]** caracteres imprimibles, incluido el espacio.
 - **[punct:]** signos de puntuación.
 - **[space:]** espacio
 - **[upper:]** mayúsculas.
 - **[xdigit:]** dígitos hexadecimales.

Comando tr

- **La traducción se realiza cambiando el primer caracter de CONJ1 por el primer caracter de CONJ2, el 2o. de CON1 por el 2o. de CONJ2, etc.**
- **Los caracteres no indicados en CONJ1 pasan incambiados.**

Comando tr

Ejemplos:

```
tr abcde...xyz ABCDE...XYZ
```

```
tr a-z A-Z
```

```
tr '[:lower:]' '[:upper:]'
```

- convierten minúsculas en mayúsculas.

```
tr -d '\000'
```

- elimina caracteres nulos (octal 000).

Comando tr

Ejemplos:

tr -cs '[a-zA-Z0-9]' '[\n*]'

» **separa las palabras una por línea.**

tr -s '\n'

» **comprime caracteres nueva línea en uno solo (elimina líneas en blanco).**

Comando uniq

uniq [OPCION]... [ENTRADA [SALIDA]]

- muestra una única línea para una entrada ordenada, eliminando líneas repetidas sucesivas.
- Opcionalmente, puede mostrar solo líneas que aparecen una vez, o sólo líneas que aparecen varias veces.
- La entrada debe estar ordenada; si no lo está, puede usar sort -u para lograr un efecto similar.

Comando uniq

Parámetros:

- **f N** saltar N campos antes de verificar unicidad.
- **s N** saltar N caracteres antes de verificar unicidad.
- **c** indicar junto a cada línea el número de veces que aparece.
- **i** ignorar mayúsculas y minúsculas al comparar.
- **d** mostrar sólo líneas repetidas.
- **u** mostrar sólo líneas únicas.

Comando cut

cut -f CAMPOS [ARCHIVO]

cut -c COLUMNAS [ARCHIVO]

- extrae partes de cada línea de los archivos indicados o de la entrada estándar.
- Los campos y columnas se indican con números 1, 2,... o con intervalos n-m.
- Intervalos incompletos:
 - n » 1-n
 - m- » m hasta el último campo o columna.

Comando cut

cut -f CAMPOS [-d DELIM][ARCHIVO]

cut -c COLUMNAS [ARCHIVO]

- f** » campos numerados
- c** » columnas, ubicación posicional de caracteres en la línea
- d** » fija el caracter delimitador de campos, tabulador por defecto
- s** » con -f para no imprimir líneas que no contengan el delimitador

Comando cut

cut -f CAMPOS [-d DELIM][ARCHIVO]

cut -c COLUMNAS [ARCHIVO]

Ejemplos:

cut -d: -f1,3,5-6 /etc/passwd

cut -c1-15 /etc/passwd

ls -l | cut -c57-

- corta sólo el nombre del archivo (puede requerir ajuste en el valor de columna).

Ejercicio

Usar el archivo `/etc/passwd` para resolver las siguientes preguntas. En estos archivos pueden haber comentarios, líneas empezadas por `#`, que deben eliminarse de la búsqueda. Recordar que el carácter `#` es interpretado por el shell.

- a) eliminar los comentarios.
- b) extraer una línea que muestre directorio propio y shell del usuario `lcarambula`.
- c) listar los usuarios que usan el shell `/bin/bash`.
- d) listar los usuarios que no usan el shell `/bin/bash`.
- e) contar la cantidad de usuarios.

Ejercicio

Realizar las tareas indicadas, si es posible; si no lo es, indicar los motivos. Presentar el archivo `/etc/passwd` ordenado según los siguientes criterios:

- a) ordenado por nombre de login de usuario.**
- b) ordenado por número de usuario.**
- c) ordenado por número de grupo y dentro de cada grupo por número de usuario.**
- d) ordenado por número de grupo y dentro de cada grupo por campo de comentario.**

Ejercicio

Encontrar las 5 palabras más frecuentes en el texto `historia.txt`, e indicar la cantidad de veces que fue usada cada una.

Indicación: transformar primero todas las palabras a minúscula.