sed: Stream EDitor là một công cụ dùng để thao tác trực tiếp với văn bản như thay thế, xoá dòng, in ra một số dòng v.v.

## 1. Thay thế 1 chuỗi bằng 1 chuỗi khác (không có option):

$ sed 's/pattern/replace\_string/' file

hoặc

$ cat file | sed 's/pattern/replace\_string/'

## 2. Lưu các thay đổi vào tập tin (option –i)

Khi thực hiện lệnh thay thế trên, sed sẽ chỉ in ra các văn bản được thay thế, để có thể lưu chúng lại ta sử dụng tuỳ chọn -i:

$ sed -i 's/text/replace/' file

## 3. Thay thế tất cả sự xuất hiện của mẫu (hậu tố /g)

Nếu sử dụng cú pháp trên, sed sẽ chỉ thay thế sự xuất hiện đầu tiên của mẫu (pattern) trong mỗi dòng. Nếu muốn thay thế tất cả xuất hiện của mẫu trong văn bản, ta thêm tham số g vào cuối:

$ sed 's/pattern/replace\_string/g' file

Mặc định khi thêm hậu tố /g sẽ thay thế sự xuất hiện thứ 1 của mẫu đến cuối văn bản.

Nếu chỉ muốn thay thế sự xuất hiện thứ N cho đến cuối, ta sử dụng dạng /Ng như sau:

$ echo thisthisthisthis | sed 's/this/THIS/3g'

thisthisTHISTHIS

Nếu chỉ muốn thay thế sự xuất hiện tại vị trí thứ N của mẫu trong văn bản, ta sử dụng dạng /N như sau:

$ echo thisthisthisthis | sed 's/this/THIS/2'

thisTHISthisthis

## 4. Ký tự dấu phân cách (dấu :, |)

Ngoài dấu / như trong các ví dụ trên, ta có thể sử dụng các ký tự phân cách khác như sau:

$ sed 's:text:replace:g'

$ sed 's|text|replace|g'

Khi một ký tự phân cách xuất hiện bên trong mẫu, chúng ta phải thoát nó bằng các sử dụng tiền tố \ như sau:

$ sed 's|te\|xt|replace|g'

## 5. Xoá các dòng trống (không có option)

$ sed '/^$/d' file

/pattern/d sẽ xoá các dòng trùng khớp với pattern.

## 6. Xoá các dòng chứa một chuỗi bất kỳ (không có option)

Giả sử ta muốn xoá các dòng chứa chuỗi "mobile phoner"

$ sed 's/ [^.]\*mobile phones[^.]\*.//g' sentence.txt

## 7. Ký hiệu chuỗi trùng khớp

Trong sed, chúng ta có thể sử dụng & như 1 chuỗi trùng khớp cho các mẫu thay thế, theo cách này, chúng ta có thể sử dụng chuỗi trùng khớp này trong chuỗi thay thế.

Ở đây, biểu thức chính quy \w+ đối chiếu tất cả các từ trong văn bản. Sau đó, chúng ta thay thế nó với [&], & tương ứng với từ trùng khớp.

## 8. Ký hiệu đối chiếu chuỗi con (\1)

& tương ứng với chuỗi trùng khớp cho mẫu được cho. Chúng ta cũng có thể đối chiếu các chuỗi con của mẫu được cho.

$ echo this is digit 7 in a number | sed 's/digit ([0-9]\)/\1/'

this is 7 in a number

$ echo this is digit 7 in a number | sed 's/digit ([0-9]\)/\1/'

this is 7 in a number

Lệnh trên thay thế digit 7 với 7. Chuỗi con trùng khớp là 7. (pattern\) được dùng để đối chiếu chuỗi con. Mẫu này được bao đóng trong (), và được thoát với . Với chuỗi con đầu tiên trùng khớp, ký hiệu tương ứng là \1; với chuỗi con thứ 2, ký hiệu là \2, và cứ thế.

([a-z]+\) khớp với từ đầu tiên, và ([A-Z]+\) khớp với từ thứ 2. \1 và \2 được dùng cho việc tham chiếu đến chúng. Trong phần thay thế, thứ tự của chúng được đổi thành \2 \1 và, do vậy, nó xuất hiện với thứ tự ngược lại.

## 9. Kết hợp nhiều biểu thức

Thông thường để kết hợp nhiều lệnh sed với nhau, chúng ta thường sử dụng toán tử pipe (|) như sau:

$ sed 'expression' | sed 'expression'

Lệnh trên tương đương với:

$ sed 'expression; expression'

hoặc:

$ sed -e 'expression' -e 'expression'

## 10. Trích dẫn

Thông thường, các biểu thức của sed được trích dẫn bởi dấu ngoặc đơn ”. Nhưng dấu ngoặc kép “” cũng có thể dùng được. Dấu ngoặc kép mở rộng biểu thức bằng việc đánh giá chúng. Sử dụng ngoặc kép hữu dụng khi chúng ta muốn sử dụng 1 biến trong 1 biểu thức của sed.

## Tìm và Thay Thế Văn Bản Sử Dụng sed

Trước khi chạy câu lệnh sed chúng ta sẽ tạo một tập tin với tên là test.txt trên máy tính với nội dung như sau:

foo 1

sample text

Foo 2

sample text

foO 3

Câu lệnh dưới đây sẽ thực hiện việc tìm kiếm cho foo và thay thế bởi bar có trong trong test.txt

$ sed -e "s/foo/bar/g" test.txt

Bạn sẽ thấy terminal hiển thị kết quả:

bar 1

sample text

Foo 2

sample text

foO 3

Trong câu lệnh trên:

## Tùy chọn -e dùng để thêm vào biểu thức thực thi bởi sed (không có option -e vẫn cho kết quả như trên (?))

Biểu thức "s/foo/bar/g" là biểu thức regular expression dùng để thực hiện việc thay thế (subtitude) foo bởi bar và việc thay thế này được thực hiện cho tất cả các kết quả foo có trong test.txt

File test.txt là tập tin chứa nội dung văn bản

Bạn có thể bỏ qua tùy chọn /g trong biểu thức truyền vào câu lệnh sed:

$ sed -e "s/foo/bar/" test.txt

Sử dụng tùy chọn i trong biểu thức regular expression, nếu bạn muốn tìm kiếm và thay thế theo kiểu không phân biệt in hoa và in thường (nghĩa là cả foo, Foo, FoO... đều sẽ được thay thế bởi bar:

$ sed -e "s/foo/bar/i" test.txt

Bạn sẽ thấy terminal hiển thị kết quả:

bar 1

sample text

bar 2

sample text

bar 3

## TEXT parsing -SED

Command line options

$ sed [-options] [-f file] [input-files]

-f: tells sed to get commands from a file (sed script)

sed [-optins] ‘commands’ <input files>

. : bất kỳ kí tự nào

^ :bắt đầu ký tự trong dòng

$ :xuất hiện cuối dòng

\* : bất kì kí tự nào

## $ sed ‘p’ : in ra các dòng

## $ sed -n ‘p’ : in ra các dòng và không có sự lập đi lặp lại các dòng trùng nhau

## $ sed ‘s/pattern1/pattern2/’ : thay thế pattern1 thành pattern2 : chỉ đúng trong lần đầu tiên tìm thấy trong 1 dòng

## $ sed ‘s/pattern1/pattern2/g’ : thay thế tất cả nếu đúng pattern1 thì chuyển thành pattern2

## $ sed ‘s/\[sau dấu \ : là ký tự đặt biệt]pattern1/ \[ký tự đặt biệt]pattern2 /g’

## $ sed ‘s/[pattern1]//g’ : thay pattern1 thành không có gì

## $ sed ‘s/[pattern1]/ /g’ : thay pattern1 thành khoảng trắng

## $ sed ‘/^[pattern1]/d’ : bắt đầu bẳng pattern1 thì xoá nó đi

## $ sed -n ‘/[pattern1]/p’ : những dòng nào xuất hiện số pattern1 thì in ra

## $ sed -n ‘/^[137]/p’ : in ra những dòng có xuất hiện từ đầu tiên trong list [137]

## $ sed -n ‘/^[a-z]/p’ : in ra các dòng có kí tự đầu tiên là chữ cái (thuộc từ a-z) viết thường

## $ sed -n ‘/^[0-9]/p’ : in ra những dòng đầu tiên là kí tự số

## $ sed -n -e ‘/a/,/d/p’ : chỉ in ra từ a đến d là những dù có e f g h …

## $ sed ‘s/.\*/ &/’ :chèn khoảng trắng trước các kí tự

## $ sed ‘1d’ : xoá hàng đầu tiên

## $ sed ‘$d’ : xoá hàng cuối cùng

## $ sed -e ‘/^ $/d’ : xoá hết tất cả các dòng có khoản trắng

## $ sed ‘/^$/d’ : xoá những hàng không có ý nghĩa gì hết

## $ sed ‘/[a-z]\{20\}/p’ : in ra các dòng thoả đúng 20 từ

## $ sed ‘/^[a-z] \{1,19\}$/p’ : in ra các dòng đúng pattern là 1 đến 19 kí tự

sed advance:

file test

111111111111

222222222222

333333333333

$ cat test | sed -e ‘s/^[0-9]\{3\}/(&)/g’

(111)111111111

(222)222222222

(333)333333333

\{number\} : lấy bao nhiêu phần tử

& : kết quả nhận được

$ cat test | sed -e ‘s/^[0-9]\{3\}/(&)/g’ -e ‘s/)[0-9]\{3\}/&-/g’

(111)111-111111

(222)222-222222

(333)333-333333

định nghĩa một pattern:

\( pattern1 \) \( pattern2\) : pattern1 là do ta định nghĩa tương tự với pattern2

(phần 1)