实验86 Linux命令：sed命令

86.1  实验目的

1． 学会linux命令sed的使用方法；

86.2  实验要求

1． 掌握linux命令sed基本用法。

86.3  实验原理

86.3.1 sed命令

sed命令是一种流编辑器，它是文本处理中非常中的工具，能够完美的配合正则表达式使用，功能不同凡响。处理时，把当前处理的行存储在临时缓冲区中，称为“模式空间”（pattern space），接着用sed命令处理缓冲区中的内容，处理完成后，把缓冲区的内容送往屏幕。接着处理下一行，这样不断重复，直到文件末尾。文件内容并没有 改变，除非你使用重定向存储输出。sed主要用来自动编辑一个或多个文件；简化对文件的反复操作；编写转换程序等。

86.3.2 sed命令格式

sed [-hnV][-e<script>][-f<script文件>][文本文件]

86.3.3 sed**命令常用参数**

-e<script>或--expression=<script>：以选项中的指定的script来处理输入的文本文件； -f<script文件>或--file=<script文件>：以选项中指定的script文件来处理输入的文本文件；

-h或--help：显示帮助；

-n或--quiet或——silent：仅显示script处理后的结果；

-V或--version：显示版本信息。

86.3.4 sed动作说明

a\ 在当前行下面插入文本。

i\ 在当前行上面插入文本。

c\ 把选定的行改为新的文本。

d 删除，删除选择的行。

D 删除模板块的第一行。

s 替换指定字符。

h 拷贝模板块的内容到内存中的缓冲区。

H 追加模板块的内容到内存中的缓冲区。

g 获得内存缓冲区的内容，并替代当前模板块中的文本。

G 获得内存缓冲区的内容，并追加到当前模板块文本的后面。

l 列表不能打印字符的清单。

n 读取下一个输入行，用下一个命令处理新的行而不是用第一个命令。

N 追加下一个输入行到模板块后面并在二者间嵌入一个新行，改变当前行号码。

p 打印模板块的行。

P(大写) 打印模板块的第一行。

q 退出Sed。

b lable 分支到脚本中带有标记的地方，如果分支不存在则分支到脚本的末尾。

r file 从file中读行。

t label if分支，从最后一行开始，条件一旦满足或者T，t命令，将导致分支到带有标号的命令处，或者到脚本的末尾。

T label 错误分支，从最后一行开始，一旦发生错误或者T，t命令，将导致分支到带有标号的命令处，或者到脚本的末尾。

w file 写并追加模板块到file末尾。

W file 写并追加模板块的第一行到file末尾。

! 表示后面的命令对所有没有被选定的行发生作用。

= 打印当前行号码。

# 把注释扩展到下一个换行符以前。

86.3.5 sed替换标记

g 表示行内全面替换。

p 表示打印行。

w 表示把行写入一个文件。

x 表示互换模板块中的文本和缓冲区中的文本。

y 表示把一个字符翻译为另外的字符（但是不用于正则表达式）。

\1 子串匹配标记。

& 已匹配字符串标记

86.3.6 sed元字符集

^匹配行开始，如：/^sed/匹配所有以sed开头的行。

$ 匹配行结束，如：/sed$/匹配所有以sed结尾的行。

. 匹配一个非换行符的任意字符，如：/s.d/匹配s后接一个任意字符，最后是d。

\* 匹配0个或多个字符，如：/\*sed/匹配所有模板是一个或多个空格后紧跟sed的行。

[] 匹配一个指定范围内的字符，如/[ss]ed/匹配sed和Sed。

[^] 匹配一个不在指定范围内的字符，如：/[^A-RT-Z]ed/匹配不包含A-R和T-Z的一个字母开头，紧跟ed的行。

\(..\) 匹配子串，保存匹配的字符，如s/\(love\)able/\1rs，loveable被替换成lovers。

& 保存搜索字符用来替换其他字符，如s/love/\*\*&\*\*/，love这成\*\*love\*\*。

\< 匹配单词的开始，如:/\ 匹配单词的结束，如/love\>/匹配包含以love结尾的单词的行。

\> 匹配单词的结束，如/love\>/匹配包含以love结尾的单词的行。

x\{m\} 重复字符x，m次，如：/0\{5\}/匹配包含5个0的行。

x\{m,\} 重复字符x，至少m次，如：/0\{5,\}/匹配至少有5个0的行。

x\{m,n\} 重复字符x，至少m次，不多于n次，如：/0\{5,10\}/匹配5~10个0的行。

86.4  实验步骤

### 86.4.1 创建实验集群

登录大数据实验平台，选择启动实验86。

86.4.2  编写带参数的 sed命令(篇幅有限,列举6个)

连接OpenVPN后，使用Xshell软件ssh登录master服务器。

* 86.4.2.1 预准备数据文件test.txt

[root@master ~]# vi test.txt

[root@master ~]# cat test.txt

hi,this is sed command test file

linux world is so amazing

you will like it!

[root@master ~]#

* 86.4.2.2 简单的文本替换（没有修改原文件）

[root@master ~]# sed "s/will/WILL/" test.txt

hi,this is sed command test file

linux world is so amazing

you WILL like it!

* 86.4.2.3 文本替换，加-i参数，将替换结果应用于原文件

[root@master ~]# sed -i 's/like/LOVE/' test.txt

[root@master ~]# cat test.txt

hi,this is sed command test file

linux world is so amazing

you will LOVE it!

[root@master ~]#

* 48.4.2.4 关于后缀/g

[root@master ~]# echo "test1 test2 test3 test4 " | sed "s/test/TEST/g" ### /g默认表示替换所有符合条件的文本

TEST1 TEST2 TEST3 TEST4

[root@master ~]# echo "test1 test2 test3 test4 " | sed "s/test/TEST/3g" ### /Ng表示从第N个符合条件的开始替换

test1 test2 TEST3 TEST4

* 48.4.2.5 移除空白行

[root@master ~]# cat test.txt

hi,this is sed command test file

linux world is so amazing

you will LOVE it!

[root@master ~]# sed '/^$/d' test.txt ### /pattern/d 会移除匹配样式的行，^$行尾标记紧临表示空白行

hi,this is sed command test file

linux world is so amazing

you will LOVE it!

* 86.4.2.6 **&代表已经匹配的字符串**

[root@master ~]# echo this is a test | sed 's/\w\+/[&]/g' ### \w\+ 正则表达式表示匹配每一个单词，&对应于之前所匹配到的单词，使用[&]进行替换

[this] [is] [a] [test]

* 86.4.2.7 组合多个表达式

[root@master ~]# cat test.txt

hi,this is sed command test file

linux world is so amazing

you will LOVE it!

[root@master ~]# cat test.txt | sed 's/linux/LINUX/' | sed 's/sed/SED/'

hi,this is SED command test file

LINUX world is so amazing

you will LOVE it!

86.5  实验结果

* 86.4.2.1运行结果如下：

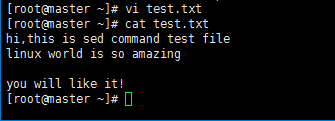


图86-1 运行结果

* 86.4.2.2运行结果如下：

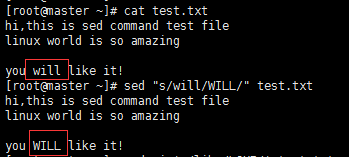


图86-2 运行结果

* 86.4.2.3运行结果如下：

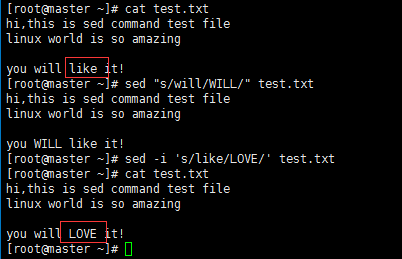


图86-3 运行结果

* 86.4.2.4运行结果如下：

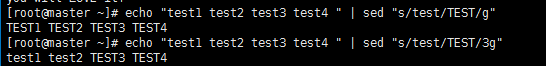


图86-4 运行结果

* 86.4.2.5运行结果如下：

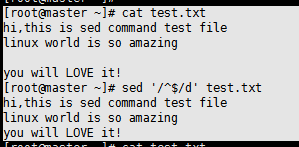


图86-5 运行结果

* 86.4.2.6运行结果如下：

C:\Users\dongguangming\AppData\Roaming\Tencent\Users\1056764180\QQ\WinTemp\RichOle\CKJ7`6]UCV210C_PRMF]H9E.png

图86-6 运行结果

* 86.4.2.7运行结果如下：

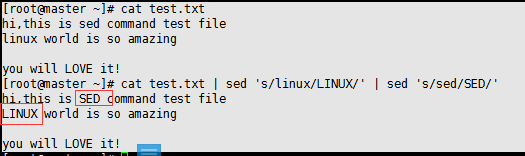


图86-7 运行结果