

课程目标

- 掌握sed的基本语法结构
- 熟悉sed常用的命令，如打印p，删除d，插入i等

Windows：

Linux：

==vim vi== gedit nano emacs

一、sed介绍

1. sed的工作流程



- 首先sed把当前正在处理的行保存在一个临时缓存区中（也称为模式空间），然后处理临时缓冲区中的行，完成后把该行发送到屏幕上。
- sed把每一行都存在临时缓冲区中，对这个**副本**进行编辑，所以不会修改原文件。
- 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。

2. sed使用方法

sed常见的语法格式有两种，一种叫==命令行==模式，另一种叫==脚本==模式。

2.1 命令行格式

- 格式

```
sed [option] 'sed的命令|地址定位' filename
```

说明：引用shell script中的变量应使用双引号，而非通常使用的单引号

option:

- e 进行多项编辑，即对输入行应用多条sed命令时使用
- n 取消默认的输出
- f 指定sed脚本的文件名
- r 使用扩展正则表达式
- i inplace，原地编辑（修改源文件）

- 常用命令和选项

参考文件：

```
[root@server shell06]# cat 2.txt
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
298374837483
172.16.0.254
10.1.1.1
```

p 打印行

d 删除行

```
[root@server ~]# sed -n 'p' a.txt
[root@server ~]# sed -n '1p' a.txt
[root@server ~]# sed -n '2p' a.txt
[root@server ~]# sed -n '1,5p' a.txt
[root@server ~]# sed -n '5,10p' a.txt
[root@server ~]# sed -n '$p' a.txt
[root@server ~]# sed '5p' a.txt
[root@server ~]# sed -n '5p' a.txt
[root@server ~]# sed '1d' a.txt
[root@server ~]# sed -n '1d' a.txt
[root@server ~]# sed '1d' a.txt
[root@server ~]# sed '1,5d' a.txt
[root@server ~]# sed '$d' a.txt
```

i\ 在当前行之前插入文本。多行时除最后一行外，每行末尾需用"\续行 vim-->O

a\ 在当前行后添加一行或多行。多行时除最后一行外，每行末尾需用"\续行 vim--> o

c\ 用此符号后的新文本替换当前行中的文本。多行时除最后一行外，每行末尾需用"\续行 整行替换

```
[root@server ~]# sed '$a99999' a.txt
[root@server ~]# sed 'a99999' a.txt
[root@server ~]# cat -n a.txt
[root@server ~]# sed '5chello world' a.txt
[root@server ~]# sed 'chello world' a.txt
[root@server ~]# cat -n a.txt
[root@server ~]# sed '1,5chello world' a.txt
```

```
[root@server ~]# sed 'i\
aaaaa\
bbbbbb\
88888' 1.txt
```

```
# sed '$a\
yyyyyy\
8888' 1.txt
```

```
[root@server ~]# sed '/^user01/c888888' 1.txt
[root@server ~]# sed '18chello world' 1.txt
```

命令2:

r 从文件中读取输入行

w 将所选的行写入文件

```
[root@server ~]# sed '3r /etc/hosts' 2.txt
[root@server ~]# sed '$r /etc/hosts' 2.txt
[root@server ~]# sed '/root/w a.txt' 2.txt
[root@server ~]# sed '/[0-9]{4}/w a.txt' 2.txt
[root@server ~]# sed -r '/([0-9]{1,3}\.){3}[0-9]{1,3}/w b.txt' 2.txt
```

! 对所选行以外的所有行应用命令，放到行数之后

```
[root@server ~]# sed -n '1!p' 1.txt
[root@server ~]# sed -n '4p' 1.txt
[root@server ~]# sed -n '4!p' 1.txt
[root@server ~]# cat -n 1.txt
[root@server ~]# sed -n '1,17p' 1.txt
[root@server ~]# sed -n '1,17!p' 1.txt
```

s 用一个字符串替换另一个

g 在行内进行全局替换

```
[root@server ~]# sed -n 's/root/ROOT/p' 1.txt
[root@server ~]# sed -n 's/root/ROOT/gp' 1.txt
[root@server ~]# sed -n 's/^#//gp' 1.txt
[root@server ~]# sed -n 's@/sbin/nologin@itcast@gp' a.txt
[root@server ~]# sed -n 's/\/sbin\/nologin/itcast/gp' a.txt
[root@server ~]# sed -n '10s#/sbin/nologin#itcast#p' a.txt
uucp:x:10:14:uucp:/var/spool/uucp:itcast
[root@server ~]# sed -n 's@/sbin/nologin@itcastheima@p' 2.txt
```

注意：搜索替换中的分隔符可以自己指定

```
[root@server ~]# sed -n '1,5s/^#/#/p' a.txt      注释掉文件的1-5行内容
#root:x:0:0:root:/root:/bin/bash
#bin:x:1:1:bin:/bin:/sbin/nologin
#daemon:x:2:2:daemon:/sbin:/sbin/nologin
#adm:x:3:4:adm:/var/adm:/sbin/nologin
#lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
```

& 保存查找串以便在替换串中引用 \(\)

```
[root@server ~]# sed -n '/root/p' a.txt
root:x:0:0:root:/root:/bin/bash
[root@server ~]# sed -n 's/root/#&/p' a.txt
#root:x:0:0:root:/root:/bin/bash
```

```
# sed -n 's/^root/#&/p' passwd  注释掉以root开头的行
# sed -n -r 's/^root|^stu/#&/p' /etc/passwd 注释掉以root开头或者以stu开头的行
# sed -n '1,5s/^[a-z].*/#&/p' passwd  注释掉1~5行中以任意小写字母开头的行
# sed -n '1,5s/^#/#/p' /etc/passwd  注释1~5行
```

或者

```
sed -n '1,5s/^#/#/p' passwd 以空开头的加上#
sed -n '1,5s/^#//p' passwd 以#开头的替换成空
```

```
[root@server ~]# sed -n '/^root/p' 1.txt
[root@server ~]# sed -n 's/^root/#&/p' 1.txt
[root@server ~]# sed -n 's/^(^root\)/#\1/p' 1.txt
[root@server ~]# sed -nr '/^root|^stu/p' 1.txt
[root@server ~]# sed -nr 's/^root|^stu/#&/p' 1.txt
```

= 打印行号

```
# sed -n '/bash$/=' passwd 打印以bash结尾的行的行号
# sed -ne '/root/=' -ne '/root/p' passwd
# sed -n '/nologin$/=;/nologin$/p' 1.txt
# sed -ne '/nologin$/=' -ne '/nologin$/p' 1.txt
```

综合运用：

```
[root@server ~]# sed -n '1,5s/^#/#/p' 1.txt
#root:x:0:0:root:/root:/bin/bash
#bin:x:1:1:bin:/bin:/sbin/nologin
#daemon:x:2:2:daemon:/sbin:/sbin/nologin
#adm:x:3:4:adm:/var/adm:/sbin/nologin
#lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
```

```
[root@server ~]# sed -n '1,5s/\(^\\)/#\1/p' 1.txt
#root:x:0:0:root:/root:/bin/bash
#bin:x:1:1:bin:/bin:/sbin/nologin
#daemon:x:2:2:daemon:/sbin:/sbin/nologin
#adm:x:3:4:adm:/var/adm:/sbin/nologin
#lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
```

选项: -e -r -i

-e 多项编辑

-r 扩展正则

-i 修改原文件

```
[root@server ~]# sed -ne '/root/p' 1.txt -ne '/root/='
root:x:0:0:root:/root:/bin/bash
1
[root@server ~]# sed -ne '/root/=' -ne '/root/p' 1.txt
1
root:x:0:0:root:/root:/bin/bash
```

在1.txt文件中的第5行的前面插入“hello world”;在1.txt文件的第8行下面插入“哈哈哈哈”

```
[root@server ~]# sed -e '5ihello world' -e '8a哈哈哈哈' 1.txt -e '5=;8='
```

过滤vsftpd.conf文件中以#开头和空行:

```
[root@server ~]# grep -Ev '^#|^$' /etc/vsftpd/vsftpd.conf
[root@server ~]# sed -e '/^#/d' -e '/^$/d' /etc/vsftpd/vsftpd.conf
[root@server ~]# sed '/^#/d;/^$/d' /etc/vsftpd/vsftpd.conf
[root@server ~]# sed -r '/^#|^$/d' /etc/vsftpd/vsftpd.conf
```

过滤smb.conf文件中生效的行:

```
[root@server shell06]# sed -e '/^#/d' -e '/^;/d' -e '/^$/d' -e '/^\\t$/d' -e
'/^\\t#/d' smb.conf
[root@server shell06]# sed -r '/^(#|;|\\t#|\\t$)/d' smb.conf
[root@server shell06]# sed -e '/^#/d' -e '/^;/d' -e '/^$/d' -e '/^\\t$/d' -e
'/^\\t#/' smb.conf
```

```
[root@server ~]# grep '^^[a-z]' 1.txt
[root@server ~]# sed -n '/^[a-z]/p' 1.txt
```

过滤出文件中的IP地址:

```
[root@server ~]# grep -E '([0-9]{1,3}\\.){3}[0-9]{1,3}' 1.txt
192.168.0.254
[root@server ~]# sed -nr '/([0-9]{1,3}\\.){3}[0-9]{1,3}/p' 1.txt
192.168.0.254
```

```
[root@server ~]# grep -o -E '([0-9]{1,3}\\.){3}[0-9]{1,3}' 2.txt
10.1.1.1
10.1.1.255
255.255.255.0
```

```
[root@server ~]# sed -nr '/([0-9]{1,3}\\.){3}[0-9]{1,3}/p' 2.txt
10.1.1.1
10.1.1.255
255.255.255.0
```

过滤出ifcfg-eth0文件中的IP、子网掩码、广播地址

```
[root@server shell06]# grep -Eo '([0-9]{1,3}\.){3}[0-9]{1,3}' ifcfg-eth0
10.1.1.1
255.255.255.0
10.1.1.254
[root@server shell06]# sed -nr '/([0-9]{1,3}\.){3}[0-9]{1,3}/p' ifcfg-eth0|cut -d'=' -f2
10.1.1.1
255.255.255.0
10.1.1.254
[root@server shell06]# sed -nr '/([0-9]{1,3}\.){3}[0-9]{1,3}/p' ifcfg-eth0|sed -n 's/[A-Z=]//gp'
10.1.1.1
255.255.255.0
10.1.1.254

[root@server shell06]# ifconfig eth0|sed -n '2p'|sed -n 's/[:a-z]//gp'|sed -n 's/ /\n/gp'|sed '/^$/d'
10.1.1.1
10.1.1.255
255.255.255.0
[root@server shell06]# ifconfig | sed -nr '/([0-9]{1,3}\.){3}[0-9]{1,3}/p' | head -1|sed -r 's/([a-z:]|[A-Z/t])//g'|sed 's/ /\n/g'|sed '/^$/d'

[root@server shell06]# ifconfig eth0|sed -n '2p'|sed -n 's/.*addr:(.*) Bcast:(.*) Mask:(.*)/\1\n\2\n\3/p'
10.1.1.1
10.1.1.255
255.255.255.0

-i 选项 直接修改原文件
# sed -i 's/root/ROOT/;s/stu/STU/' 11.txt
[root@server ~]# sed -i '17{s/YUNWEI/yunwei/;s#/bin/bash#/sbin/nologin#}' 1.txt
[root@server ~]# sed -i '1,5s/^/#&/' a.txt
注意:
-ni 不要一起使用
p命令 不要再使用-i时使用
```

• 总结

sed 选项 =='==sed==命令==或者==正则表达式==或者==地址定位===='== 文件名

地址用于决定对哪些行进行编辑。地址的形式可以是数字、正则表达式、或二者的结合。如果没有指定地址，sed将处理输入文件的所有行。

x	指定x行号	sed -n '5p' 1.txt
x,y	指定x到y行号	sed -n '1,5p' 1.txt
/key/	查询包含关键字的行	sed -n '/root/p' 1.txt
/key1/,/key2/	匹配包含两个关键字之间的行	sed -n '/^adm/,^mysql/p' 1.txt
/key/,x	从匹配关键字的行开始到文件第x行之间的行（包含关键字所在行）	sed -n '/^lp/,7p'
x,/key/	从第x行开始到与关键字的匹配行之间的行	
x,y!	不包含x到y行	

```
[root@server ~]# sed -n '/bash$/!p' 1.txt
```

注意：sed使用的正则表达式是括在斜杠线"/"之间的模式。

//以下命令是找出以lp开头或者以mail开头的行

- 其他命令讲解

y命令

该命令与UNIX/Linux中的tr命令类似，字符按照一对一的方式从左到右进行转换。

正则表达式元字符对y命令不起作用。与s命令的分隔符一样，斜线可以被替换成其它的字符。

```
s/xxx/xxx/
y/xxx/xxx/
# sed '39,41y/stu/STU/' /etc/passwd
# sed '39,41y/stu:x/STU@%/' /etc/passwd
```

q 退出

```
# sed '5q' 1.txt
# sed '/mail/q' 1.txt
# sed -r '/^yunwei|^mail/q' 1.txt
[root@server ~]# sed -n '/bash$/p;10q' 1.txt
ROOT:x:0:0:root:/root:/bin/bash
```

2.2 脚本格式

- 用法

```
# sed -f scripts.sed file      //使用脚本处理文件
建议使用 ./sed.sh file
```

脚本的第一行写上

```
#!/bin/sed -f
1,5d
s/root/hello/g
3i777
5i888
a
p
```

- 注意事项

- 1) 脚本文件是一个sed的命令行清单。'commands'
- 2) 在每行的末尾不能有任何空格、制表符（tab）或其它文本。
- 3) 如果在一行中有多个命令，应该用分号分隔。
- 4) 不需要且不可用引号保护命令
- 5) #号开头的行为注释

- 示例

```
# cat passwd
stu3:x:509:512::/home/user3:/bin/bash
stu4:x:510:513::/home/user4:/bin/bash
stu5:x:511:514::/home/user5:/bin/bash
```

```
# cat sed.sh
#!/bin/sed -f
2a\
*****
2,$s/stu/user/
$a\
we inster new line
s/^[a-z].*/#&/

[root@server ~]# cat 1.sed
#!/bin/sed -f
3a*****
$chelloworld
1,3s/^\#&/

[root@server ~]# sed -f 1.sed -i 11.txt
[root@server ~]# cat 11.txt
#root:x:0:0:root:/root:/bin/bash
#bin:x:1:1:bin:/bin:/sbin/nologin
#daemon:x:2:2:daemon:/sbin:/sbin/nologin
*****
adm:x:3:4:adm:/var/adm:/sbin/nologin
helloworld
```

3. sed和正则的综合运用

1、正则表达式必须以"/"前后规范间隔

例如: `sed '/root/d' file`

例如: `sed '/^root/d' file`

2、如果匹配的是扩展正则表达式，需要使用-r选来扩展sed

`grep -E`

`sed -r`

`+ ? () {n,m} | \d`

注意:

在正则表达式中如果出现特殊字符(`^$.*/[]`),需要以前导 `"\"` 号做转义

eg: `sed '/\ $foo/p' file`

3、逗号分隔符

例如: `sed '5,7d' file` 删除5到7行

例如: `sed '/root/,/ftp/d' file`

删除第一个匹配字符串"root"到第一个匹配字符串"ftp"的所有行本行不找 循环执行

4、组合方式

例如: `sed '1,/foo/d' file` 删除第一行到第一个匹配字符串"foo"的所有行

例如: `sed '/foo/,+4d' file` 删除从匹配字符串"foo"开始到其后四行为止的行

例如: `sed '/foo/,~3d' file` 删除从匹配字符串"foo"开始删除到3的倍数行 (文件中)

例如: `sed '1~5d' file` 从第一行开始删每五行删除一行

例如: `sed -nr '/foo|bar/p' file` 显示配置字符串"foo"或"bar"的行

例如: `sed -n '/foo/,/bar/p' file` 显示匹配从foo到bar的行

例如: `sed '1~2d' file` 删除奇数行

例如: `sed '0~2d' file` 删除偶数行 `sed '1~2!d' file`

5、特殊情况

例如: `sed '$d' file` 删除最后一行

例如: `sed '1d' file`

删除第一行

6、其他:

`sed 's/./ /' a.txt`

删除每一行中的第一个字符

`sed 's/././2' a.txt`

删除每一行中的第二个字符

`sed 's/././N' a.txt`

从文件中第N行开始, 删除每行中第N个字符 (N>2)

`sed 's/.$//' a.txt`

删除每一行中的最后一个字符

```
[root@server ~]# cat 2.txt
```

1 a

2 b

3 c

4 d

5 e

6 f

7 u

8 k

9 o

```
[root@server ~]# sed '/c/,~2d' 2.txt
```

1 a

2 b

5 e

6 f

7 u

8 k

9 o

4. 课堂练习

1. 将任意数字替换成空或者制表符
2. 去掉文件1-5行中的数字、冒号、斜杠
3. 匹配root关键字替换成hello itcast, 并保存到test.txt文件中
4. 删除vsftpd.conf、smb.conf、main.cf配置文件里所有注释的行及空行 (不要直接修改原文件)
5. 使用sed命令截取自己的ip地址
6. 使用sed命令一次性截取ip地址、广播地址、子网掩码
7. 注释掉文件的2-3行和匹配到以root开头或者以ftp开头的行

1、将文件中任意数字替换成空或者制表符

2、去掉文件1-5行中的数字、冒号、斜杠

3、匹配root关键字的行替换成hello itcast, 并保存到test.txt文件中

4、删除vsftpd.conf、smb.conf、main.cf配置文件里所有注释的行及空行 (不要直接修改原文件)

5、使用sed命令截取自己的ip地址

```
# ifconfig eth0|sed -n '2p'|sed -n 's/.*addr://pg'|sed -n 's/Bcast.*//gp'
```

10.1.1.1

```
# ifconfig eth0|sed -n '2p'|sed 's/.*addr://g'|sed 's/ Bcast:.*//g'
```

6、使用sed命令一次性截取ip地址、广播地址、子网掩码

```
# ifconfig eth0|sed -n '2p'|sed -n 's#.*addr: \(.*\) Bcast: \(.*\) Mask: \
```

```
(.*)# \1\n\2\n\3#p'
```

10.1.1.1

10.1.1.255

255.255.255.0

7、注释掉文件的2-3行和匹配到以root开头或者以ftp开头的行


```
# sed -nr '2,3s/^/#&/p;s/^ROOT|^ftp/#&/p' 1.txt
#ROOT:x:0:0:root:/root:/bin/bash
#bin:x:1:1:bin:/bin:/sbin/nologin
#3daemon:x:2:2:daemon:/sbin:/sbin/nologin

# sed -ne '1,2s/^/#&/gp' a.txt -nre 's/^lp|^mail/#&/gp'
# sed -nr '1,2s/^/#&/gp;s/^lp|^mail/#&/gp' a.txt
```

二、课后作业

1、写一个初始化系统的脚本

1) 自动修改主机名 (如: ip是192.168.0.88, 则主机名改为server88.itcast.cc)

a. 更改文件非交互式 sed

/etc/sysconfig/network

b. 将本主机的IP截取出来赋值给一个变量ip;再然后将ip变量里以.分割的最后一位赋值给另一个变量ip1

2) 自动配置可用的yum源

3) 自动关闭防火墙和selinux

2、写一个搭建ftp服务的脚本, 要求如下:

1) 不支持本地用户登录 local_enable=NO

2) 匿名用户可以上传 新建 删除 anon_upload_enable=YES anon_mkdir_write_enable=YES

3) 匿名用户限速500KBps anon_max_rate=500000

仅供参考:

```
#!/bin/bash
ipaddr=$(ifconfig eth0|sed -n '2p'|sed -e 's/.*inet addr: \(.*\) Bcast.* /\1/g')
iptail=$(echo $ipaddr|cut -d'.' -f4)
ipremote=192.168.1.10
#修改主机名
hostname server$iptail.itcast.com
sed -i "/HOSTNAME/CHOSTNAME=server$iptail.itcast.com" /etc/sysconfig/network
echo "$ipaddr server$iptail.itcast.cc" >>/etc/hosts
#关闭防火墙和selinux
service iptables stop
setenforce 0 >/dev/null 2>&1
sed -i '/^SELINUX=/cSELINUX=disabled' /etc/selinux/config
#配置yum源(一般是内网源)
#test network
ping -c 1 $ipremote > /dev/null 2>&1
if [ $? -ne 0 ];then
    echo "你的网络不通, 请先检查你的网络"
    exit 1
else
    echo "网络ok."
fi
cat > /etc/yum.repos.d/server.repo << end
[server]
name=server
baseurl=ftp://$ipremote
enabled=1
gpgcheck=0
end
```

#安装软件

```
read -p "请输入需要安装的软件，多个用空格隔开：" soft
yum -y install $soft &>/dev/null
```

#备份配置文件

```
conf=/etc/vsftpd/vsftpd.conf
\cp $conf $conf.default
```

#根据需求修改配置文件

```
sed -ir '/^#|^$/d' $conf
sed -i '/local_enable/c\local_enable=NO' $conf
sed -i '$a anon_upload_enable=YES' $conf
sed -i '$a anon_mkdir_write_enable=YES' $conf
sed -i '$a anon_other_write_enable=YES' $conf
sed -i '$a anon_max_rate=512000' $conf
```

#启动服务

```
service vsftpd restart &>/dev/null && echo"vsftpd服务启动成功"
```

#测试验证

```
chmod 777 /var/ftp/pub
cp /etc/hosts /var/ftp/pub
```

#测试下载

```
cd /tmp
lftp $ipaddr <<end
cd pub
get hosts
exit
end
```

```
if [ -f /tmp/hosts ];then
    echo "匿名用户下载成功"
    rm -f /tmp/hosts
else
    echo "匿名用户下载失败"
fi
```

#测试上传、创建目录、删除目录等

```
cd /tmp
lftp $ipaddr << end
cd pub
mkdir test1
mkdir test2
put /etc/group
rmdir test2
exit
end
```

```
if [ -d /var/ftp/pub/test1 ];then
    echo "创建目录成功"
    if [ ! -d /var/ftp/pub/test2 ];then
        echo "文件删除成功"
    fi
else
    if [ -f /var/ftp/pub/group ];then
        echo "文件上传成功"
    else
        echo "上传、创建目录删除目录部ok"
    fi
fi
[ -f /var/ftp/pub/group ] && echo "上传文件成功"
```

