

双星教育
Linux 系统管理员

awk sed 语法篇

Terry Tsang
曾绍坤
signmem@netease.com

sed awk 作用

awk 数据打印 具有编程语法

sed 数据打印 文件内容修改 文件另存为 文件合并

awk

awk 的作用

```
awk '/char1/' filename
```

显示域

```
awk '{ print $1,$2 }' filename
```

```
awk '{ print $1,$5 }' filename
```

```
awk '{ print $0 }' filename
```

重定义域

```
awk '{ print "this is the new line " $1 " and " $2 }' filename
```

转义字符

date | awk '{ print "Month: " \$2 "\nYear: ", \$6}'

\b 倒格

\f 换页

\n 换行

\r 回车

\t 制表符

\047 十进制47，代表单引号

\char 任意字符

打印表头表尾

```
awk ' BEGIN { print "==" } { print $1 } ' filename
```

```
awk ' { print $1 } END { print "--" } ' filename
```

自定义分隔符

```
awk 'BEGIN { FS=":" } { print $1 "\t" $5 }' /etc/passwd
```

```
awk -F: '{ print $1 "\t" $5 }' /etc/passwd
```

练习使用 `awk` 现实 `eth0` ip 地址

awk 变量及判断

```
awk 'BEGIN { a=123 } {print a, $1}' filename
```

```
awk -F: '{ if ( $1 ~ "root" ) print $0 }' filename
```

```
awk -F: '{ if ( $1 ~ "^my" ) print $0 }' filename
```

```
awk -F: '{ if ( $1 ~ "root" ) print $0 ; else print $2 }' filename
```

```
awk 'BEGIN { a=123 } { if ($1 ~ a) print a, $1}' filename
```


多次判断

```
awk '{ if ($2 ~ "root" && $4 ~ "new" print ($1,$2) }' filename
```

```
awk '{ if ($2 ~ "root" || $4 ~ "new" print ($1,$2) }' filename
```

分隔符

awk -F:

awk 'BEGIN {FS=:} '

常用语法

NF 计算列数量

\$NF 最后一列

NR 计算行号

数据截取定义符

RS 行分隔符

FS 列分隔符

数据打印定义符

OFS 列分隔符

ORS 行分隔符

循环

```
tail -3 /etc/passwd | awk ' BEGIN { FS="\n"; RS=""; ORS="" } { x=1 ;  
    while ( x<NF )    { print $x "\t" ; x++ }    print $NF "\n" }'
```

printf 格式化

%d, %i 整数

%e, %E 浮点数

%s 字符

%u 正整数

awk -F: '{print f "%15-s %10d %20s" \$1, \$3, \$7}'

打印

p 命令打印

```
sed '/bash/p' /etc/passwd
```

-n 与 --quiet

```
sed -n '/bash/p' /etc/passwd
```

```
sed --quiet '/bash/p' /etc/passwd
```

删除

d 用于删除

sed '3d' filename

sed '3,\$d' filename

sed '3,5d' filename

sed '\$d' filename

sed '/character/d' filename

替换

s 用于替换

g 用于指定行尾

```
sed 's/char1/char2/g' filename
```

```
sed -n 's/^char/char2/p' filename
```

```
sed 's[0-9][0-9]$/&.5/' filename
```

```
sed -n 's/char1/char2/gp' filename
```

```
sed -n 's/(May\)host\domain/p' filename
```


复合条件

```
sed -e '1,3d' -e 's/char1/char2/' filename
```

```
sed -e 's/char1/char2' -e '/char3/d' filename
```

文件修改

r 文件追加

sed '/char1/r newfile' filename

R 文件追加(每匹配一次追加一行)

sed '/char/'R newfile' filename

w 另存为

sed -n '/char/w newfile2' filename

a 字符追加(匹配后)

sed '/^char1 /a\new world' filename

i 文字追加(匹配前)

sed '/^char1 /a\new world' filename

c 字符覆盖

sed '/char/'c new' filename

练习

尝试通过 **awk** 截取 **IP** 地址

每分钟截取远程登陆用户 **IP** 及用户名并记录在 **/var/log/relogin** 中

每分钟检测系统是否打开新端口并记录到 **/var/log/port** 中

获得可以登陆的用户名及家目录

每小时检测硬盘空间一次如果磁盘空间不足 **80%** 自动发送邮件给管理员