**常用选项**

　　-E ：开启扩展（Extend）的正则表达式。

　　-i ：忽略大小写（ignore case）。

　　-v ：反过来（invert），只打印没有匹配的，而匹配的反而不打印。

　　-n ：显示行号

　　-w ：被匹配的文本只能是单词，而不能是单词中的某一部分。

　　-c ：显示总共有多少行被匹配到了，而不是显示被匹配到的内容，注意如果同时使用-cv选项是显示有多少行没有被匹配到。

　　-o ：只显示被模式匹配到的字符串。

　　--color :将匹配到的内容以颜色高亮显示。

　　-A  n：显示匹配到的字符串所在的行及其后n行，after

　　-B  n：显示匹配到的字符串所在的行及其前n行，before

　　-C  n：显示匹配到的字符串所在的行及其前后各n行，context

**基本正则模式匹配**

**匹配字符：**

　　　　　　. ：任意一个字符。

　　　　　　[abc] ：表示匹配一个字符

　　　　　　[a-Z] ：表示匹配一个字符

　　　　　　[^123] ：匹配一个字符，这个字符是除了1、2、3以外的所有字符。

　　　　　　对于一些常用的字符集，系统做了定义：

　　　　　　[a-Z] 等价于 [[:alpha:]]

　　　　　　[0-9] 等价于 [[:digit:]]

　　　　　　[a-Z0-9] 等价于 [[:alnum:]]

　　　　　　tab,space 等空白字符 [[:space:]]

　　　　　　[A-Z] 等价于 [[:upper:]]

　　　　　　[a-z] 等价于 [[:lower:]]

　　　　　　标点符号 [[:punct:]]

**匹配次数：**

　　　　　　\{m,n\} ：匹配其前面出现的字符至少m次，至多n次。  
　　　　　　\? ：匹配其前面出现的内容0次或1次，等价于\{0,1\}。  
　　　　　　\* ：匹配其前面出现的内容任意次，等价于\{0,\}，所以 ".\*" 表述任意字符任意次，即无论什么内容全部匹配。

**位置锚定：**

　　　　　　^ ：锚定行首

　　　　　　$ ：锚定行尾。技巧："^$"用于匹配空白行。

　　　　　　\b或\<：锚定单词的词首。如"\blike"不会匹配alike，但是会匹配liker

　　　　　　\b或\>：锚定单词的词尾。如"\blike\b"不会匹配alike和liker，只会匹配like

　　　　　　\B ：与\b作用相反。

**分组及引用：**

　　　　　　\(string\) ：将string作为一个整体方便后面引用

　　　　　　　　\1 ：引用第1个左括号及其对应的右括号所匹配的内容。

　　　　　　　　\2 ：引用第2个左括号及其对应的右括号所匹配的内容。

　　　　　　　　\n ：引用第n个左括号及其对应的右括号所匹配的内容。

**扩展正则模式匹配**

**匹配字符：**

　　　　　　跟基本正则用法一样

**匹配次数：**

　　　　　　{m,n} ：匹配其前面出现的字符至少m次，至多n次。  
　　　　　　? ：匹配其前面出现的内容0次或1次，等价于\{0,1\}。  
　　　　　　\* ：匹配其前面出现的内容任意次，等价于\{0,\}，所以 ".\*" 表述任意字符任意次，即无论什么内容全部匹配。

　　　　　　+：匹配前面出现的内容1次或1次以上

**位置锚定：**

　　　　　　跟基本正则用法一样

**分组及引用：**

　　　　　　(string) ：将string作为一个整体方便后面引用

　　　　　　　　\1 ：引用第1个左括号及其对应的右括号所匹配的内容。

　　　　　　　　\2 ：引用第2个左括号及其对应的右括号所匹配的内容。

　　　　　　　　\n ：引用第n个左括号及其对应的右括号所匹配的内容。

**扩展增用法：**

　　　　　　str | num：匹配 str 或 num 内容

1、搜索/var/log/secure下，失败的log。

[复制代码](javascript:void(0);)

[root@maintain data]# grep -i 'failed' /var/log/secure

Sep 24 13:25:11 localhost login: pam\_systemd(login:session): Failed to release session: Interrupted system call

Sep 27 02:58:13 maintain login: FAILED LOGIN 1 FROM tty1 FOR (unknown), User not known to the underlying authentication module

Sep 27 06:45:29 maintain sshd[1115]: pam\_systemd(sshd:session): Failed to release session: Interrupted system call

Sep 27 06:45:32 maintain sshd[1093]: pam\_systemd(sshd:session): Failed to release session: Interrupted system call

Sep 27 07:03:45 maintain sshd[1138]: pam\_systemd(sshd:session): Failed to release session: Interrupted system call

Sep 27 07:40:19 maintain sshd[1188]: pam\_systemd(sshd:session): Failed to release session: Interrupted system call

[复制代码](javascript:void(0);)

2、在/etc/selinux/config下，忽略#号开头和空白行

[root@maintain data]# egrep -v '^#|^$' /etc/selinux/config

SELINUX=disabled

SELINUXTYPE=targeted

3、只匹配Gogo的行 和 只匹配Gogo的单词

[复制代码](javascript:void(0);)

[root@maintain data]# cat reg.txt

Zhang Dandan 41117397 :250:100:175

Zhang Xiaoyu 390320151 :155:90:201

Meng Feixue 80042789 :250:60:50

Wu waiwai 70271111 :250:80:75

Liu Bingbing 41117483 :250:100:175

Wang Xiaoai 3515064655 :50:95:135

Zi Gege 1986787350 :250:168:200

Li Youjiu 918391635 :175:75:300

Lao Nanhai 918691635 :250:100:175

[root@maintain data]# grep -w 'Gege' reg.txt

Zi Gege 1986787350 :250:168:200

[root@maintain data]# grep -o 'Gege' reg.txt

Gege

[复制代码](javascript:void(0);)

4、显示/etc/passwd文件中不以/bin/bash结尾的行

grep -v "/bin/bash$" /etc/passwd

5、找出/etc/passwd文件中两位数或三位数

grep '[1-9]\{1,2\}[1-9]' /etc/passwd

6、找出/etc/rc.d/rc.sysinit或/etc/grub2.cfg文件中，以至少一个空白字符开头，且后面非空白字符的行

grep '^[[:space:]]\+[[:alpha:]]' test.txt

7、找出”netstat -tan”命令的结果中以’LISTEN’后跟0,1或多个空白字符结尾行

netstat -tan | grep 'LISTEN[[:space:]].\*$'

8、创建test.txt文件内容如下：  
He love his lover  
He likes his lover  
She likes her liker  
She loves her liker  
查找单词中出现 l和e之间存在两个字符，且一行中要出现两次查找到的内容。

[复制代码](javascript:void(0);)

[root@maintain data]# cat > test.txt <<EOF

> He love his lover

> He likes his lover

> She likes her liker

> She loves her liker

> EOF

[root@maintain data]# grep '\(l..e\).\*\1' test.txt

He love his lover

She likes her liker

[复制代码](javascript:void(0);)

9、找出/proc/meminfo文件中，所有在大写或小写S开头的行，至少有两种实现方式

egrep "^(S|s)" /proc/meminfo

egrep ^[Ss] /proc/meminfo

10、显示当前系统上root、Python、maintain用户的相关信息

egrep ^\(root\|python\|maintain\) /etc/passwd

11、找出/etc/rc.d/init.d/functions文件中某单词后面跟一个小括号的行

grep "[[:alpha:]]\+(" /etc/rc.d/init.d/functions

egrep '[[:alpha:]]\+|\)|\(' /etc/rc.d/init.d/functions

12、使用echo命令输出一绝对路径，使用egrep取出几名

echo /home/maintain/test.txt | grep -E -o "[^/]+/?$"

13、进一步：取出其路径名；类似于对执行dirname命令结果  
找出ifconfig命令结果中的1-255之间的数值

ifconfig eth0 | egrep -o "[1-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]"

14、添加用户bash,testbash,basher以及nologin（其中shell为/sbin/nologin），而后找出/etc/passwd文件中用户名同shell名行

egrep "(^[[:alnum:]]+\>).\*\1$" /etc/passwd