

grep命令详解

如果你是一个新手，请从头阅读这篇文章，如果你只是忘记了grep命令的一些常用选项，直接查看文章尾部的总结部分即可。

先说说grep命令能做什么？

我们可以使用grep命令在文本中查找指定的字符串，就像你在windows中打开txt文件，使用快捷键“Ctrl+F”在文本中查找某个字符串一样，说白了，可以把grep找工具。

grep是Linux中最常用的“文本处理工具”之一，grep与sed、awk合称为Linux中的三剑客。

grep的全称为：Global search Regular Expression and Print out the line

全称中的“Global search”为全局搜索之意。

全称中的“Regular Expression”表示正则表达式。

所以，从grep的全称中可以了解到，grep是一个可以利用“正则表达式”进行“全局搜索”的工具，grep会在文本文件中按照指定的正则进行全局搜索，并将搜索出的当然，不使用正则表达式时也可以使用grep，但是当grep与正则表达式结合在一起时，威力更强大。

我们先来看一个最简单的使用示例，从最简单的示例开始认识grep。

为了实验方便，我们先准备一个测试文件，文件名为testgrep，文件内容如下。

```
[www.zsythink.net]#cat testgrep
zsy test
zsythink

www.zsythink.net
TEST 123
Zsy's articles
grep Grep
abc
abc123abc
123zsy123
[www.zsythink.net]#
```

假设，现在我们要从testgrep文本文件中搜索包含“test”字符串的行，则可以使用如下命令

```
[www.zsythink.net]#grep "test" testgrep
zsy test
[www.zsythink.net]#
```

上图中的命令表示使用grep命令，在testgrep文件中搜索包含“test”字符串的行，并将包含test字符串的行打印出来。

于是，testgrep文件中的第一行被打印了出来，默认情况下，grep是区分大小写的，所以，文件中包含大写“TEST”的行没有被打印出来。

grep的使用是不是很简单，我们继续聊。

如果我们想要在搜索字符串的时候，不区分大小写，应该怎样做呢？grep很贴心，为我们准备了一个选项，使用“-i”选项，即可在搜索时不区分大小写，示例如下：

```
[www.zsythink.net]#grep -i "test" testgrep
zsy test
TEST 123
[www.zsythink.net]#
```

如上图所示，在不区分大小写的情况下，由于testgrep文本中的第一行与第五行中都包含“test”，所以，这两行都被打印了出来。

由于testgrep文本中的内容不较少、行比较少，所以，我们能数过来，是第一行与第五行包含“test”字符，如果文本中有1000行，我们还想要知道哪行文本包含“test”可以使用“-n”选项，表示显示打印出的行在文本中的行号，示例如下。

```
[www.zsythink.net]#grep -i -n "test" testgrep
1:zsy test
5:TEST 123
[www.zsythink.net]#
```

如上图所示，grep不仅将符合条件的行输出了，同时还显示了行号，证明testgrep文本中的第1行与第5行，在不区分大小写的情况下，都包含字符串“test”。

在centos6中，我们使用grep在文本中搜索出的行虽然会被打印了出来，但是在打印这些行时，被匹配到的关键字没有高亮显示，如果我们想要高亮显示行中的关键字呢？我们可以使用“--color”选项，高亮显示行中的关键字，示例如下

注：“--color”选项是长选项

```
[www.zsythink.net]#grep -i --color "test" testgrep
zsy test
TEST 123
[www.zsythink.net]#
```

使用“--color”与使用“--color=auto”的效果相同，都表示高亮显示关键字，我比较懒，习惯使用“--color”。

在centos7中，系统默认为grep命令配置了别名，所以在centos7中使用grep命令时，不用显式的指定"--color"选项，默认就会高亮显示被匹配到的关键字，在alias命令，即可查看到grep命令的别名，如下

```
[www.zsythink.net]#alias
alias cp='cp -i'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l.='ls -d .* --color=auto'
alias ll='ls -l --color=auto'
alias ls='ls --color=auto'
alias mv='mv -i'
alias rm='rm -i'
alias which='alias | /usr/bin/which --tty-only --read-alias --show-dot --show-tilde'
[www.zsythink.net]#
[www.zsythink.net]#grep -i "test" testgrep
zsy test
TEST 123
[www.zsythink.net]#
```

zsythink.net 朱双印博客

可以看到，centos7中为grep命令配置的别名中，默认包含了--color选项，所以，不用显示指定，也可高亮显示。

所以，我们也可以在centos6中借鉴这种方法。

```
[www.zsythink.net]#alias "grep=grep --color"
[www.zsythink.net]#grep -i "test" testgrep
zsy test
TEST 123
[www.zsythink.net]#
```

zsythink.net 朱双印博客

上图中，testgrep文本中一共有两行包含"test"字符串，如果我们只想知道有多少行包含指定的字符串，而不在乎哪些行包含这些字符串，我们可以使用如下命令，件的总行数。

```
[www.zsythink.net]#grep -i "test" testgrep
zsy test
TEST 123
[www.zsythink.net]#grep -i -c "test" testgrep
2
[www.zsythink.net]#
```

zsythink.net 朱双印博客

没错，使用"-c"选项即可只统计符合条件的总行数，而不会打印出行。

之前的示例中，包含关键词的行都会被打印出来，整行都会被打印出来，如果我们只想看被匹配到的关键字，不想整行都被打印出来，可以吗？必须的，使用"-o"逆出匹配到的关键字，而不打印出整行，示例如下。

```
[www.zsythink.net]#grep -i -o "test" testgrep
test
TEST
[www.zsythink.net]#
```

zsythink.net 朱双印博客

如上图所示，使用"-o"选项，可以只显示被匹配到的关键字，而不是讲整行的内容都输出。

但是需要注意，"-o"选项会把每个匹配到的关键字都单独显示在一行中进行输出，什么意思呢？看如下示例即可明白。

```
[www.zsythink.net]#cat testgrep
zsy test
zsythink

www.zsythink.net
TEST 123
Zsy's articles
grep Grep
abc
abc123abc
123zsy123
[www.zsythink.net]#grep -i "123" testgrep
TEST 123
abc123abc
123zsy123
[www.zsythink.net]#grep -i -o "123" testgrep
123
123
123
123
[www.zsythink.net]#grep -i -n -o "123" testgrep
5:123
9:123
10:123
10:123
[www.zsythink.net]#
```

zsythink.net 朱双印博客

如上图所示，当没有使用"-o"选项时，包含"123"字符串的行都会被打印出来，当同一行中包含多个"123"时，所在行会被打印出来，对应的关键字也会高亮显示，;-o"选项时，每个被匹配到的关键字都会被单独打印在一行中，如上图所示，第三个"123"与第四个"123"都属于第10行的文本，但是它们仍然各自独占一行的输出了

其实，我们在使用grep命令搜索文本时，往往有这种需求：在找到对应的关键字时，同时需要显示关键字附近的信息，什么意思呢？

我们来看一个场景，就能明白，我们新建了一个测试文件：testgrep1，测试文件内容如下

```
[www.zsythink.net]#cat testgrep1
姓名：朱双印
年龄：18
颜值：自认为爆表，其实很Low

姓名：王尼玛
年龄：30
颜值：带着头套，看不到脸

姓名：王尼美
年龄：18
颜值：目测很丑
[www.zsythink.net]#
```

假设，我们想从testgrep1文件中找出"年龄：18"的人，我们该怎么找呢？你可能会尝试使用如下命令。

```
[www.zsythink.net]#grep "年龄：18" testgrep1
年龄：18
年龄：18
[www.zsythink.net]#
```

如上图所示，我们是匹配到了"年龄：18"的行，但是我们并不能从结果中得知年龄为18的人的姓名，因为姓名与年龄并不在一行中，那么我们该怎么办呢？

我们可以使用"-B"选项，显示符合条件的行之前的行，"-B"有before之意，示例如下

```
[www.zsythink.net]#cat testgrep1
姓名：朱双印
年龄：18
颜值：自认为爆表，其实很Low

姓名：王尼玛
年龄：30
颜值：带着头套，看不到脸

姓名：王尼美
年龄：18
颜值：目测很丑
[www.zsythink.net]#grep "年龄：18" testgrep1
年龄：18
年龄：18
[www.zsythink.net]#grep -B1 "年龄：18" testgrep1
姓名：朱双印
年龄：18
--
姓名：王尼美
年龄：18
[www.zsythink.net]#
```

如上图所示，包含字符串"年龄：18"的行被高亮输出了，同时，符合条件的行之前的一行"也被打印了出来，这时，我们就能从结果中得知，朱双印今年18岁，王尼玛岁。

没错，上例中的"-B1"选项表示显示符合条件的行的同时还显示之前的1行，举一反三，"-B5"代表同时显示之前的5行，"-B3"代表同时显示之前的3行，"-B"选项的所字，否则会报错。

与"-B"选项对应的选项是"-A"选项，"-B"有Before之意，"-A"有After之意，聪明如你，一定已经猜到了"-A"的含义，没错，"-A"代表显示符合条件的行的同时，还显示，"-A3"表示同时显示符合条件的行之后的3行，我就不再赘述了。

说了"-A"，说了"-B"，现在说说"-C"，"-C"选项可以理解为"-A与-B"的结合，"-C"选项表示在显示符合条件的行的同时，也会显示其前后的行，如"-C1"，"-C1"表示的行的同时，也打印出之前的一行与之后的一行，"-C"有Context之意（上下文之意），示例如下。

```
[www.zsythink.net]#grep -C1 "年龄：18" testgrep1
姓名：朱双印
年龄：18
颜值：自认为爆表，其实很Low
--
姓名：王尼美
年龄：18
颜值：目测很丑
[www.zsythink.net]#
```

这样我们就能看到"年龄是18岁"的人的所有信息了。

有的时候，我们往往需要进行所谓的“精确匹配”，但是使用之前的方法似乎无法满足我们，示例如下。

```
[www.zsythink.net]#cat testgrep
zsy test
zsythink

www.zsythink.net
TEST 123
Zsy's articles
grep Grep
abc
abc123abc
123zsy123
[www.zsythink.net]#grep "zsy" testgrep
zsy test
zsythink
www.zsythink.net
123zsy123
[www.zsythink.net]#
```

上图中，当我们在文本中搜索“zsy”字符串的时候，“zsy”、“zsythink”、“123zsy123”所在的行都被匹配到了，因为“zsythink”中也包含了“zsy”，所以也被匹配到了。想要“精确匹配”zsy字符串的时候，按照上例中的方法就无法做到了，所谓的精确匹配，就是“zsy”作为一个独立的单词存在，而不是包含于某个字符串中，那么，如求，我们怎么办呢？使用“-w”选项可以实现我们的需求，示例如下。

```
[www.zsythink.net]#cat testgrep
zsy test
zsythink

www.zsythink.net
TEST 123
Zsy's articles
grep Grep
abc
abc123abc
123zsy123
[www.zsythink.net]#grep "zsy" testgrep
zsy test
zsythink
www.zsythink.net
123zsy123
[www.zsythink.net]#grep -w "zsy" testgrep
zsy test
[www.zsythink.net]#
```

如上图所示，只有“zsy”作为一个独立的单词存在的时候，才会被匹配到，“zsy”包含于某个字符串的时候，则不会被匹配到，这就是所谓的精确匹配，“-w”有word：字符串作为一个独立的单词时才会被匹配到。

有的时候，我们需要反向查找，比如，查找“不包含某个字符串”的行，这个时候，我们需要用到“-v”选项，示例如下。

```
[www.zsythink.net]#cat testgrep
zsy test
zsythink

www.zsythink.net
TEST 123
Zsy's articles
grep Grep
abc
abc123abc
123zsy123
[www.zsythink.net]#grep -i -v "zsy" testgrep

TEST 123
grep Grep
abc
abc123abc
[www.zsythink.net]#
```

上例表示查找出文本中不包含“zsy”字符串的行。

某些场景下，我们可能想要同时从多个目标中匹配，什么意思呢？看了示例就秒懂，示例如下。

