

Linux grep 命令

作者: [Leif160519](#)

原文链接: <https://ld246.com/article/1587520287746>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<p></p>

<p>Linux grep 命令用于查找文件里符合条件的字符串。</p>

<p>grep 指令用于查找内容包含指定的范本样式的文件，如果发现某文件的内容符合所指定的范本式，预设 grep 指令会把含有范本样式的那一行显示出来。若不指定任何文件名称，或是所给予的文名为 -，则 grep 指令会从标准输入设备读取数据。</p>

<h2 id="语法">语法</h2>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">grep [-abcEFgHilLnqrsVwxy][-A&lt;显示列数&gt;][-B&lt;显示列数&gt;][-C&lt;显示列数&gt;][-d&lt;进行动作&gt;][-e&lt;范本样式&gt;][-f&lt;范本文件&gt;][--help][范本样式][文件或目录...]</span></span></code></pre>
```

<h2 id="参数">参数</h2>

-a 或 --text : 不要忽略二进制的数。

-A< 显示行数 > 或 --after-context=< 显示行数 > : 除了显示符合范本样式的那一行外，并显示该行之后的内容。

-b 或 --byte-offset : 在显示符合样式的那一行之前，标示出该行第一个字符的编号。

-B< 显示行数 > 或 --before-context=< 显示行数 > : 除了显示符合样式的那一行之外，并显示该行之前的内容。

-c 或 --count : 计算符合样式的列数。

-C< 显示行数 > 或 --context=< 显示行数 > 或-< 显示行数 > : 除了显示符合式的那一行之外，并显示该行之前后的内容。

-d < 动作 > 或 --directories=< 动作 > : 当指定要查找的是目录而非文件时，必须使这项参数，否则 grep 指令将回报信息并停止动作。

-e< 范本样式 > 或 --regexp=< 范本样式 > : 指定字符串做为查找文件内容的样式。

-E 或 --extended-regexp : 将样式为延伸的正则表达式来使用。

-f< 规则文件 > 或 --file=< 规则文件 > : 指定规则文件，其内容含有一个或多个规则式，让 grep 查找符合规则条件的文件内容，格式为每行一个规则样式。

-F 或 --fixed-regexp : 将样式视为固定字符串的列表。

-G 或 --basic-regexp : 将样式视为普通的表示法来使用。

-h 或 --no-filename : 在显示符合样式的那一行之前，不标示该行所属的文件名称。

-H 或 --with-filename : 在显示符合样式的那一行之前，表示该行所属的文件名称。

-i 或 --ignore-case : 忽略字符大小写的差别。

-l 或 --file-with-matches : 列出文件内容符合指定的样式的文件名称。

-L 或 --files-without-match : 列出文件内容不符合指定的样式的文件名称。

-n 或 --line-number : 在显示符合样式的那一行之前，标示出该行的列数编号。

-o 或 --only-matching : 只显示匹配 PATTERN 部分。

-q 或 --quiet 或--silent : 不显示任何信息。

-r 或 --recursive : 此参数的效果和指定"-d recurse"参数相同。

-s 或 --no-messages : 不显示错误信息。

-v 或 --invert-match : 显示不包含匹配文本的所有行。

-V 或 --version : 显示版本信息。

-w 或 --word-regexp : 只显示全字符合的列。

-x --line-regexp : 只显示全列符合的列。

-y : 此参数的效果和指定"-i"参数相同。

<h2 id="实例">实例</h2>

<p>1、在当前目录中，查找后缀有 file 字样的文件中包含 test 字符串的文件，并打印出该字符串的。此时，可以使用如下命令：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">$ grep test *file</span></span></code></pre>
```

<p>结果如下所示：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">$ grep test test* #查找前缀有 "test" 的文件包含 "test" 字符串的文件
</span></span><span class="highlight-line"><span class="highlight-cl">testfile1:This a Lin
x testfile! #列出testfile1 文件中包含test字符的行
</span></span><span class="highlight-line"><span class="highlight-cl">testfile_2:This is a l
inux testfile! #列出testfile_2 文件中包含test字符的行
</span></span><span class="highlight-line"><span class="highlight-cl">testfile_2:Linux tes
#列出testfile_2 文件中包含test字符的行
</span></span></code></pre>
```

2、以递归的方式查找符合条件的文件。例如，查找指定目录/etc/acpi 及其子目录（如果存在子目录的话）下所有文件中包含字符串"update"的文件，并打印出该字符串所在行的内容，使用的命令为

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">$ grep -r update /etc/acpi
</span></span></code></pre>
```

输出结果如下：

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">$ grep -r update /etc/acpi #以递归的方式查找 "etc/acpi"
</span></span><span class="highlight-line"><span class="highlight-cl">#下包含 "update
的文件
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">/etc/acpi/ac.d/85-
nacron.sh:# (Things like the slocate updatedb cause a lot of IO.)
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">Rather than
</span></span><span class="highlight-line"><span class="highlight-cl">/etc/acpi/resume.
/85-anacron.sh:# (Things like the slocate updatedb cause a lot of
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">IO.) Rather than
</span></span><span class="highlight-line"><span class="highlight-cl">/etc/acpi/events/t
inkpad-cmos:action=/usr/sbin/thinkpad-keys--update
```

```
</span></span></code></pre>
```

3、反向查找。前面各个例子是查找并打印出符合条件的行，通过"-v"参数可以打印出不符合条

查找文件名中包含 test 的文件中不包含 test 的行，此时，使用的命令为：

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">grep -v test *test*
</span></span></code></pre>
```

结果如下所示：

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">$ grep -v test* #查找文件名中包含test 的文件中不包含test 的行
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">testfile1:helLinux!
</span></span><span class="highlight-line"><span class="highlight-cl">testfile1:Linis a fre
Unix-type operating system.
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">testfile1:Lin
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">testfile_1:HELLO L
INUX!
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">testfile_1:LINUX IS
A FREE UNIX-TYPE OPERATING SYSTEM.
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">testfile_1:THIS IS A
LINUX TESTFILE!
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">testfile_2:HELLO L
INUX!
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">testfile_2:Linux is a
free unix-type operatng system.
```

```
</span></span></code></pre>
```

场景

<p>查询含有特定文本的文件，并拿到这些文本所在的行</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight cl">grep -n '2019-10-24 00:01:11' *.log</span></span></code></pre>
```

<p>从文件内容查找匹配指定字符串的行: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight cl">$ grep "被查找的字符串" 文件名</span></span></code></pre>
```

<p>从文件内容查找与正则表达式匹配的行: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight cl">$ grep -e "正则表达式" 文件名</span></span></code></pre>
```

<p>查找时不区分大小写: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight cl">$ grep -i "被查找的字符串" 文件名</span></span></code></pre>
```

<p>查找匹配的行数: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight cl">$ grep -c "被查找的字符串" 文件名</span></span></code></pre>
```

<p>从文件内容查找不匹配指定字符串的行: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight cl">$ grep -v "被查找的字符串" 文件名</span></span></code></pre>
```

<p>从根目录开始查找所有扩展名为 .log 的文本文件，并找出包含 "ERROR" 的行: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight cl">$ find / -type f -name "*.log" | xargs grep "ERROR"</span></span></code></pre>
```

<p>参考: </p>

Linux grep 命令

