Grep&正则表达式

主要内容

1、 What grep?

2、 What Regular expressions

3、 Grep Basic usage

4、 Basic Regular expressions

5、 Extend Regular expressions

6、 Actual combat exercise

What grep?

Grep is one among the system administrator’s “Swiss Army knife” set of tools, and is extremely useful to search for strings and patterns in a group of files, or even sub-folders.

What grep?

Grep finds a string in a given file or input, quickly and efficiently. While most everyday uses of the command are simple, there are a variety of more advanced uses that most people don’t know about — including regular expressions and more, which can become quite complicated.

汪汪标题

What grep?

There are two ways to provide input to Grep, each with its own particular uses. First, Grep can be used to search a given file or files on a system (including a recursive search through sub-folders). Grep also accepts inputs (usually via a pipe) from another command or series of commands.

汪汪标题

What Regular expressions?

A regular expression, often shortened to “regex” or “regexp”, is a way of specifying a pattern (a particular set of characters or words) in text that can be applied to variable inputs to find all occurrences that match the pattern. Regexes enhance the ability to meaningfully process text content, especially when combined with other commands.

Grep Basic usage

grep [options] [regexp] [filename] command

[options]:

- -i
- -w
- -r
- -v
- -H
- -L
- -A num
- -B num
- -C num
- -q

Basic Regular expressions

- 字符匹配
- 字数匹配
- 位置锚定
- 分组引用

Basic Regular expressions

字符匹配：

.：匹配任意单个字符
[]：匹配指定范围内的任意单个字符
[^]：匹配指定范围外的任意单个字符

POSIX CLASS

Basic Regular expressions

匹配次数（贪婪模式）：

*: 匹配其前面的字符任意次

a, b, ab, aab, acb, adb, amnb

a*b， a?b

a.*b

.*: 任意长度的任意字符

\?: 匹配其前面的字符1次或0次

\{m,n\}:匹配其前面的字符至少m次，至多n次

\{1,\}

\{0,3\}

Basic Regular expressions

位置锚定：

^：锚定行首，此字符后面的任意内容必须出现在行首
$：锚定行尾，此字符前面的任意内容必须出现在行尾
^$：空白行
\<或\b：锚定词首，其后面的任意字符必须作为单词首部出现
\>或\b：锚定词尾，其前面的任意字符必须作为单词的尾部出现

Basic Regular expressions

分组引用：

\(\)

\(ab\)*

后向引用

\1：引用第一个左括号以及与之对应的右括号所包括的所有内容

\2：

\3：

like-->liker
love-->lover
like-->Like
love-->Love

Extend Regular expressions

扩展正则表达式
字符匹配：

.
[]
[^]

次数匹配：
*:
?:
+: 匹配其前面的字符至少1次
{m,n}

位置锚定：
^
$
\<
\>

分组：
(): 分组
\1, \2, \3, ...
或者
|: or
C|cat: Cat或cat, C或cat

Actual combat exercise

1.邮箱格式匹配
2.IP地址匹配
3.用户名匹配
4.手机号码匹配
5.Any things

项目实战(根据时间充裕情况，可选讲解):
1.判断网络是否通 ping
2.获取网卡地址 ifconfig
3.获取内存使用率 free
4.获取CPU的负载情况 w
5.获取CPU的空闲率 vmstat

谢谢！

附录A

POSIX-Style Character Classes

BBEdit now provides support for POSIX-style character classes. These classes are used in the form [:CLASS:] , and are only available inside regular character classes (in other words, inside another set of square brackets).

Class	Meaning
alnum	letters and digits
alpha	letters
ascii	character codes 0 - 127
cntrl	control characters
digit	decimal digits (same as \d)
graph	printing characters, excluding space
lower	lower case letters
print	printing characters, including space
punct	punctuation characters
space	white space (same as \s)
upper	upper case letters
word	"word" characters (same as \w)
xdigit	hexadecimal digits

附录B

```
tar [-] A --catenate --concatenate | c --create | d --diff --compare | --delete | r --append | t --list | --test-label | u
    --update | x --extract --get [options] [pathname ...]
```

```
docker [OPTIONS] COMMAND [arg...]
```