

8 正则表达式

主讲：杨东平
中国矿大计算机学院

正则表达式是什么？

- 正则表达式是对字符串操作的一种逻辑公式，即用事先定义好的一些特定字符、及这些特定字符的组合，组成一个“规则字符串”，这个“规则字符串”用来表达对字符串的一种过滤逻辑
- 正则表达式的目的：
 - 给定的字符串是否符合正则表达式的过滤逻辑(称作“匹配”)
 - 正则表达式还用于字符串的模式分割、查找和替换
- 正则表达式的特点：
 - 灵活性、逻辑性和功能性非常的强
 - 可以迅速地用极简单的方式达到字符串的复杂控制
 - 对于刚接触的人来说，比较晦涩难懂

网络安全与网络工程系杨东平 jsxhbc@163.com

Linux操作系统

2018年9月14日1时25分

2

正则表达式与通配符

- 正则表达式用来在文件中匹配符合条件的字符串，正则包含匹配。grep、awk、sed等命令可以支持正则表达式
- 通配符用来匹配符合条件的文件名，通配符是完全匹配。ls、find、cp这些命令不支持正则表达式，所以只能使用shell 自己的通配符来进行匹配了

网络安全与网络工程系杨东平 jsxhbc@163.com

Linux操作系统

2018年9月14日1时25分

3

正则表达式分类

- 基本的正则表达式(Basic Regular Expression，又称为 Basic RegEx，简称 BRES)
- 扩展的正则表达式(Extended Regular Expression，又称为 Extended RegEx，简称 EREs)
- Perl 的正则表达式(Perl Regular Expression，又称为 Perl RegEx，简称 PREs)

网络安全与网络工程系杨东平 jsxhbc@163.com

Linux操作系统

2018年9月14日1时25分

4

正则表达式的组成



网络安全与网络工程系杨东平 jsxhbc@163.com

Linux操作系统

2018年9月14日1时25分

5

字符类

- 字符类(Character Class)在模式中表示一个字符，但是取值范围是一类字符中的任意一个

字符	含义	举例
.	匹配任意一个字符	abc 可以匹配 abcd,abc9
[]	匹配括号中的任意一个字符	[abc] d 可以匹配 ad、bd 或 cd
-	在 [] 括号中使用，表示字符范围	[0-9a-fA-F] 可以匹配一位十六进制数字
^	位于 [] 括号内的开始，匹配除括号中的字符外的任意一个字符	[^xy] 匹配除 x、y 之外的任意一个字符
[:xxx:]	grep 工具预定义的一些命名字符	[:digit:] 匹配一个数字 [:alpha:] 匹配一个字符

网络安全与网络工程系杨东平 jsxhbc@163.com

Linux操作系统

2018年9月14日1时25分

6

字符类(续)

►例(视频: 14 正则表达式: 字符类):

```
root@localhost ~]# echo "abc9" | grep --color 'abc.'
abc9
root@localhost ~]# echo "abcdef" | grep --color 'abc.'
abcdef
root@localhost ~]# echo "ad" | grep --color '[abcd]'
ad
root@localhost ~]# echo "bd" | grep --color '[abcd]'
bd
root@localhost ~]# echo "cd" | grep --color '[abcd]'
cd
root@localhost ~]# echo "hello world" | grep --color '[^xy]'
hello world
root@localhost ~]# echo "hello xy" | grep --color '[^xy]'
hello xy
root@localhost ~]# echo "xy hello" | grep --color '[^xy]'
xy hello
root@localhost ~]# echo "xy567 hello" | grep --color '[[digit]]'
xy567 hello
root@localhost ~]# echo "xy567 hello" | grep --color '[[alpha]]'
xy567 hello
root@localhost ~]# echo "xy567 hello" | grep --color '[[0-9]]'
xy567 hello
root@localhost ~]# echo "xy567 hello" | grep --color '[0-9]'
xy567 hello
```

网络安全与网络工程系杨永平 jsxhbc@163.com Linux操作系统 2018年9月14日1时25分 7

数量限定符

字符	含义	举例
?	紧跟在它前面的单元应匹配零次或一次	[0-9]? 匹配 0.0、2.3、.5 等(. 在正则中是一个特殊字符, 所以用 \ 转义)
+	紧跟在它前面的单元应匹配一次或多次	[a-zA-Z0-9_-]+@ [a-zA-Z0-9_-]+.[a-zA-Z0-9_-]+ 匹配email地址
*	紧跟在它前面的单元应匹配零次或多次	[0-9]* 匹配至少一位数字
{N}	紧跟在它前面单元应精确匹配 N 次	[1-9]{0-9}{2} 匹配 100-999 的整数
{N,}	紧跟在它前面的单元应匹配至少 N 次	[1-9]{0-9}{2,} 匹配三位及以上的整数
{,M}	紧跟在它前面的单元应匹配最多 M 次	[0-9]{,1} 相当于 [0-9]?, 最多匹配一次数字
{N,M}	紧跟在它前面的单元应匹配至少 N 次, 最多 M 次	[0-9]{1,3} 表示 0-9 数字至少匹配一次, 最多匹配3次

网络安全与网络工程系杨永平 jsxhbc@163.com Linux操作系统 2018年9月14日1时25分 8

数量限定符(续)

►例(视频: 15 正则表达式: 数量限定符):

```
root@localhost ~]# echo "hello world" | grep -E --color 'he?'
hello world
root@localhost ~]# echo "hellooooo world" | grep -E --color 'hello(3)'
hellooooo world
root@localhost ~]# echo "hellooooo world" | grep -E --color 'hello(3,)'
hellooooo world
root@localhost ~]# echo "hellooooo world" | grep -E --color 'hello(3,5)'
hellooooo world
root@localhost ~]# echo "hellooooo world" | grep -E --color 'hello(3,4)'
hellooooo world
```

网络安全与网络工程系杨永平 jsxhbc@163.com Linux操作系统 2018年9月14日1时25分 9

位置限定符

►位置限定符(Anchor): 描述各种字符类以及普通字符之间的位置关系

字符	含义	举例
^	匹配行首的位置	^content 匹配行首为 content 的行
\$	匹配行末的位置	;\$ 匹配行末为 “.” 的行; ^\$ 匹配空行
<	匹配单词开头的位置	<th 匹配 "...this...", 但不匹配 tenth 等
>	匹配单词词尾的位置	p> 匹配 leap, 但不匹配 parent 等
\b	匹配单词的开头或结尾的位置	\batb 匹配 "...at ...", 但不匹配 cat、batch 等
\B	匹配非单位的开头或结尾的位置	\BatB 匹配 battery, 但不匹配 "...attend."、"hat..."

网络安全与网络工程系杨永平 jsxhbc@163.com Linux操作系统 2018年9月14日1时25分 10

位置限定符(续)

►例(视频: 16 正则表达式: 位置限定符):

```
root@localhost ~]# echo "hello world" | grep -E --color '^hello'
hello world
root@localhost ~]# echo "hello world" | grep -E --color '^ld$'
root@localhost ~]# echo "hello world" | grep -E --color 'ld$'
hello world
root@localhost ~]# echo "hello atexitat world" | grep -E --color '\<at'
hello atexitat world
root@localhost ~]# echo "hello atexitat world" | grep -E --color 'at\>'
hello atexitat world
root@localhost ~]# echo "hello at atexitat world" | grep -E --color '\<at\>'
hello at atexitat world
root@localhost ~]# echo "hello at atexitat world" | grep -E --color '\<at\b'
hello at atexitat world
root@localhost ~]# echo "hello at atexitat world" | grep -E --color '\<at\b'
hello at atexitat world
root@localhost ~]# echo "hello at atexitat world" | grep -E --color '\<at\b'
hello at atexitat world
root@localhost ~]# echo "hello at atexitat world" | grep -E --color '\<at\b'
hello at atexitat world
root@localhost ~]# echo "hello at atexitat world" | grep -E --color '\<at\b'
hello at atexitat world
```

网络安全与网络工程系杨永平 jsxhbc@163.com Linux操作系统 2018年9月14日1时25分 11

其它特殊字符

字符	含义	举例
\	转义符, 普通字符转为特殊字符, 特殊字符转为普通字符	普通字符 < 写成 \< 表示单词开头的位置; 特殊字符 \ 写成 \ 就当作普通字符来匹配
	将正则表达式的一部分括起来组成一个单元, 可以对整个单元使用数量限定符	([0-9]{1,3}){3}[0-9]{1,3} 匹配 IP 地址
	连接两个子表达式, 表示或的关系	n(o e)ither 匹配 no 或 neither

►例(视频: 17 正则表达式: 特殊字符):

```
root@localhost ~]# echo "12352@qq.com" | grep -E --color '^([0-9]{5,12})@qq\.'
12352@qq.com
root@localhost ~]# echo "192.168.185.28" | grep -E --color '^([0-9]{1,3}\.){3}[0-9]{1,3}$'
192.168.185.28
root@localhost ~]# echo "hello world" | grep -E --color '^H'h'
hello world
root@localhost ~]# echo "Hello world" | grep -E --color '^H'h'
Hello world
```

网络安全与网络工程系杨永平 jsxhbc@163.com Linux操作系统 2018年9月14日1时25分 12