

第5天-文件查找/打包压缩

一、文件查找

1、which/whereis/whatis

which 只能查询命令

```
[root@qfedu.com ~]#which rpm
```

whereis

可以查询命令和配置文件的位置

```
[root@qfedu.com ~]#whereis rpm
```

```
[root@qfedu.com ~]#whereis passwd
```

whatis

```
[root@qfedu.com ~]#whatis rpm 和下面命令一样的效果，查询rpm命令都在哪章man有解释
```

```
[root@qfedu.com ~]#man -f rpm
```

2、locate

维护着一个查询数据库

```
[root@qfedu.com ~]#vim /etc/updatedb.conf
```

1) 文件系统类型

2) 目录

如果被更改之后，需要更新数据库

```
[root@qfedu.com ~]#updatedb 手动更新数据库
```

```
[root@qfedu.com ~]#locate 被查找的关键字
```

```
[root@qfedu.com ~]#locate *.txt
```

*是通配符

3、find

-exec -atime -mtime -ctime -size -name

```
[root@qfedu.com ~]#find 路径 条件 跟条件相关的操作符 [-exec|-ok 动作]
```

路径

默认不写路径时查找的是当前路径

例：

/etc

./

/

/var/ftp

条件

名称 大小 时间 文件类型 用户 组 权限 ...

-name 文件名称 按名称查找

```
# find / -name a.txt
# find / -name a.t??
# find / -name a.tx?
# find / -name 'a.txt'
# find / -iname 'a.txt'
```

-iname 不区分大小写

?表示单个字符

*表示所有字符

一般情况下{}不能用

{1..100}

{abc,abd,efg}

通配符:

```
* ? [ ] {}
* 表示所有字符
? 表示任意单个字符
[ ] 表示其中任意一个单个字符
例:
[abc]
[a-z]
[a-Z]
[a-zA-Z]
[!a-z]      !取反
[0-9]
```

a到z的匹配顺序是aAbBcC...

```
[root@server python]# ls
```

```
a.txt  E.txt  j.txt  N.txt  s.txt  W.txt
A.txt  f.txt  J.txt  o.txt  S.txt  x.txt
b.txt  F.txt  k.txt  O.txt  t.txt  X.txt
B.txt  g.txt  K.txt  p.txt  T.txt  y.txt
c.txt  G.txt  l.txt  P.txt  u.txt  Y.txt
C.txt  h.txt  L.txt  q.txt  U.txt  z.txt
d.txt  H.txt  m.txt  Q.txt  v.txt  Z.txt
D.txt  i.txt  M.txt  r.txt  V.txt
e.txt  I.txt  n.txt  R.txt  w.txt
```

```
{hello,hi,king,xiaoxuan}
```

```
[root@qfedu.com ~]# touch {python,wing,haha}.txt
```

```
{a..z}
```

```
{1..100}  
/dev/vdc{1,2,3}
```

按大小查找

-size

```
[root@qfedu.com ~]#find / -size 50M
```

```
[root@qfedu.com ~]#find / -size +50M
```

```
[root@qfedu.com ~]#find / -size -50M
```

查找大于10M小于20M

```
[root@qfedu.com ~]#find / -size +10M -a -size -20M
```

-a可以换成**-and**

```
[root@qfedu.com ~]#find / -size -10M -o -size +20M
```

-o可以换成**-or**

```
[root@qfedu.com ~]# find ./ ! -size -10M
```

```
[root@qfedu.com ~]# find / -size -50M -a -name "*wing*"
```

```
[root@qfedu.com ~]# find / ! \( -size -50M -a -name "*wing*" \)
```

!取反

```
\( \)
```

\ 转义字符 把有意义的变的没意义 把没意义的变的有意义

附加:用dd命令做测试数据

```
[root@qfedu.com ~]#dd if=/dev/zero of=/tmp/aa.txt bs=5M count=2
```

按文件类型查找

-type

f

d

b

c

l

s

p

```
[root@qfedu.com ~]# find / -type c -exec ls -l {} \;
```

```
[root@qfedu.com ~]# find /tmp/ -name aa.txt -exec rm -i {} \;
```

```
[root@qfedu.com ~]# find /tmp/ -name aa.txt -ok rm {} \;
```

```
< rm ... /tmp/aa.txt > ? y
```

-exec 对之前查找出来的文件做进一步操作

-ok 和**-exec**一样,只不过多了提示

按权限查找:

-perm

```
[root@qfedu.com ~]# find ./ -perm 644 -ls
```

```
./dd.txt
```

按用户和组查找

`-user`

`-group`

```
[root@qfedu.com ~]# find ./ -user wing
./bb.txt
[root@qfedu.com ~]# find ./ -group user3
./cc.txt
```

按时间

`-atime` access时间

`-mtime` modify时间

`-ctime` change时间

`-amin`

`-mmin`

`-cmin`

time表示单位是天

min 表示分钟

```
[root@qfedu.com ~]# stat 文件
```

查找两分钟内访问过的文件

```
[root@qfedu.com ~]# find /tmp -amin -2
/tmp/a.txt
```

查找两分钟前访问过的文件

```
[root@qfedu.com ~]# find /tmp -amin +2
```

查找一个文件的硬链接:

```
[root@qfedu.com ~]# ln a.txt heihei
[root@qfedu.com ~]# ll -i
439360 -rw-r--r-- 2 root root 12 Nov 29 22:22 a.txt
439360 -rw-r--r-- 2 root root 12 Nov 29 22:22 heihei
[root@qfedu.com ~]# find . -samefile a.txt
./a.txt
./heihei
```

指定查找的目录深度:

`-maxdepth` levels

`-mindepth` levels

```
[root@qfedu.com ~]# find / -maxdepth 3 -a -name "ifcfg-eth0"
```

按正则表达式查找:

```
[root@qfedu.com ~]# find /etc -regex '.*ifcfg-ens[0-9][0-9]'
```

`-exec` `-ok`

```
[root@qfedu.com ~]# find . -name wing.txt -exec cp {} /root/Desktop/ \;
```

防止被查找到的文件过多, 导致内存溢出错误

```
[root@qfedu.com ~]# find . -name wing.txt | xargs -i cp {} /root/Desktop
```

4、xargs

防止被查找到的文件过多，导致内存溢出错误

```
[root@qfedu.com ~]# find . -name wing.txt | xargs -i cp {} /root/Desktop
```

二、文件查找实战

使用各种文件查找命令对系统内的文件进行各种常用条件下的查找操作

三、文件打包及压缩

归档：

```
[root@qfedu.com ~]# tar cvf file.tar 被打包的文件 ...
```

```
[root@qfedu.com ~]# tar cvf abc.tar a b c d e
```

c create

v verbose

f file

解包：

```
[root@qfedu.com ~]# tar xvf 打包文件 [-C /root/Desktop]
```

x extract

-C 指定解包路径

gzip bzip2

压缩：

```
[root@qfedu.com ~]# gzip 源文件 ...
```

```
[root@qfedu.com ~]# bzip2 源文件 ...
```

解压：

```
[root@qfedu.com ~]# gunzip 压缩文件
```

```
[root@qfedu.com ~]# bunzip2 压缩文件
```

```
[root@qfedu.com ~]# gzip -d 压缩文件
```

```
[root@qfedu.com ~]# bzip2 -d 压缩文件
```

打包压缩一起：

```
[root@qfedu.com ~]# tar cvzf file.tar.gz 源文件 ...
```

```
[root@qfedu.com ~]# tar cvjf file.tar.bz2 源文件 ...
```

解压解包一起：

```
[root@qfedu.com ~]# tar xvzf 压缩文件 [-C 解压路径]
```

```
[root@qfedu.com ~]# tar xvjf 压缩文件 [-C 解压路径]
```

查看压缩文件里面的子文件

-t

```
[root@qfedu.com ~]# tar -tf 压缩文件
```

```
[root@qfedu.com ~]# tar -cJf etc3.tar.xz /etc
```

-J 调用xz

四、归档压缩实战

- 1、使用常用打包压缩命令对指定文件和目录进行打包压缩操作
- 2、将海量小文件快速复制至远程主机

特例：

数据库数据目录特别大 几十个G 如果rm删除会把io用满，用truncate 一点一点的删:truncate -s 10M /path/*

干锋云计算学院