

Linux文件查找和打包压缩详解

一.文件查找工具locate和find

1.locate

- locate依靠查询系统上预建的文件索引数据库来查询某个文件，按名字来查找，速度很快

数据库：/var/lib/mlocate/mlocate.db

- 其工作依赖于事先构建的索引
- 索引的构建是在系统较为空闲时自动进行(周期性任务)，管理员手动更新数据库(updatedb 命令)
- 索引构建过程需要遍历整个根文件系统，极消耗资源 工作特点: •查找速度快 •模糊查找 •非实时查找 •搜索的是文件的全路径，不仅仅是文件名 •可能只搜索用户具备读取和执行权限的目录
- 用法

```
locate [OPTION]... PATTERN...
```

- i 不区分大小写的搜索
- n N 只列举前N个匹配项目
- r 使用基本正则表达式

示例

搜索名称或路径中带有“conf”的文件

```
locate conf
```

使用Regex来搜索以“.conf”结尾的文件

```
locate -r '\.conf$'
```

2.find

- 实时查找工具，通过遍历指定路径完成文件查找

工作特点:

- 查找速度略慢
- 精确查找
- 实时查找
- 可能只搜索用户具备读取和执行权限的目录

- 语法

```
find [OPTION]... [查找路径] [查找条件] [处理动作]
```

2.1查找路径：指定具体目标路径；默认为当前目录

2.2查找条件：指定的查找标准，可以文件名、大小、类型、权限等标准进行；默认为找出指定路径下的所有文件

指定搜索层级

```
-maxdepth level 最大搜索目录深度,指定目录下的文件为第1级  
-mindepth level 最小搜索目录深度
```

- 先处理目录内的文件，再处理指定目录

```
-depth
```

根据文件名和inode查找

```
-name "文件名称": 支持使用glob  
    *, ?, [], [^]  
-iname "文件名称": 不区分字母大小写  
-inum n 按inode号查找  
-samefile name 相同inode号的文件  
-links n 链接数为n的文件  
-regex "PATTERN": 以PATTERN匹配整个文件路径，而非文件名称
```

根据属主、属组查找

```
user USERNAME: 查找属主为指定用户(UID)的文件  
group GRPNAME: 查找属组为指定组(GID)的文件  
uid UserID: 查找属主为指定的UID号的文件  
gid GroupID: 查找属组为指定的GID号的文件  
nouser: 查找没有属主的文件  
nogroup: 查找没有属组的文件
```

根据文件类型查找

```
-type TYPE  
    •f: 普通文件  
    •d: 目录文件  
    •l: 符号链接文件  
    •s: 套接字文件  
    •b: 块设备文件  
    •c: 字符设备文件  
    •p: 管道文件  
空文件或目录
```

-empty

示例: `find /app -type d -empty`

组合条件

与: -a

或: -o

非: -not !

德·摩根定律:

(非 A) 或 (非 B) = 非(A 且 B)

(非 A) 且 (非 B) = 非(A 或 B)

示例:

`!A -a !B = !(A -o B)`

`!A -o !B = !(A -a B)`

find 示例

```
find -name snow.png
find -iname snow.png
find / -name "*.txt"
find /var -name "*log*"
find -user joe -group joe
find -user joe -not -group joe
find -user joe -o -user jane
find -not \( -user joe -o -user jane \)
find / -user joe -o -uid 500
```

找出/tmp目录下, 属主不是root, 且文件名不以f开头的文件

```
find /tmp \( -not -user root -a -not -name 'f*' \) -ls
```

```
find /tmp -not \( -user root -o -name 'f*' \) -ls
```

排除目录

查找/etc/下, 除/etc/sane.d目录的其它所有.conf后缀的文件

```
find /etc -path '/etc/sane.d' -a -prune -o -name "*.conf"
```

查找/etc/下, 除/etc/sane.d和/etc/fonts两个目录的所有.conf后缀的文件

```
find /etc \( -path "/etc/sane.d" -o -path "/etc/fonts" \) -a -prune -o -name "*.conf"
```

根据文件大小来查找

-size [+|-]#UNIT

常用单位: k, M, G, c (byte)

#UNIT: (#-1, #]

如: 6k 表示(5k,6k]

-#UNIT: [0, #-1]

如: -6k 表示[0,5k]

```

+#UNIT: (#,∞)
    如: +6k 表示(6k,∞)

```

根据时间戳

```

以“天”为单位
    -atime [+|-]#,
        #: [#,#+1)
        +#: [#+1,∞]
        -#: [0,#)
    -mtime
    -ctime
以“分钟”为单位
    -amin
    -mmin
    -cmin

```

根据权限查找

```

-perm [ / | - ] MODE
    MODE: 精确权限匹配
    /MODE: 任何一类(u,g,o)对象的权限中只要能一位匹配即可, 或关系, +号从centos7开始
淘汰
    -MODE: 每一类对象都必须同时拥有指定权限, 与关系
        0: 表示不关注
• find -perm 755 会匹配权限模式恰好是755的文件
• 只要当任意人有写权限时, find -perm +222就会匹配
• 只有当每个人都有写权限时, find -perm -222才会匹配
• 只有当其它人(other)有写权限时, find -perm -002才会匹配

```

2.3处理动作：对符合条件的文件做操作，默认输出至屏幕

```

-print: 默认的处理动作, 显示至屏幕
-ls: 类似于对查找到的文件执行“ls -l”命令
-delete: 删除查找到的文件
-fls file: 查找到的所有文件的长格式信息保存至指定文件中
-ok COMMAND {} \; 对查找到的每个文件执行由COMMAND指定的命令, 对于每个文件执行命令之前,
都会交互式要求用户确认
-exec COMMAND {} \; 对查找到的每个文件执行由COMMAND指定的命令
    {}: 用于引用查找到的文件名称自身
find传递查找到的文件至后面指定的命令时, 查找到的所有符合条件的文件一次性传递给后面的命令

```

2.4由于很多命令不支持管道|来传递参数，xargs用于产生某个命令的参数，xargs

- `xargs`可以读入`stdin` 的数据，并且以空格符或回车符将`stdin`的数据分隔成为参数;许多命令不能接受过多参数，命令执行可能会失败，`xargs`可以解决
- - 注意：文件名或者是其他意义的名词内含有空格符的情况使用`xargs`需要指明其它的分隔符

`find`和`xargs`的组合: `find | xargs COMMAND`

示例:

```
ls | xargs rm
删除当前目录下的大量文件
find /sbin/ -perm +700 | ls -l      这个命令是错误的
find /bin/ -perm /7000 | xargs ls -Sl  查找有特殊权限的文件，并排序
find /bin/ -perm -7000 | xargs ls -Sl  此命令和上面有何区别？
find -type f -name "*.txt" -print0 | xargs -0 rm 以字符nul分隔
```

find示例

```
备份配置文件，添加.orig这个扩展名
find -name "*.conf" -exec cp {} {}.orig \;
提示删除存在时间超过 3 天以上的joe的临时文件
find /tmp -ctime +3 -user joe -ok rm {} \;
在主目录中寻找可被其它用户写入的文件
find ~ -perm -002 -exec chmod o-w {} \;
查找/data下的权限为644，后缀为sh的普通文件，增加执行权限
find /data -type f -perm 644 -name "*.sh" -exec chmod 755 {} \;
查看/home的目录
find /home -type d -ls
```

练习

- 1、查找/var目录下属主为root，且属组为mail的所有文件
`[root@centos7 ~]# find /var/ -user root -group mail/var/spool/mail /var/spool/mail/root`
- 2、查找/var目录下不属于root、lp、gdm的所有文件
`find /var/ \(-not -user root -a -not -user lp -a -not -user gdm \) -ls`
- 3、查找/var目录下最近一周内其内容修改过，同时属主不为root，也不是postfix的文件
`find /var/ -mtime -7 -o \(-not -user root -a -not -user postfix \) -ls`
- 4、查找当前系统上没有属主或属组，且最近一个周内曾被访问过的文件
`find / -atime -7 -nouser -o -nogroup`
- 5、查找/etc目录下大于1M且类型为普通文件的所有文件
`find /etc/ -size +1M -a -type f`
- 6、查找/etc目录下所有用户都没有写权限的文件
`find /etc/ -not -perm /222`

- 7、查找/etc目录下至少有一类用户没有执行权限的文件

```
find /etc/ -not -perm -111
```

- 8、查找/etc/init.d目录下，所有用户都有执行权限， 且其它用户有写权限的文件

```
find /etc/init.d -perm -113
```

二.文本打包工具tar和cpio

1.tar

- tar (Tape ARchive, 磁带归档的缩写)

tar [OPTION]...

- (1) 创建归档，保留权限

```
tar -cpvf /PATH/FILE.tar FILE...
```

- (2) 追加文件至归档： 注：不支持对压缩文件追加

```
tar -r -f /PATH/FILE.tar FILE...
```

- (3) 查看归档文件中的文件列表

```
tar -t -f /PATH/FILE.tar
```

- (4) 展开归档

```
tar -x -f /PATH/FILE.tar
```

```
tar -x -f /PATH/FILE.tar -C /PATH/
```

- (5) 结合压缩工具实现： 归档并压缩

```
-j: bzip2, -z: gzip, -J: xz
```

--exclude 排除文件

```
tar zcvf /root/a3.tgz --exclude=/app/host1 --exclude=/app/host2 /app
```

-T 选项指定输入文件 -X 选项指定包含要排除的文件列表

```
tar zcvf mybackup.tgz -T /root/includefilelist -X /root/excludefilelist
```

split: 分割一个文件为多个文件

分割大的 tar 文件为多份小文件

```
split -b Size -d tar-file-name prefix-name
```

```
split -b 1M -d mybackup.tgz mybackup-parts
```

```
split -b 1M mybackup.tgz mybackup-parts
```

合并:

```
cat mybackup-parts* > mybackup.tar.gz
```

2.cpio

- 功能：打包文件或解包
- - cpio命令是通过重定向的方式将文件进行打包备份，还原恢复的工具，它可以解压以".cpio"或者".tar"结尾的文件
- cpio [选项] > 文件名或者设备名
- cpio [选项] < 文件名或者设备名

选项

-o output模式，打包，将标准输入传入的文件名打包后发送到标准输出

```
-i input模式，解包，对标准输入传入的打包文件名解包到当前目录
-t 预览，查看标准输入传入的打包文件中包含的文件列表
-O filename 输出到指定的归档文件名
-A 向已存在的归档文件中追加文件
-I filename 对指定的归档文件名解压
-F filename 使用指定的文件名替代标准输入或输出
-d 解包生成目录，在cpio还原时，自动的建立目录
-v 显示打包过程中的文件名称
```

- 示例

将etc目录备份:

```
find ./etc -print | cpio -ov > bak.cpio
```

将/data内容追加bak.cpio

```
find /data | cpio -oA -F bak.cpio
```

内容预览

```
cpio -tv < etc.cpio
```

解包文件

```
cpio -idv < etc.cpio
```

三.文件压缩和解压缩工具

1.compress,uncompress

```
compress [-dfvcVr] [-b maxbits] [file ...]
-d 解压缩，相当于uncompress
-c 结果输出至标准输出,不删除原文件
-v 显示详情
uncompress file.Z 解压缩
zcat file.Z 不显式解压缩的前提下查看文本文件内容
示例: zcat file.Z >file
```

2.gzip,gunzip,zcat

```
gzip [OPTION]... FILE ...
-d 解压缩，相当于gunzip
-c 结果输出至标准输出，保留原文件不改变
-# 指定压缩比，#取值为1-9，值越大压缩比越大
gunzip file.gz 解压缩
zcat file.gz 不显式解压缩的前提下查看文本文件内容
示例:
gzip -c messages >messages.gz
gzip -c -d messages.gz > messages
zcat messages.gz > messages
cat messages | gzip > m.gz
```

3.bz2,bunzip2,bzcat

```
bzip2 [OPTION]... FILE ...  
-k keep, 保留原文件  
-d 解压缩  
-# 1-9, 压缩比, 默认为9  
bunzip2 file.bz2 解压缩  
bzcat file.bz2 不显式解压缩的前提下查看文本文件内容
```

4.zip,unzip,xzcat

```
xz [OPTION]... FILE ...  
-k keep, 保留原文件  
-d 解压缩  
-# 压缩比, 取值1-9, 默认为6  
unxz file.xz 解压缩  
xzcat file.xz 不显式解压缩的前提下查看文本文件内容
```