

## 【1-10】打包、压缩与软件安装

笔记本： 备课\_linux

创建时间： 2022/4/17 8:53

更新时间： 2022/4/28 17:55

作者： 兰鸣人花道

位置： 30°46'37 N 103°56'14 E

### 【前言】压缩与打包的概念？

- 压缩
  - 把文件经过一些算法的处理编程体积上更小的文件
  - 用于传输
- 打包
  - 把文件和目录的结构内容拷贝到一个文件里
  - 用于备份、归档
- WinRAR?
  - 一般我们都是归档的同时也进行压缩

## 1、压缩

- gzip
  - 压缩一个或者多个文件
  - 但是压缩出来的是单独的压缩文件
  - 语法
    - gzip 文件名
      - 压缩完成后，原来的源文件会被替换掉，成 \*.gz
    - gzip -d \*.gz
  - 不能对整个目录压缩，但是可以通过加-r参数来分别压缩目录下的文件
- zcat
  - zcat xxx.gz，可以直接查看压缩文件的内容
- bzip2
  - bzip2的压缩比会比gzip更好，语法类似
  - bzip2 xxx
  - bzip2 -d xxx.bz2
  - yum install -y bzip2
- zip
  - 用得少，了解即可
  - 先安装
    - yum install -y zip
    - yum install -y unzip
  - 语法：
    - zip xxx.zip file1 file2 ... 将file1、file2。。。压缩到xxx.zip中
    - zip -r root.zip /root 将root目录压缩
    - zip xxx.zip ./\* 将当前目录下的内容全部压缩到xxx.zip中
    - zipinfo 可以查看zip包
- unzip
  - unzip xxx.zip 解压

```
# gzip
[root@lanhai gzip_test]# gzip file1
[root@lanhai gzip_test]# ls file1*
file1.gz
[root@lanhai gzip_test]# gzip -d file1.gz
[root@lanhai gzip_test]# ls file1*
file1
[root@lanhai gzip_test]#
```

```
# gzip不能对目录本身进行压缩
[root@localhost ~]# gzip /test
gzip: /test is a directory -- ignored

# 通过-r可以对目录操作，会把目录下的所有文件，分别进行压缩，压缩成对应的文件
[root@localhost ~]# gzip -r /test
[root@localhost ~]# cd /test
[root@localhost test]# ls -l
total 28
-rw----r--. 1 tom root    38 Apr 28 16:07 file1.gz
-rw-r--r--. 1 tom root    26 Apr 28 16:02 file2.gz
-rw-rw-r--. 1 tom root    34 Apr 28 16:13 file_by_jerry.gz
-rw-r--r--. 1 tom root    32 Apr 28 16:36 file_by_tom.gz
-rw-r--r--. 1 root root  5112 Apr 28 17:04 functions.gz
drwxrwxr-x. 3 tom group1  56 Apr 28 17:10 jerry1
-rw-rw-r--. 1 tom root    31 Apr 28 16:46 jerry_file.gz
[root@localhost test]#

# 可以通过 gzip -d *.gz 全部解压，为压缩文件对应出来的文件
[root@localhost test]# ll
total 28
-rw----r--. 1 tom root    38 Apr 28 16:07 file1.gz
-rw-r--r--. 1 tom root    26 Apr 28 16:02 file2.gz
-rw-rw-r--. 1 tom root    34 Apr 28 16:13 file_by_jerry.gz
-rw-r--r--. 1 tom root    32 Apr 28 16:36 file_by_tom.gz
-rw-r--r--. 1 root root  5112 Apr 28 17:04 functions.gz
drwxrwxr-x. 3 tom group1  56 Apr 28 17:10 jerry1
-rw-rw-r--. 1 tom root    31 Apr 28 16:46 jerry_file.gz
[root@localhost test]# gzip -d *.gz
[root@localhost test]# ll
total 24
-rw----r--. 1 tom root    12 Apr 28 16:07 file1
-rw-r--r--. 1 tom root     0 Apr 28 16:02 file2
-rw-rw-r--. 1 tom root     0 Apr 28 16:13 file_by_jerry
-rw-r--r--. 1 tom root     0 Apr 28 16:36 file_by_tom
-rw-r--r--. 1 root root 18281 Apr 28 17:04 functions
drwxrwxr-x. 3 tom group1  56 Apr 28 17:10 jerry1
-rw-rw-r--. 1 tom root     0 Apr 28 16:46 jerry_file
[root@localhost test]#

# zcat
[root@lanhai gzip_test]# zcat file1.gz
hello python

# bzip2
[root@lanhai gzip_test]# bzip2 file1
[root@lanhai gzip_test]# ls file1*
file1.bz2
[root@lanhai gzip_test]# bzip2 -d file1.bz2
[root@lanhai gzip_test]# ls file1*
file1

# zip
[root@lanhai gzip_test]# zip file1.zip file1 file2 file3
  adding: file1 (stored 0%)
  adding: file2 (stored 0%)
  adding: file3 (stored 0%)
[root@lanhai gzip_test]# ls
a b c file1 file1.zip file2 file3

# unzip
[root@lanhai gzip_test]# mkdir test
[root@lanhai gzip_test]# mv file1.zip test/
[root@lanhai gzip_test]# cd test/
[root@lanhai test]# ls
file1.zip
[root@lanhai test]# unzip file1.zip
```

```
Archive: file1.zip
extracting: file1
extracting: file2
extracting: file3
[root@lanhai test]# ls
file1 file1.zip file2 file3
```

## 2、打包

- tar
  - tar可以将多个目录或者文件打包成一个大文件，同时还可以结合gzip/bzip2，归档的同时进行压缩
  - 打出来的格式 .tar, .tar.gz, .tar.bz2, windows的WinRAR也支持tar.gz文件的解压缩
  - tar参数:
    - -c: 新建打包文件，可搭配-v来查看过程中被打包的文件名；create
    - -t: 查看打包文件的内容包含哪些文件名重点查看文件名；
    - -x: 解打包或压缩的功能，可以搭配-C（大写）在特定目录解开；
      - 特别留意的是，-c，-t，-x不可同时出现在一串命令行中；
    - -j : 通过bzip2的支持进行压缩/解压缩，此时文件名最好为\*.tar.bz2
    - -z: 通过gzip的支持进行压缩/解压缩，此时文件名最好为\*.tar.gz
  - -v: 在压缩/解压缩的过程中，将正在处理的文件名显示出来
  - -f filename: -f后面要接被处理的文件名。建议 -f 单独写一个参数
  - -C 目录: 这个参数用在解压缩时，若要在特定目录解压缩，可以使用这个参数
    - 这个目录必须要先存在

```
# tar -cvf 打包成test.tar, 针对当前目录下的所有文件
[root@localhost test]# tar -cvf test.tar ./*
./aa
./anaconda-ks.cfg
./file1
./passwd
./test.sh
[root@localhost test]# ls
aa anaconda-ks.cfg file1 passwd test.sh test.tar
[root@localhost test]#

# 接多个指定的要打包的文件
[root@localhost test]# tar -cvf test111.tar anaconda-ks.cfg file1 passwd
anaconda-ks.cfg
file1
passwd
[root@localhost test]# ls
aa anaconda-ks.cfg file1 passwd test111.tar test.sh test.tar
[root@localhost test]#

# 查看tar文件
[root@localhost test]# tar -tvf test.tar
-rw-r--r-- root/root      0 2022-04-28 15:45 ./aa
-rw----- root/root    1253 2022-04-24 20:17 ./anaconda-ks.cfg
-rw-rw-rw- root/root      0 2022-04-28 15:37 ./file1
-rw-r--r-- root/root     886 2022-04-27 11:51 ./passwd
-rwxr-xr-x root/root      33 2022-04-25 11:16 ./test.sh
[root@localhost test]#

[root@localhost test]# tar -tvf test111.tar
-rw----- root/root    1253 2022-04-24 20:17 anaconda-ks.cfg
-rw-rw-rw- root/root      0 2022-04-28 15:37 file1
-rw-r--r-- root/root     886 2022-04-27 11:51 passwd
```

```
[root@localhost test]#

# tar -xvf 解压到指定的文件夹/tmp/test/目录
[root@localhost test]# tar -xvf test.tar -C /tmp/test/
./aa
./anaconda-ks.cfg
./file1
./passwd
./test.sh
[root@localhost test]#

# 文件夹不存在的时候, 要报错
[root@localhost test]# tar -xvf test.tar -C ./aaaaabbbb
tar: aaaaaabbbb: Cannot open: No such file or directory
tar: Error is not recoverable: exiting now
[root@localhost test]#

# tar打包的同时进行压缩, 参数加z
[root@localhost test]# tar -czvf test.tar.gz file1 passwd test.sh
file1
passwd
test.sh
[root@localhost test]# ls
aa anaconda-ks.cfg file1 passwd test.sh test.tar.gz
[root@localhost test]#

# 查看.tar.gz格式 用-tzvf
[root@localhost test]# tar -tzvf test.tar.gz
-rw-rw-rw- root/root      0 2022-04-28 15:37 file1
-rw-r--r-- root/root     886 2022-04-27 11:51 passwd
-rwxr-xr-x root/root      33 2022-04-25 11:16 test.sh
[root@localhost test]#

[root@localhost test]# mkdir /root/tar_test
[root@localhost test]#
[root@localhost test]# tar -xzvf test.tar.gz -C /root/tar_test
file1
passwd
test.sh
[root@localhost test]# ls /root/tar_test
file1 passwd test.sh
[root@localhost test]#

# tar结合bzip2的压缩方式进行打包、查看、解压缩
[root@localhost test]# ls
aa anaconda-ks.cfg file1 passwd test.sh
[root@localhost test]# tar -cjvf test.tar.bz2 anaconda-ks.cfg file1 passwd
test.sh
anaconda-ks.cfg
file1
passwd
test.sh
[root@localhost test]# ls
aa anaconda-ks.cfg file1 passwd test.sh test.tar.bz2
[root@localhost test]# tar -tjvf test.tar.bz2
-rw----- root/root     1253 2022-04-24 20:17 anaconda-ks.cfg
-rw-rw-rw- root/root      0 2022-04-28 15:37 file1
-rw-r--r-- root/root     886 2022-04-27 11:51 passwd
-rwxr-xr-x root/root      33 2022-04-25 11:16 test.sh
[root@localhost test]#
[root@localhost test]#
[root@localhost test]# mkdir /root/bz2_test
[root@localhost test]#
[root@localhost test]# tar -xjvf test.tar.bz2 -C /root/bz2_test
anaconda-ks.cfg
file1
passwd
test.sh
[root@localhost test]# ls /root/bz2_test
```

```
anaconda-ks.cfg file1 passwd test.sh
[root@localhost test]#
```

### 3、软件安装

- 源码安装
  - Linux上的软件几乎都是经过GPL（GNU General Public License）授权，所以每个软件几乎都会提供源代码，并且你可以自行修改程序代码，以符合你个人的需求，这就是开放源码的优势；
  - Linux系统上真正识别的可执行文件是二进制文件，比如/bin/cat二进制程序代码；
  - 通过源码包安装软件的过程就是把软件的源代码编译成可执行的二进制文件然后进行安装；
  - 通过源码包安装的步骤：安装工具-》配置-》编译-》安装安装需要的编译器，如gcc；
    - 执行configure命令找到所需要的函数库、编译器及其他资料，生成Makefile文件；
    - 执行make命令，根据Makefile文件进行编译；
    - 执行make install命令完成安装。
  - 参考以下文档，完成Nginx服务器的安装。  
<https://zhuanlan.zhihu.com/p/38285324>

```
yum -y install gcc-c++ pcre* openssl*
cd /usr/local/src/
wget http://nginx.org/download/nginx-1.14.0.tar.gz
tar -zxvf nginx-1.14.0.tar.gz -C /usr/local/
cd /usr/local/nginx-1.14.0/
./configure --prefix=/usr/local/nginx
make && make install

# 启动nginx服务
[root@lanhai nginx-1.14.0]# cd /usr/local/nginx/sbin/
[root@lanhai sbin]# ./nginx
[root@lanhai sbin]# ps -ef | grep nginx
root      10968      1  0 19:06 ?        00:00:00 nginx: master process ./nginx
nobody    10969 10968    0 19:06 ?        00:00:00 nginx: worker process
root      10971  7902    0 19:06 pts/0    00:00:00 grep --color=auto nginx
[root@lanhai sbin]#

# 访问1
[root@lanhai sbin]# curl localhost
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
  body {
    width: 35em;
    margin: 0 auto;
  }
</style>
</html>

# 访问2
http://192.168.xxx.xxx
```

- rpm安装
  - RPM的全称是RedHat Package Manager，是RedHat发明的一种软件包安装工具和安装包格式；
  - RPM的最大特点是将你要安装的软件先编译好，并且打包成RPM机制的安装包，用户只需要下载相应的RPM包安装即可；
  - 优点：

- 1. 由于已经编译完成并且打包完毕，所以软件传输与安装上很方便（不需要再重新编译）；
- 2. 由于软件的信息都已经记录在Linux主机的数据库上，很方便查询、升级与卸载；
- 缺点：
  - 不能根据自己的需求灵活配置；
  - 可能存在打包环境与安装环境不一致导致安装失败。
  - 更新慢
- 安装
  - rpm -ivh xxx.rpm
- 查看
  - rpm -qa | grep xxx
- 卸载
  - rpm -e xxx.rpm
- 参考如下步骤，用rpm方式安装unix2dos

```
# 1、下载rpm包
https://developer.aliyun.com/packageSearch?word=dos2unix
wget https://mirrors.aliyun.com/centos/8-stream/BaseOS/x86_64/os/Packages/dos2unix-7.4.0-3.el8.x86_64.rpm?spm=a2c6h.13651111.0.0.652f2f70Ln4pDQ&file=dos2unix-7.4.0-3.el8.x86_64.rpm

# 2、安装
[root@lanhai soft]# rpm -ivh dos2unix-7.4.0-3.el8.x86_64.rpm
warning: dos2unix-7.4.0-3.el8.x86_64.rpm: Header V3 RSA/SHA256 Signature, key ID 8483c65d: NOKEY
Preparing... ##### [100%]
Updating / installing...
 1:dos2unix-7.4.0-3.el8 ##### [100%]
[root@lanhai soft]#

# 3、查看已安装的软件
[root@lanhai soft]# rpm -qa | grep dos2unix
dos2unix-7.4.0-3.el8.x86_64

# 4、删除已安装的软件
[root@lanhai soft]# rpm -qa | grep dos
dos2unix-7.4.0-3.el8.x86_64
[root@lanhai soft]# rpm -e dos2unix-7.4.0-3.el8.x86_64
[root@lanhai soft]# rpm -qa | grep dos
[root@lanhai soft]#
```

- yum安装/apt-get安装
  - 什么是yum安装？
    - 在Fedora和Redhat以及SUSE、CentOS中的软件包管理器
    - 基于rpm包管理
    - 能够从指定的服务器自动下载rpm包并且安装
    - 可以自动处理包依赖关系，并且一次安装所有依赖的软件包
  - /etc/yum.repos.d/\*.repo
    - mirrorlist
  - 安装
    - yum install xxx
  - 列出已安装的包
    - yum list installed | grep xxx
  - 删除
    - yum remove xxx
  - 安装wget
    - yum install -y wget
  - apt-get?
    - ubuntu系列的包管理命令，与yum类似
    - apt-get install wget

