

每天一个linux命令 (28) : tar命令 - peida - 博客园

通过SSH访问服务器，难免会要用到压缩，解压缩，打包，解包等，这时候tar命令就是必不可少的一个功能强大的工具。linux中最流行的tar是麻雀虽小，五脏俱全，功能强大。

tar命令可以为linux的文件和目录创建档案。利用tar，可以为某一特定文件创建档案（备份文件），也可以在档案中改变文件，或者向档案中加入新的文件。tar最初被用来在磁带上创建档案，现在，用户可以在任何设备上创建档案。利用tar命令，可以把一大堆的文件和目录全部打包成一个文件，这对于备份文件或将几个文件组合成为一个文件以便于网络传输是非常有用的。

首先要弄清两个概念：打包和压缩。打包是指将一大堆文件或目录变成一个总的文件；压缩则是将一个大的文件通过一些压缩算法变成一个小文件。

为什么要区分这两个概念呢？这源于Linux中很多压缩程序只能针对一个文件进行压缩，这样当你想要压缩一大堆文件时，你得先将这一大堆文件先打成一个包（tar命令），然后再用压缩程序进行压缩（gzip bzip2命令）。

linux下最常用的打包程序就是tar了，使用tar程序打出来的包我们常称为tar包，tar包文件的命令通常都是以.tar结尾的。生成tar包后，就可以用其它的程序来进行压缩。

1. 命令格式：

tar[必要参数][选择参数][文件]

2. 命令功能：

用来压缩和解压文件。tar本身不具有压缩功能。他是调用压缩功能实现的

3. 命令参数：

必要参数有如下：

- A 新增压缩文件到已存在的压缩
- B 设置区块大小
- c 建立新的压缩文件
- d 记录文件的差别
- r 添加文件到已经压缩的文件
- u 添加改变了和现有的文件到已经存在的压缩文件
- x 从压缩的文件中提取文件
- t 显示压缩文件的内容
- z 支持gzip解压文件
- j 支持bzip2解压文件
- Z 支持compress解压文件
- v 显示操作过程
- l 文件系统边界设置
- k 保留原有文件不覆盖
- m 保留文件不被覆盖
- W 确认压缩文件的正确性

可选参数如下：

- b 设置区块数目

-C 切换到指定目录

-f 指定压缩文件

--help 显示帮助信息

--version 显示版本信息

4. 常见解压/压缩命令

tar

解包: tar xvf FileName.tar

打包: tar cvf FileName.tar DirName

(注: tar是打包, 不是压缩!)

.gz

解压1: gunzip FileName.gz

解压2: gzip -d FileName.gz

压缩: gzip FileName

.tar.gz 和 .tgz

解压: tar zxvf FileName.tar.gz

压缩: tar zcvf FileName.tar.gz DirName

.bz2

解压1: bzip2 -d FileName.bz2

解压2: bunzip2 FileName.bz2

压缩: bzip2 -z FileName

.tar.bz2

解压: tar jxvf FileName.tar.bz2

压缩: tar jcvf FileName.tar.bz2 DirName

.bz

解压1: bzip2 -d FileName.bz

解压2: bunzip2 FileName.bz

压缩: 未知

.tar.bz

解压: tar jxvf FileName.tar.bz

压缩: 未知

.Z

解压: uncompress FileName.Z

压缩: compress FileName

.tar.Z

解压: tar Zxvf FileName.tar.Z

压缩: tar Zcvf FileName.tar.Z DirName

.zip

解压: unzip FileName.zip

压缩: zip FileName.zip DirName

.rar

解压: rar x FileName.rar

压缩: rar a FileName.rar DirName

5. 使用实例

实例1: 将文件全部打包成tar包

命令:

```
tar -cvf log.tar log2012.log
```

```
tar -zcvf log.tar.gz log2012.log
```

```
tar -jcvf log.tar.bz2 log2012.log
```

输出：

```
[root@localhost test]#ls -al log2012.log
---xrw-r-- 1 root root 302108 11-13 06:03 log2012.log
[root@localhost test]#tar -cvf log.tar log2012.log
log2012.log
[root@localhost test]#tar -zcvf log.tar.gz log2012.log
log2012.log
[root@localhost test]#tar -jcvf log.tar.bz2 log2012.log
log2012.log
[root@localhost test]#ls -al *.tar*
-rw-r--r-- 1 root root 307200 11-29 17:54 log.tar
-rw-r--r-- 1 root root 1413 11-29 17:55 log.tar.bz2
-rw-r--r-- 1 root root 1413 11-29 17:54 log.tar.gz
```

说明：

tar -cvf log.tar log2012.log 仅打包，不压缩！

tar -zcvf log.tar.gz log2012.log 打包后，以 **gzip** 压缩

tar -zcvf log.tar.bz2 log2012.log 打包后，以 **bzip2** 压缩

在参数 f 之后的文件档名是自己取的，我们习惯上都用 **.tar** 来作为辨识。如果加 z 参数，则以 **.tar.gz** 或 **.tgz** 来代表 **gzip** 压缩过的 **tar**包；如果加 j 参数，则以 **.tar.bz2** 来作为 **tar**包名。

实例2：查阅上述 **tar**包内有哪些文件

命令：

```
tar -ztvf log.tar.gz
```

输出：

```
[root@localhost test]#tar -ztvf log.tar.gz
---xrw-r-- root/root 302108 2012-11-13 06:03:25 log2012.log
```

说明：

由于我们使用 **gzip** 压缩的 **log.tar.gz**，所以要查阅 **log.tar.gz** 包内的文件时，就得要加上 **z** 这个参数了。

实例3：将 **tar**包解压缩

命令：

```
tar -zxvf /opt/soft/test/log.tar.gz
```

输出：

```
[root@localhost test3]#ll
总计 0 [root@localhost test3]#tar -zxvf /opt/soft/test/log.tar.gz
log2012.log
[root@localhost test3]#ls
log2012.log
[root@localhost test3]#
```

说明：

在预设的情况下，我们可以将压缩档在任何地方解开的

实例4：只将 **/tar** 内的 部分文件解压出来

命令：

```
tar -zxvf /opt/soft/test/log30.tar.gz log2013.log
```

输出：

```
[root@localhost test]#tar -zcvf log30.tar.gz log2012.log log2013.log
log2012.log
log2013.log
[root@localhost test]#ls -al log30.tar.gz
-rw-r--r-- 1 root root 1512 11-30 08:19 log30.tar.gz
[root@localhost test]#tar -zxvf log30.tar.gz log2013.log
log2013.log
[root@localhost test]#ll
-rw-r--r-- 1 root root 1512 11-30 08:19 log30.tar.gz
[root@localhost test]#cd test3
[root@localhost test3]#tar -zxvf /opt/soft/test/log30.tar.gz log2013.log
log2013.log
[root@localhost test3]#ll
总计 4
-rw-r--r-- 1 root root 61 11-13 06:03 log2013.log
[root@localhost test3]#
```

说明：

我可以透过 `tar -zvf` 来查阅 `tar` 包内的文件名称，如果单只要一个文件，就可以透过这个方式来解压部分文件！

实例5：文件备份下来，并且保存其权限

命令：

```
tar -zcvpf log31.tar.gz log2014.log log2015.log log2016.log
```

输出：

```
[root@localhost test]#ll
总计 0
-rw-r--r-- 1 root root 0 11-13 06:03 log2014.log
-rw-r--r-- 1 root root 0 11-13 06:06 log2015.log
-rw-r--r-- 1 root root 0 11-16 14:41 log2016.log
[root@localhost test]#tar -zcvpf log31.tar.gz log2014.log log2015.log log2016.log
log2014.log
log2015.log
log2016.log
[root@localhost test]#cd test6
[root@localhost test6]#ll
[root@localhost test6]#tar -zxvpf /opt/soft/test/log31.tar.gz
log2014.log
log2015.log
log2016.log
[root@localhost test6]#ll
总计 0
-rw-r--r-- 1 root root 0 11-13 06:03 log2014.log
-rw-r--r-- 1 root root 0 11-13 06:06 log2015.log
-rw-r--r-- 1 root root 0 11-16 14:41 log2016.log
```

```
[root@localhost test6]#
```

说明：

这个 **-p** 的属性是很重要的，尤其是当您要保留原本文件的属性时

实例6：在文件夹当中，比某个日期新的文件才备份

命令：

```
tar -N "2012/11/13" -zcvf log17.tar.gz test
```

输出：

```
[root@localhost soft]#tar -N "2012/11/13" -zcvf log17.tar.gz test
tar: Treating date `2012/11/13' as 2012-11-13 00:00:00 + 0 nanoseconds
test/test/log31.tar.gz
test/log2014.log
test/linklog.log
test/log2015.log
test/log2013.log
test/log2012.log
test/log2017.log
test/log2016.log
test/log30.tar.gz
test/log.tar
test/log.tar.bz2
test/log.tar.gz
```

说明：

实例7：备份文件夹内容是排除部分文件

命令：

```
tar --exclude scf/service -zcvf scf.tar.gz scf/*
```

输出：

```
[root@localhost test]#tree scf
scf
|-- bin
|-- doc
|-- lib
`-- service

`-- deploy

|-- info

`-- product

7 directories, 0 files
[root@localhost test]#tar --exclude scf/service -zcvf scf.tar.gz scf/*
scf/bin/
scf/doc/
scf/lib/
```

```
[root@localhost test]#
```