

tar命令的详细解释 - eros的linux平台技术资料库 - CSDN博客

tar [-ctzjvfpPN] 文件与目录

参数:

-c : 建立一个压缩文件的参数指令(create 的意思);

-x : 解开一个压缩文件的参数指令!

-t : 查看 tarfile 里面的文件!

特别注意, 在参数的下达中, c/x/t 仅能存在一个! 不可同时存在!

因为不可能同时压缩与解压缩。

-z : 是否同时具有 gzip 的属性? 亦即是否需要用 gzip 压缩?

-j : 是否同时具有 bzip2 的属性? 亦即是否需要用 bzip2 压缩?

-v : 压缩的过程中显示文件! 这个常用, 但不建议用在背景执行过程!

-f : 使用档名, 请留意, 在 f 之后要立即接档名喔! 不要再加参数!

例如使用『 tar -zcvfP tfile sfile』就是错误的写法, 要写成

『 tar -zcvPf tfile sfile』才对喔!

-p : 使用原文件的原来属性(属性不会依据使用者而变)

-P : 可以使用绝对路径来压缩!

-N : 比后面接的日期(yyyy/mm/dd)还要新的才会被打包进新建的文件中!

--exclude FILE: 在压缩的过程中, 不要将 FILE 打包!

范例:

范例一: 将整个 /etc 目录下的文件全部打包成为 /tmp/etc.tar

tar -cvf /tmp/etc.tar /etc<==仅打包, 不压缩!

tar -zcvf /tmp/etc.tar.gz /etc<==打包后, 以 gzip 压缩

tar -jcvf /tmp/etc.tar.bz2 /etc<==打包后, 以 bzip2 压缩

特别注意, 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 file ~

如果加 j 参数, 则以 .tar.bz2 来作为附档名啊~

上述指令在执行的时候, 会显示一个警告讯息:

『tar: Removing leading `/' from member names』那是关于绝对路径的特殊设定。

范例二: 查阅上述 /tmp/etc.tar.gz 文件内有哪些文件?

tar -ztvf /tmp/etc.tar.gz

由於我们使用 gzip 压缩, 所以要查阅该 tar file 内的文件时,

就得要加上 z 这个参数了! 这很重要的!

范例三: 将 /tmp/etc.tar.gz 文件解压缩在 /usr/local/src 底下

[root@linux ~]# **cd /usr/local/src**

[root@linux src]#**tar -zxvf /tmp/etc.tar.gz**

在预设的情况下, 我们可以将压缩档在任何地方解开的! 以这个范例来说,

我先将工作目录变换到 /usr/local/src 底下, 并且解开 /tmp/etc.tar.gz ,

则解开的目录会在 /usr/local/src/etc 呢! 另外, 如果您进入 /usr/local/src/etc

则会发现, 该目录下的文件属性与 /etc/ 可能会有所不同喔!

范例四: 在 /tmp 底下, 我只想要将 /tmp/etc.tar.gz 内的 etc/passwd 解开而已

[root@linux ~]#**cd /tmp**

[root@linux tmp]# **tar -zxvf /tmp/etc.tar.gz etc/passwd**

我可以透过 tar -ztvf 来查阅 tarfile 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下达! 注意到! etc.tar.gz 内的根目录 / 是被拿掉了!

范例五：将 /etc/ 内的所有文件备份下来，并且保存其权限！

```
[root@linux ~]#tar -zxvpf /tmp/etc.tar.gz /etc
```

这个 -p 的属性是很重要的，尤其是当您保留原本文件的属性时！

范例六：在 /home 当中，比 2005/06/01 新的文件才备份

```
[root@linux ~]#tar -N '2005/06/01' -zcvf home.tar.gz /home
```

范例七：我要备份 /home, /etc，但不要 /home/dmtsai

```
[root@linux ~]# tar --exclude /home/dmtsai -zcvf myfile.tar.gz /home/* /etc
```

范例八：将 /etc/ 打包后直接解开在 /tmp 底下，而不产生文件！

```
[root@linux ~]# cd /tmp
```

```
[root@linux tmp]# tar -cvf - /etc | tar -xvf -
```

这个动作有点像是 cp -r /etc /tmp 啦～依旧是有其有用途的！

要注意的地方在於输出档变成 - 而输入档也变成 -，又有一个 | 存在～

这分别代表 standard output, standard input 与管线命令啦！

这部分我们会在 Bash shell 时，再次提到这个指令跟大家再解释啰！