

## 使用 tar 备份

### 内容提要

1. 熟悉 GNU tar 工具
2. 使用 tar 进行备份和恢复
3. 学会使用 tar 编写备份脚本

## tar 命令

tar是一个已移植到Linux中的经典UNIX命令。tar是Tape ARchive（磁带归档）的缩写，最初设计用于将文件打包到磁带上。它是一个基于文件的命令，它本质上是连续地、首尾相连地堆放文件。

使用tar可以打包整个目录树，这使得它特别适合于备份。归档文件可以全部还原，或从中展开单独的文件和目录。备份可以保存到基于文件的设备或磁带设备上。文件可以在还原时重定向，以便将它们重新放到一个与最初保存它们的目录（或系统）不同的目录（或系统）。tar是与文件系统无关的，它可以使用在ext2、ext3、jfs、Reiser和其他文件系统上。

在Linux世界里经常使用tar命令进行备份，此时将涉及tar命令的一些参数。 tar命令的完整格式是：

```
tar <operation> [options] <files_to_backup_or_restore>
```

其中：

- **operation:** 用于指定 tar 要进行的操作
- **options:** 用于指定一系列的选项
- **files\_to\_backup\_or\_restore:** 用于给出要备份或要恢复的文件或目录名，在指定目录时也包括了这个目录下的子目录。

下表给出了 tar 命令的操作说明。

操作	说明
[-]A	连接多个归档文件为一个归档文件。
[-]c	用于创建一个新的存档文件。
[-]x	从归档文件中恢复备份文件。
[-]t	用于列出一个存档文件中的文件名。
[-]u	仅仅添加比存档文件中更新的文件。即，用新增的文件取代原备份文件，如果在归档文件中找不到要更新的文件，则把它追加到备份文件的最后。
[-]d	将归档文件的内容与文件系统上的当前文件作比较。
[-]r	将文件追加到指定的归档文件中。
--delete	从归档文件中删除指定的文件。

下表给出了 tar 命令的常用选项说明。

选项	说明
-f name	使用name指定存档文件名或设备名。
-v	列出处理的详细信息。
-z	用GNU的gzip压缩文件或解压。
-j	用GNU的bzip2压缩文件或解压。
-C directory	将当前目录切换到 directory。
-M	创建/列出/恢复多卷存档文件，以便在几个备份介质中存放。
-N DATE	指定仅对那些比DATE新的文件进行操作。
-p	表示希望保留文件许可权限。
-P	保留文件的绝对路径，即不去掉/。
-w	要求等待用户确认每一个操作。
-W	表示在写入备份内容到备份设备以后再读出来进行验证以提高可靠性。
-T filename	从指定的文件中读需要备份或恢复的文件名。
-X filename	不处理给定文件中列出的文件。
--exclude=PATTERN	不处理指定的文件。

## 使用 tar 备份文件

通常将备份文件存储在单独的分区中，可以是系统本地硬盘中的一个分区，也可以是另外挂装的移动设备中的一个分区。因此，在备份之前，应该创建挂载点目录，并挂装文件系统。

```
# mkdir /backups
# mount -t ext3 /dev/sdc1 /backups
# mkdir /backups/logs /backups/last-full
```

### 指定要备份的文件或目录

1、使用如下的命令备份指定的一个或多个目录到 **/backups** 目录中的一个归档文件

```
# tar -zcvpf /backups/full-backup.tar.gz /home
# tar -zcvpf /backups/full-backup.tar.gz /home /etc
```

2、使用命令替换生成要备份的目录

下面的命令备份整个 **/** 系统，除了 **mnt**、**media**、**dev**、**proc**、**backups** 目录和 **lost+found** 目录。

```
# tar -zcvpf /backups/full-backup.tar.gz -C / \
> $(ls / | egrep -v "backups|mnt|media|dev|lost+found|proc")
```

3、使用 **exclude** 选项剔除无需备份的文件或目录

1) 下面的命令备份整个 **/** 系统，除了 **mnt**、**media**、**dev**、**proc**、**backups**、**var/spool/squid** 目录和所有的 **lost+found** 目录。

```
# tar -zcvpf /backups/full-backup.tar.gz -C / \
> --exclude=mnt --exclude=media --exclude=dev --exclude=proc \
> --exclude=backups --exclude=*/lost+found \
> --exclude=var/spool/squid
> . # 此处的 “.” 表示要备份根目录（因为前面的 “-C /” 将当前目录切换为根目录）
```

2) 下面的命令备份 **etc**、**home**、**usr/local** 和 **var/spool**（不包括 **var/spool/squid**）目录。

```
# tar -zcvpf /backups/full-backup.tar.gz -C / \
> --exclude=var/spool/squid \
> etc home usr/local var/spool
```

4、将要备份的文件或目录名放入文本文件

```
# cat whatsbackup.txt
/home
/etc
/usr/local
/root

# tar -zcvpf /backups/full-backup.tar.gz -T whatsbackup.txt
```

1. **-T** 参数后指定的文件中，不能使用文件通配符
2. 可以使用 **ls** 或 **find** 命令生成 **whatsbackup.txt** 文件

### 为归档文件名添加时间

在归档文件名中使用带有命令替换的 **date** 命令。

1、只添加日期

```
# tar -zcvpf /backups/full-backup_$(date +%F).tar.gz /home
# ls -l /backups
-rw-r--r-- 1 root root 125855 01-29 19:09 full-backup_2008-01-29.tar.gz
```

2、添加日期、小时和分钟

```
# tar -zcvpf /backups/full-backup_$(date +%Y%m%d-%H%M).tar.gz /home
# ls -l /backups
-rw-r--r-- 1 root root 125855 01-29 20:17 full-backup_20080129-2017.tar.gz
```

## 保留备份日志

使用输出重定向将 **tar** 命令的输出结果存入文件。

```
# tar -zcvpf /backups/full-backup_$(date +%F).tar.gz /home \  
> > /backups/logs/full-backup_$(date +%F).log \  
> 2> /backups/logs/full-backup_$(date +%F).err  
# tar -zcvpf /backups/full-backup_$(date +%Y%m%d-%H%M).tar.gz /home \  
> > /backups/logs/full-backup_$(date +%Y%m%d-%H%M).log \  
> 2> /backups/logs/full-backup_$(date +%Y%m%d-%H%M).err
```

## 增量备份

### 1、使用带 **N** 选项的 **tar** 命令实现增量备份

下面的命令将备份 **/home** 目录自 **2008-01-29** 以来修改过的文件

```
# tar -N 2008-01-29 -zcvpf /backups/inc-backup_$(date +%F).tar.gz /home
```

下面的命令将备份 **/home** 目录昨天以来修改过的文件

```
# tar -N $(date -d yesterday +%F) \  
> -zcvpf /backups/inc-backup_$(date +%F).tar.gz /home
```

以下两种书写方式均可，但含义有所不同：

- **-N yesterday**：比昨天的当前时间新的文件，例如：若当前时间为 **2:31**，则表示自昨天 **2:31** 以来的新文件
- **-N \$(date -d yesterday +%F)**：自昨天 **0:00** 以来的新文件

下面的命令使用命令替换从文本文件中获得日期

```
# tar -N $(cat /backups/last-full/full-backup-date) \  
> -zcvpf /backups/inc-backup_$(date +%F).tar.gz /home
```

**/backups/last-full/full-backup-date** 是在创建完全备份的同时使用如下命令创建的文件：

```
date +%F > /backups/last-full/full-backup-date
```

### 2、使用 **find** 命令获取增量备份的文件列表

使用如下命令找出 **n** 天（如：**7**天）内修改过的文件，生成备份内容的文件列表

```
# find /home /etc /root -mtime -7 -print > /backups/logs/inc-backup_$(date +%F).log
```

对指定文件列表中的文件实现增量备份

```
# tar -zcvpf /backups/inc-backup_$(date +%F).tar.gz \  
> -T /backups/logs/inc-backup_$(date +%F).log
```

## 使用磁带备份

与向硬盘文件备份有所不同：

- **-f** 参数后要跟磁带机的设备名，如：**/dev/sd0**、**/dev/nst0**。
- 由于设备名是固定的，不能像硬盘文件那样在文件名中添加备份时间。此时可以使用 **--label** 选项将有关的备份信息写入归档本身。
- 反对将压缩资料写入磁带，因为如果磁带上资料有一部份损坏，会失去整个备份。

下面是两个使用磁带备份的例子：

```
# tar -cvpf /dev/st0 --label="Full-backup created on $(date +%Y%m%d-%H%M)." \  
> -C / --exclude=var/spool/ etc home usr/local var/spool  
  
# tar -cvpf /dev/st0 --label="Incremental-backup created on $(date +%F)." \  
> -N $(date -d yesterday +%F) /etc /home
```

除了使用 **/dev/st0** 之外，还可以使用 **/dev/nst0**，他们的区别在于：

- **/dev/st0** **--** 磁带在备份完毕会自动回带
- **/dev/nst0** **--** 磁带在备份完毕不会自动回带，有可能将好几组备份写到同一磁带

对于不能自动回带的设备，可以使用如下的命令进行手工回带并退出磁带：

```
# mt -f /dev/nst0 rewind
# mt -f /dev/nst0 offline
```

## 使用 tar 恢复文件

当建立一个归档时，**tar** 会将文件路径前面的 / (斜线) 去掉。因此，默认恢复文件时将文件释放到当前目录下。若要将文件释放到指定的目录下，可以使用 **-C** 选项指定。

1. 建议在释放文件之前先使用 **-t** 选项替换 **-x** 选项，进行检查。
2. 一个更安全的方法是在不同的目录释放文件 (例如您的 **home** 目录)，然后比对确认后，再将释放的文件移动到原始位置。

## 恢复全部文件

```
# 从磁盘文件恢复全部文件
# tar -zxvpf /backups/full-backup_2008-01-29.tar.gz -C /
# 从磁带中恢复全部文件
# tar -xvpf /dev/st0 -C /
```

## 恢复指定文件

```
# 从磁盘文件恢复指定的文件
# tar -zxvpf /backups/full-backup_2008-01-29.tar.gz -C / \
> etc/passwd etc/shadow
# 从磁带中恢复指定的文件
# tar -xvpf /dev/st0 -C / etc/passwd etc/shadow
```

## 恢复文件的顺序

当系统中存在一个完全备份和多个增量备份时，应该首先恢复完全备份，然后按时间顺序依次恢复增量备份。

例如：系统存在如下的备份文件：

```
full-backup_2008-01-29.tar.gz
inc-backup_2008-01-30.tar.gz
inc-backup_2008-01-31.tar.gz
inc-backup_2008-02-01.tar.gz
inc-backup_2008-02-02.tar.gz
inc-backup_2008-02-03.tar.gz
```

应该按照上面的列出顺序依次进行恢复。

当系统中存在一个完全备份和多个基于此完全备份的累计备份时，应该首先恢复完全备份，然后再恢复一个累计备份即可。

例如：系统存在如下的备份文件：

```
full-backup_2008-01-29.tar.gz
diff-backup_2008-01-30.tar.gz
diff-backup_2008-01-31.tar.gz
diff-backup_2008-02-01.tar.gz
diff-backup_2008-02-02.tar.gz
diff-backup_2008-02-03.tar.gz
```

要恢复到 **2008-02-02** 备份前的状态，只要依次恢复 **full-backup\_2008-01-29.tar.gz** 和 **diff-backup\_2008-02-02.tar.gz** 即可。

## 使用 tar 的备份脚本

### 完全备份和累计备份

下面的脚本来自 <http://www.faqs.org/docs/securing/chap29sec306.html> [<http://www.proxyserve.net/index.php?q=aHR0cDovL3d3dy5mYXFzLm9yZy9kb2NzL3NlY3VyaW5nL2NoYXAyOXBNIYzMwNi5odG1s>]

```
#!/bin/sh
# NAME : /etc/cron.daily/backup.cron
# FROM : http://www.faqs.org/docs/securing/chap29sec306.html
# full and incremental backup script
```

```

# created 07 February 2000
# Based on a script by Daniel O'Callaghan <danny@freebsd.org>
# and modified by Gerhard Mourani <gmourani@videotron.ca>

# 更改如下的 5 个变量以适应您的 computer/backup

COMPUTER=deep                # 本机的机器名称
DIRECTORIES="/home"         # 要备份的目录
BACKUPDIR=/backups          # 备份文件存放的目录
TIMEDIR=/backups/last-full  # 存放上次完全备份时间文件的目录
TAR=/bin/tar                 # tar 的完整路径

# 您不应该更改以下的内容
LANG=C                       # 设置语言（避免产生中文日期输出）
PATH=/usr/local/bin:/usr/bin:/bin
DOW=`date +%a`               # 周几 例如: Mon
DOM=`date +%d`               # 天数 例如: 27
DM=`date +%d%b`              # 天数和月份 例如: 27Sep

# 每月 1 日生成完全备份。
# 每周日生成完全备份，并覆盖上周日生成的完全备份。
# 月内的其他日子生成累计备份，每个累计备份都覆盖上周生成的累计备份。
#
# 如果 NEWER = "", 那么备份指定目录中的所有文件（完全备份），
# 否则备份比 $NEWER 新的文件，NEWER 的值由每周日写在
# /backups/last-full/deep-full-date 的内容获得

# 每月的完全备份
if [ $DOM = "01" ]; then
    NEWER=""
    $TAR $NEWER -cf $BACKUPDIR/$COMPUTER-$DM.tar $DIRECTORIES
fi

# 每周的完全备份
if [ $DOW = "Sun" ]; then
    NEWER=""
    NOW=`date +%d-%b`

    # 更新完全备份的日期
    echo $NOW > $TIMEDIR/$COMPUTER-full-date
    $TAR $NEWER -cf $BACKUPDIR/$COMPUTER-$DOW.tar $DIRECTORIES

# 生成累计备份 - 覆盖上周相应的文件
else

    # 获取上次完全备份的日期
    NEWER=`--newer `cat $TIMEDIR/$COMPUTER-full-date` `
    $TAR $NEWER -cf $BACKUPDIR/$COMPUTER-$DOW.tar $DIRECTORIES
fi

```

为了使用上面的脚本，需要执行如下步骤：

```

# 编辑文件 /etc/cron.daily/backup.cron
# vi /etc/cron.daily/backup.cron
# 添加执行权限
# chmod +x /etc/cron.daily/backup.cron

```

1. 在使用脚本之前，要保证存放备份文件和存放完全备份时间的目录已经存在。
2. 若您希望保留备份的属性，请在 **tar** 命令中添加 **-p** 选项；若您希望压缩数据，请在 **tar** 命令中添加 **-z** 或 **-j** 选项。
3. 若您不是在一个月的 1 日或非周日开始运行此脚本，可以依照脚本中的文件名约定手工创建一个完全备份文件，并用如下的命令将完全备份的日期写入 **/backups/last-full** 目录下的文件 **\$COMPUTER-full-date** 中，此脚本默认的文件名是 **deep-full-date**。

```
# LANG=C; date +%d%b > /backups/last-full/deep-full-date
```

## 完全备份和增量备份

考察如下的备份脚本：

```

#!/bin/bash
# NAME : backup.sh
# 2006-03-24      Ver 1.0      Osmond
#
# 请根据您的需要修改如下变量的值
BACKUPDIR=/backups/dat      # 备份文件存放的目录
BACKUPLISTDIR=/backups/list # 要备份目录的文件列表
BACKUPLLOGDIR=/backups/logs # 存放上次完全备份时间文件的目录
TARPATH="/bin/tar"          # tar 的完整路径
TARCOMPRESS="-z"            # 压缩算法 ("z"|"j"| "")
BACKUPKEEPDAYS=28           # 保留多少天的备份
MAILUSER="smartraining.cn@gmail.com" # 设置备份管理员的 E-mail
COMPUTER=$(hostname -s)      # 本机的机器名称

```

```
# 若备份目录不存在，则创建之
[ ! -e "/backups/dat" ] && mkdir -p /backups/dat
[ ! -e "/backups/list" ] && mkdir -p /backups/list
[ ! -e "/backups/logs" ] && mkdir -p /backups/logs

PATH=/usr/local/bin:/usr/bin:/bin
DOW=`date +%w`           # 周几 [0-6]
DATE=`date +%F`          # 日期 YYYY-MM-DD

case $TARCOMPRESS in
    "-z") suffix=".gz" ;;
    "-j") suffix=".bz2" ;;
    *) suffix="" ;;
esac

# 生成备份文件名的变量
BACKUP_FULL="/$BACKUPDIR/full-backup_`$COMPUTER`-$DATE.tar$suffix"
BACKUP_INC="/$BACKUPDIR/inc-backup_`$COMPUTER`-$DATE.tar$suffix"

# 生成 tar 命令及参数的变量
TAR="$TARPATH $TARCOMPRESS -cvpf"

# 生成备份日志文件名的变量
BACKUP_FULL_LOG="/$BACKUPLOGDIR/full-backup_`$COMPUTER`-$DATE.log"
BACKUP_FULL_ERR="/$BACKUPLOGDIR/full-backup_`$COMPUTER`-$DATE.err"
BACKUP_INC_LOG="/$BACKUPDIR/inc-backup_`$COMPUTER`-$DATE.log"
BACKUP_INC_ERR="/$BACKUPDIR/inc-backup_`$COMPUTER`-$DATE.err"

# 生成完全备份和增量备份的列表文件名的变量
BACKUP_FULL_LIST=$BACKUPLISTDIR/fulldirlist
BACKUP_INC_LIST=$BACKUPLISTDIR/incfilelist

if [ $DOW == "0" -o $1 == "init" ]
then
    # 周日进行完全备份
    echo -e "\n==== $DATE (FULL) =====\n" > $BACKUP_FULL_LOG
    echo -e "\n==== $DATE (FULL ERR) =====\n" > $BACKUP_FULL_ERR
    $TAR $BACKUP_FULL -T $BACKUP_FULL_LIST >>$BACKUP_FULL_LOG 2>>$BACKUP_FULL_ERR
    # 将错误日志作为内容，备份日志作为附件发送给管理员
    cat $BACKUP_FULL_ERR | mutt -s "Backup LOG -- $COMPUTER" -a $BACKUP_FULL_LOG $MAILUSER
else
    # 周1-6进行增量备份
    echo -e "\n==== $DATE (INC) =====\n" > $BACKUP_INC_LOG
    echo -e "\n==== $DATE (INC ERR) =====\n" > $BACKUP_INC_ERR
    cp /dev/null $BACKUP_INC_LIST
    # 生成增量备份的文件列表
    while read dirname
    do
        find $dirname -mtime -1 -print >> $BACKUP_INC_LIST
    done < $BACKUP_FULL_LIST
    $TAR $BACKUP_INC -T $BACKUP_INC_LIST >>$BACKUP_INC_LOG 2>>$BACKUP_INC_ERR
    cat $BACKUP_INC_ERR | mutt -s "Backup LOG -- $COMPUTER" -a $BACKUP_INC_LOG $MAILUSER
fi

# 清理备份文件和备份日志文件
find $BACKUPDIR $BACKUPLOGDIR -mtime +$BACKUPKEEPDAYS -exec rm -f {} \;
```

为了使用上面的脚本，需要执行如下步骤：

### 1、编辑文件 /root/bin/backup.sh 并添加执行权限

```
# vi /root/bin/backup.sh
# chmod +x /root/bin/backup.sh
```

### 2、创建完全备份目录列表文件

```
# vi /backups/list/fulldirlist
```

请根据您的需要决定此文件的内容，例如：

```
/etc
/var
/usr/local
/home
/root
```

### 3、安排cron备份任务

```
# crontab -e
```

若每天 0: 20 执行脚本，则 crontab 为：

```
20 0 * * * /root/bin/backup.sh
```

您可以根据自己的需要进行调整。

#### 4、若当前不是周日，执行脚本进行首次完全备份

```
# /root/bin/backup.sh init
```

#### 参考

- <http://www.faqs.org/docs/securing/chap29sec306.html> [<http://www.proxyservice.net/index.php?q=aHR0cDovL3d3dy5mYXFzLm9yZy9kb2NzL3NiY3VyaW5nL2NoYXAYOXNiYzMwNi5odG1s>]
- [http://linux.vbird.org/linux\\_basic/0580backup.php#VBird\\_strategy](http://linux.vbird.org/linux_basic/0580backup.php#VBird_strategy) [[http://www.proxyservice.net/index.php?q=aHR0cDovL2xpbmV4LnZiaXJkLm9yZy9saW51eF9iYXNpYy8wNTgwYmFja3VwLnBocCNWQmlyZF9zdHJhdGVneQ%3D%3D##VBird\\_strategy](http://www.proxyservice.net/index.php?q=aHR0cDovL2xpbmV4LnZiaXJkLm9yZy9saW51eF9iYXNpYy8wNTgwYmFja3VwLnBocCNWQmlyZF9zdHJhdGVneQ%3D%3D##VBird_strategy)]
- 显示源文件
- 登录