

Практическая работа «Архивация»

Архивирование файлов - это эффективный способ создания резервных копий и передачи больших файлов. Zip и unzip - наиболее часто используемые утилиты сжатия.

Чтобы сжать файл слов с помощью команды `gzip`, выполните следующие команды:

```
cp /usr/share/dict/words .
ls -l words
gzip words
ls -l words.gz
```

Чтобы извлечь сжатый файл `words.gz`, созданный в примере выше, выполните следующую команду:

```
ls -l words*
gunzip words.gz
ls -l words*
```

Чтобы сохранить исходный файл при сжатии с помощью команды `gzip`, используйте параметр `-c`. Это позволяет вам перенаправить двоичный вывод команды `gzip` в другой файл. Чтобы сделать это с файлом слов, выполните следующие команды:

```
gzip -c words > words.gz
ls -l words*
```

```
root@L-FW:~# ls -al w*
-rw-r--r-- 1 root root 938848 May  8 23:19 words
-rw-r--r-- 1 root root 255996 May  8 23:31 words.gz
```

Параметры `-l` и `-v` команды `gunzip` можно использовать для просмотра отношения сжатия архивного файла к исходному файлу. Чтобы просмотреть степень сжатия в архивном файле, выполните следующую команду:

```
gunzip -l words.gz
```

```
root@L-FW:~# gunzip -l words.gz
      compressed      uncompressed  ratio uncompressed_name
      255996           938848    72.7% words
root@L-FW:~#
```

Чтобы просмотреть степень сжатия архивированного файла при распаковке файла, выполните следующую команду, нажав `y`, чтобы перезаписать файл:

```
gunzip -v words.gz
```

`y`

Чтобы сжать файл слов в текущем каталоге с помощью команды `bzip2`, выполните следующие команды:

```
bzip2 words
ls -l words*
```

Подобно командам `gzip` и `gunzip`, команды `bzip2` и `bunzip2` также используются для сжатия и распаковки файла. Алгоритм сжатия, используемый обеими командами, отличается, но использование очень похоже. Расширение файлов, созданных командой `bzip2`, - `.bz2`.

Чтобы извлечь сжатый файл `words.bz2`, созданный в примере выше, выполните следующие команды:

```
bunzip2 words.bz2
ls -l words*
```

Команда `bzip2` не поддерживает отдельный параметр для рекурсии. Таким образом, `bzip2` нельзя использовать для сжатия вложенной структуры каталогов.

Чтобы создать архив `tar` из каталога `/etc/vim` и поместить файл `tar` в текущий каталог, выполните следующие команды:

```
tar -cf vim.tar /etc/vim
```

```
ls -l *.tar
```

Команда **tar**, которая происходит от термина «архив ленты», является доминирующей командой архивирования файлов, используемой в системах на основе Linux. Основная функция tar - объединить несколько файлов в один файл. Команда tar предоставляет три основные функции - создание, просмотр и извлечение архивов.

Чтобы просмотреть содержимое файла vim.tar, выполните следующую команду:

```
tar -tf vim.tar
```

Чтобы просмотреть подробный список содержимого файла vim.tar, выполните следующую команду:

```
tar -tvf vim.tar
```

Чтобы извлечь файлы из vim.tar в текущий каталог, выполните следующие команды:

```
tar -xf vim.tar
```

```
ls etc/vim
```

Чтобы извлечь файлы из vim.tar в другой каталог, используйте параметр -C команды tar. Например, выполните следующие команды:

```
tar -xvf vim.tar -C / tmp
```

```
ls -R / tmp / etc
```

Команда **cpio** - это другая архивная команда, которая может объединять несколько файлов в один файл. Он работает в двух режимах: **cru-in** и **cru-out**. Режим **cru-out** для создания нового файла архива. Файлы могут быть предоставлены через стандартный ввод или перенаправлены с выхода другой команды для создания потока файлов, который будет заархивирован. Чтобы заархивировать все файлы *.conf в текущем каталоге, используйте параметр -o для команды cpio (-v для "verbose"):

```
cp /etc/*.conf .
```

```
ls *.conf | cpio -ov > conf.cpio
```

```
ls *.cpio
```

Режим **cru-in** используется для извлечения файлов из архивного файла cpio. Опция **-i** включает «режим копирования». Параметр **-u** используется для перезаписи существующих файлов, а параметр **-d** указывает, что каталоги должны быть созданы. Чтобы извлечь файлы из файла **conf.cpio** в текущий каталог, сначала удалите исходные файлы, а затем с помощью команды **cat** отправьте данные в команду **cpio**:

```
ls
```

```
rm *.conf
```

```
ls
```

```
cat conf.cpio | cpio -iud
```

```
ls
```

```
root@L-FW:~# cat conf.cpio | cpio -iud
67 blocks
root@L-FW:~# ls
adduser.conf          delzone1             kernel-img.conf      my.sh                rsyslog.conf
archive.cpio          discover-modprobe.conf ld.so.conf           nsswitch.conf       sestatus.conf
ca-certificates.conf example              libaudit.conf        pam.conf             sysctl.conf
conf.cpio             example.tar.xz       logrotate.conf       passwd              ucf.conf
cpio                  gai.conf             lpicfile.txt         passwd.cpio          words
debconf.conf          hdparm.conf          messages             personal_backup     words.gz
deluser.conf          host.conf            messages.gz          reportbug.conf
delzone              ipbase              mke2fs.conf         resolv.conf
```

Чтобы создать файл с именем /tmp/swapex с 500 блоками нулей размером «один мегабайт», выполните следующую команду:

```
dd if=/dev/zero of=/tmp/swapex bs=1M count=500
```

```
ls -lh /tmp
```

```
total 500M
drwxrwxr-x 1 sysadmin sysadmin 6 Oct 27 19:05 etc
-rw-rw-r-- 1 sysadmin sysadmin 500M Oct 27 19:20 swapex
```

Команда **dd** - это утилита для копирования файлов на битовом уровне. Его можно использовать для клонирования или удаления целых дисков или разделов, создания больших «пустых» файлов для использования в качестве файлов подкачки (как в примере выше) и копирования необработанных данных на съемные устройства.