Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики

Лабораторная работа №5
по операционной системе

Имя: Шубхам Кунал
Группа: К33401
Преподаватель: Ватьян Александра Сергеевна

Санкт-Петербург
2020

Задание 1

1. Объясните “странные” результаты сжатия файлов floppyZero и floppyRandom.

Решение

kunal@kunal-desktop: ~/shells × kunal@kunal-desktop: ~/shells

```
# Create two files with zeros and random numbers
dd if=/dev/zero of=floppyZero.img bs=1k count=1440
dd if=/dev/urandom of=floppyRandom.img bs=1k count=1440
du -h floppyZero.img floppyRandom.img
#1.5M floppyZero.img
#1.5M floppyRandom.img

printf "\n\ngzip:\n"
# Using gzip/bzip2 for compress files
gzip -v floppyZero.img # replace gzip -> bzip2
#floppyZero.img: 99.9% -- replaced with floppyZero.img.gz
du -h floppyZero.img.gz
#4.0K floppyZero.img.gz
gunzip floppyZero.img.gz # replace gunzip -> bunzip
gzip -v floppyRandom.img # replace gzip -> bzip2
du -h floppyRandom.img.gz
gunzip floppyRandom.img.gz

printf "\n\nbzip2:\n"
#floppyRandom.img: -0.0% -- replaced with floppyRandom.img.gz
bzip2 -v floppyZero.img
du -h floppyZero.img.bz2
bunzip2 floppyZero.img.bz2
bzip2 -v floppyRandom.img
du -h floppyRandom.img.bz2
#1.5M floppyRandom.img.gz
bunzip2 floppyRandom.img.bz2 # replace gunzip -> bunzip
```

```
kunal@kunal-desktop:~/shells$ ./task_5_1.sh
1440+0 records in
1440+0 records out
1474560 bytes (1.5 MB, 1.4 MiB) copied, 0.0144154 s, 102 MB/s
1440+0 records in
1440+0 records out
1474560 bytes (1.5 MB, 1.4 MiB) copied, 0.0237495 s, 62.1 MB/s
1.5M    floppyZero.img
1.5M    floppyRandom.img


gzip:
floppyZero.img:  99.9% -- replaced with floppyZero.img.gz
4.0K    floppyZero.img.gz
floppyRandom.img:        -0.0% -- replaced with floppyRandom.img.gz
1.5M    floppyRandom.img.gz


bzip2:
  floppyZero.img: 30093.061:1,  0.000 bits/byte, 100.00% saved, 1474560 in, 49 out.
4.0K    floppyZero.img.bz2
  floppyRandom.img:  0.995:1,  8.038 bits/byte, -0.47% saved, 1474560 in, 1481556 out.
1.5M    floppyRandom.img.bz2
kunal@kunal-desktop:~/shells$
```

These algorithms do an excellent job with a sequence of identical characters but are practically useless in the case when there are not many repeating sequences. This is because it is difficult to find duplicate lines in a random sequence, which just allow you to encode them and reduce the file size.

2. Покажите разницу в степени сжатия между утилитами gzip с уровнями сжатия 1, 6 и 9 (степень сжатия задается опцией -N, где N число от 1 до 9, указывающее степень сжатия) и bzip2 на примере файла, созданного командой:
   man man > heManFile

Решение

```
name="heManFile"

gzip -1 -v $name
du -h $name.gz
gunzip $name.gz

gzip -6 -v $name
du -h $name.gz
gunzip $name.gz

gzip -9 -v $name
du -h $name.gz
gunzip $name.gz

bzip2 -v $name
du -h $name.bz2
bunzip2 $name.bz2
```

```
kunal@kunal-desktop:~/shells$ sudo ./task_5_2.sh
[sudo] password for kunal:
heManFile:       61.1% -- replaced with heManFile.gz
16K     heManFile.gz
heManFile:       66.2% -- replaced with heManFile.gz
12K     heManFile.gz
heManFile:       66.2% -- replaced with heManFile.gz
12K     heManFile.gz
  heManFile:  3.253:1,  2.460 bits/byte, 69.25% saved, 34851 in, 10715 out.
12K     heManFile.bz2
kunal@kunal-desktop:~/shells$
```

3. Попробуйте сжать картинку в формате png (jpeg) утилитой gzip, используя различные уровни сжатия от 1 до 9. На сколько процентов от исходного размера был сжат файл? Сильно ли отличается степень сжатия между уровнями? Почему?

Решение

```
name="img.jpg"

for i in {1..9}
do
        gzip -"$i" -v $name
        du -h $name.gz
        gunzip $name.gz
done
~
```

```
kunal@kunal-desktop:~/shells$ ./task_5_3.sh
img.jpg:         3.7% -- replaced with img.jpg.gz
412K    img.jpg.gz
img.jpg:         3.7% -- replaced with img.jpg.gz
412K    img.jpg.gz
img.jpg:         3.8% -- replaced with img.jpg.gz
412K    img.jpg.gz
img.jpg:         3.4% -- replaced with img.jpg.gz
412K    img.jpg.gz
img.jpg:         3.4% -- replaced with img.jpg.gz
412K    img.jpg.gz
img.jpg:         3.4% -- replaced with img.jpg.gz
412K    img.jpg.gz
img.jpg:         3.4% -- replaced with img.jpg.gz
412K    img.jpg.gz
img.jpg:         3.4% -- replaced with img.jpg.gz
412K    img.jpg.gz
img.jpg:         3.4% -- replaced with img.jpg.gz
412K    img.jpg.gz
kunal@kunal-desktop:~/shells$
```

The file was compressed quite a bit, the new size is about 99% in all cases. This inefficient compression occurs because the JPG / PNG formats are already compressed files and have practically no redundant information.

Задание2:

Как было сказано, tar не умеет самостоятельно сжимать данные, но tar можно сообщить о том, что для сжатия данных следует использовать одну из утилит, например, gzip/bzip2. Замените опции при создании/распаковке архива -cf/-xf на -czf/-xzf (для использования gzip) или на -cjz/-xjf (для использования bzip2), после чего сравните размеры архивов с сжатием и без сжатия данных

```
mkdir sandbox && cd $_ # change directory to sandbox
touch file{1..20} tmp #this files..
mkdir images media #and the dirs we want add to archive
cd ..

#create gzip archive
tar -czf archive_gzip.tar sandbox
tar -cjf archive_bzip.tar sandbox

#Delete sandbox folder
rm -rf sandbox

#Archive sizes
du -h archive_gzip.tar
du -h archive_bzip.tar

#Extract data from archs
tar -xzf archive_gzip.tar -C files_gzip
tar -xjf archive_bzip.tar -C files_bzip
~
~
```

```
kunal@kunal-desktop:~/shells$ ./task_5_4.sh
4.0K    archive_gzip.tar
4.0K    archive_bzip.tar
kunal@kunal-desktop:~/shells$
```

Задание3:

Напишите bash сценарий для создания инкрементных бэкапов. Воспользуйтесь теми же командами, что приведены для разностных бэкапов (Hint: Обратите внимание на шаг 2).

Terminal - kunal@kunal-deskto

File Edit View Terminal Tabs Help

kunal@kunal-desktop: ~/shells | kunal@kunal-desktop: ~/shells | ku

```
# to_backup/  - files to backup
# backup/  - backups
# restored/  - restored files

# Create files for backup
mkdir to_backup
touch to_backup/file{1..3}

# Create folder for backup
mkdir backup 2> /dev/null
cd backup

# Create backup
tar -cpvzf full_backup.tar.gz.0 -g backup.snap ../to_backup
cp backup.snap backup.snap.1
touch ../to_backup/file4
tar -cpvzf diff_backup.tar.gz.1 -g backup.snap.1 ../to_backup
cat backup.snap.1

for i in {1..3}
do
    cp "backup.snap.$i" "backup.snap.$((i+1))"
    touch "../to_backup/file$((i+4))"
    tar -cpvzf "diff_backup.tar.gz.$((i+1))" -g "backup.snap.$((i+1))" ../to_backup
    cat "backup.snap.$((i+1))"
done

cd ..
mkdir restored
tar -xvf backup/full_backup.tar.gz.0 -G -C restored

for i in {1..4}
do
    tar -xvf backup/diff_backup.tar.gz.$i -G -C restored
done
```

```
kunal@kunal-desktop:~/shells$ ./task_5_5.sh
tar: ../to_backup: Directory is new
tar: Removing leading `../' from member names
../to_backup/
../to_backup/file1
../to_backup/file2
../to_backup/file3
tar: Removing leading `../' from member names
../to_backup/
../to_backup/file4
GNU tar-1.29-2
160886076681604895301608860766812904691205353768641../to_backupNfile1Nfile2Nfile3Yfile4tar: Removing leading `../' from member names
../to_backup/
../to_backup/file5
GNU tar-1.29-2
1608860766835479424016088607668329045902053537686 4../to_backupNfile1Nfile2Nfile3Nfile4Yfile5tar: Removing leading `../' from member names
../to_backup/
../to_backup/file6
GNU tar-1.29-2
160886076684409384801608860766840904550205353768641../to_backupNfile1Nfile2Nfile3Nfile4Nfile5Yfile6tar: Removing leading `../' from member names
../to_backup/
../to_backup/file7
GNU tar-1.29-2
160886076685435026501608860766848904510205353768641../to_backupNfile1Nfile2Nfile3Nfile4Nfile5Nfile6Yfile7to_backup/
to_backup/file1
to_backup/file2
to_backup/file3
to_backup/
to_backup/file4
to_backup/
to_backup/file5
to_backup/
to_backup/file6
to_backup/
to_backup/file7
kunal@kunal-desktop:~/shells$ ls
```

Задание4:

1. Для разового выполнения заданий существует команда at. Используя эту команду, запланируйте разовое выполнение полного бэкапа

   tar -cpvzf full_backup.tar.gz for_backup | at 20:30

2. Используя cron и сценарий из предыдущей части, запланируйте выполнение разностных или инкрементных бэкапов.

export EDITOR=vi

crontab -e

In file: m h D M WD

09 10 1 * * ~/shells/task_5_5.sh