Module 09
Archiving and Compression

Exam Objective

3.1 Archiving Files on the Command Line

Objective Description

Archiving files in the user home directory

Introduction

Introduction

- In this chapter, we discuss how to manage archive files at the command line.
- *File archiving* is used when one or more files need to be transmitted or stored as efficiently as possible.
- There are two fundamental aspects which this chapter explores:
    - *Archiving*: Combines multiple files into one, which eliminates the overhead in individual files and makes it easier to transmit.
    - *Compression*: Makes the files smaller by removing redundant information.

Compression

Compressing Files

- *Compression* reduces the amount of data needed to store or transmit a file while storing it in such a way that the file can be restored.
- The *compression algorithm* is a procedure the computer uses to encode the original file, and as a result, make it smaller.
- When talking about compression, there are two types:
  - *Lossless*: No information is removed from the file.
  - *Lossy*: Information might be removed from the file.

Compressing Files

- Linux provides several tools to compress files, the most common is `gzip`. Here we show a file before and after compression:

```
sysadmin@localhost:~/Documents$ ls -l longfile*
-rw-r--r-- 1 sysadmin sysadmin 66540 Dec 20  2017 longfile.txt
sysadmin@localhost:~/Documents$ gzip longfile.txt
sysadmin@localhost:~/Documents$ ls -l longfile*
-rw-r--r-- 1 sysadmin sysadmin 341 Dec 20  2017 longfile.txt.gz
```

  - The original size of the file called `longfile.txt` is 66540 bytes.
  - The file is compressed by invoking the `gzip` command with the name of the file as the argument.
  - After that command completes, the original file is gone, and a compressed version with a file extension of `.gz` is left in its place.
  - The file size is now 341 bytes.

Compressing Files

- The `gzip` command will provide this information, by using the `-l` option, as shown here:

```
sysadmin@localhost:~/Documents$ gzip -l longfile.txt.gz
         compressed        uncompressed  ratio uncompressed_name
                341               66540  99.5% longfile.txt
```

- Compressed files can be restored to their original form (*decompression*) using either the `gunzip` command or the `gzip -d` command.

- After `gunzip` does its work, the `longfile.txt` file is restored to its original size and file name:

```
sysadmin@localhost:~/Documents$ gunzip longfile.txt.gz
sysadmin@localhost:~/Documents$ ls -l longfile*
-rw-r--r-- 1 sysadmin sysadmin 66540 Dec 20  2017 longfile.txt
```

Archiving

Archiving Files

- *Archiving* is when you compress many files or directories into one file.
- The traditional UNIX utility to archive files is called `tar`, which is a short form of TApe aRchive.
- *Tar* has three modes that are helpful to become familiar with:
  - **Create** : Make a new archive out of a series of files.
  - **Extract** : Pull one or more files out of an archive.
  - **List** : Show the contents of the archive without extracting.

Archiving Files - Create Mode

```
tar -c [-f ARCHIVE] [OPTIONS] [FILE...]
```

- Creating an archive with the `tar` command requires two named options:

`-c`	Create an archive.
`-f ARCHIVE`	Use archive file. The argument `ARCHIVE` will be the name of the resulting archive file.

- The following example shows a *tar file*, also called a *tarball*, being created from multiple files:

```
sysadmin@localhost:~/Documents$ tar -cf alpha_files.tar alpha*
sysadmin@localhost:~/Documents$ ls -l alpha_files.tar
-rw-rw-r-- 1 sysadmin sysadmin 10240 Oct 31 17:07 alpha_files.tar
```

Archiving Files - Create Mode

- Tarballs can be compressed for easier transport, either by using `gzip` on the archive or by having `tar` do it with the `-z` option:

```
sysadmin@localhost:~/Documents$ tar -czf alpha_files.tar.gz alpha*
sysadmin@localhost:~/Documents$ ls -l alpha_files.tar.gz
-rw-rw-r-- 1 sysadmin sysadmin 417 Oct 31 17:15 alpha_files.tar.gz
```

- The `bzip2` compression can be used instead of `gzip` by substituting the `-j` option for the `-z` option and using `.tar.bz2`, `.tbz`, or `.tbz2` as the file extension:

```
sysadmin@localhost:~/Documents$ tar -cjf folders.tbz School
```

Archiving Files - List Mode

```
tar -t [-f ARCHIVE] [OPTIONS]
```

- Given a `tar` archive, compressed or not, you can see what's in it by using the `-t` option. The next example uses three options:

`-t`	List the files in the archive.
`-j`	Decompress with the `bzip2` command.
`-f ARCHIVE`	Operate on the given archive.

- The following example lists the contents of the `folders.tbz` archive:

```
sysadmin@localhost:~/Documents$ tar -tjf folders.tbz
```

Archiving Files - Extract Mode

```
tar -x [-f ARCHIVE] [OPTIONS]
```

- You can extract the archive with the `-x` option once it's copied into a different directory. The following example uses the similar pattern as with the other modes:

`-x`	Extract files from an archive.
`-j`	Decompress with the `bzip2` command.
`-f ARCHIVE`	Operate on the given archive.

- The following example extracts the contents of the `folders.tbz` archive:

```
sysadmin@localhost:~/Documents$ tar -xjf folders.tbz
```

ZIP Files

- The ZIP file is the default archiving utility in Microsoft.
- ZIP is not as prevalent in Linux but is well supported by the `zip` and `unzip` commands.
- The default mode of `zip` is to add files to an archive and compress it.

```
zip [OPTIONS] [zipfile [file…]]
```

- The following example shows a compressed archive called `alpha_files.zip` being created:

```
sysadmin@localhost:~/Documents$ zip alpha_files.zip alpha*
```

- The `zip` command will not recurse into subdirectories by default (`tar` does), so you must use the `-r` option to indicate recursion is to be used.

ZIP Files

- The `-l` *list* option of the `unzip` command lists files in `.zip` archives:

```
sysadmin@localhost:~/Documents$ unzip -l School.zip
Archive:  School.zip
  Length      Date    Time    Name
---------  ---------- -----   ----
        0  2017-12-20 16:46   School/
        0  2018-10-31 17:47   School/Engineering/
```

- Just like `tar`, you can pass filenames on the command line.