05 – Archiving

CS 2043: Unix Tools and Scripting, Spring 2019 [1]

Matthew Milano
February 1, 2019

Cornell University

Table of Contents

File Compression

Making Archives: Zip

Package and Compress (Archive) Files

`zip <name_of_archive> <files_to_include>`
- E.g. `zip files.zip a.txt b.txt c.txt`
- Extracts to `a.txt`, `b.txt`, and `c.txt` in *current directory*.
- To do folders, you need recursion.
    - `zip -r folder.zip my_files/`
    - Extracts to folder named `my_files` in *current directory*.
    - Good practice to ALWAYS zip a folder and distribute with the name it will extract as.
        - `zip -r folder_name.zip folder_name/`
    - Drives me *crazy* when I get a `.zip` that extracts files in the same directory… very difficult to keep track of.

List, Test and Extract Compressed Files in a zip Archive

`unzip <archive_name>`
- Use `-l` to list what would extract before doing it.

· **Note**: The original files DO stay intact.

Making Archives: Gzip

GNU zip

`gzip <files_to_compress>`

- Less time to compress, larger file: `--fast`
- More time to compress, smaller file: `--best`
- Read the **man** page, lots of options.
- By default, *replaces* the original files!

    - You can use `--keep` to bypass this.

GNU unzip

`gunzip <archive_name>`

- Use `-l` to list what would extract before doing it.

- **Notes**:
    - Does not bundle the files.
    - Reiterate: *replaces original* by default.
    - Usually has better compression than `zip`.

Additional Archive Formats

- This is a non-exhaustive list. There are **many** out there.
- Similar interface to `gzip`:
    - `bzip2`: "Burrows-Wheeler block sorting compression algorithm"
    - `xz`: "x"-zip, uses LZMA compression scheme (good)
- Honorable mentions:
    - `file.rar`: a "RAR" archive; used for distributing large files
        - `file.rar.001`, `file.rar.002`, etc: multiple archives needed to reconstruct whole.
        - You extract the first one, it looks for the others in same directory.
    - `file.7z`: "7"-zip, successor to RAR, uses LZMA
        - If you are choosing between `.rar` and `.7z`…choose `.7z`.
    - Install `unrar` to deal with these on Unix.
- **Moral**:
    - Working with `tar` and/or only Unix? **Use `xz`**.
    - Have to support Windows fools? **Use `7zip`**.

Making Archives: Tar

- Bundling files together to compress is easy!

Tape archive

`tar -cf <tar_archive_name> <files_to_compress>`

- Create a tar archive.

`tar -xf <tar_archive_name>`

- Extract all files from archive.

- `tar` is a stream tool. By default, it is expecting stream input.
- Don't forget the `-f` if you are working with files!

- Notes:
    - `tar` is just a bundling suite, creating a single file.
    - By default, it does *not* compress.
    - Original files DO stay intact.
    - Unlike `zip`, you do not need the `-r` flag for folders :)

Making tarballs

```
tar -c[zjJ]f <archive_name> <source_files>
tar -x[zjJ]f <archive_name>
```

- [zjJ] here means *either* z, j, *or* J — only one.
- YOU have to specify the file extension.
- Use gzip compression method: -z (or --gzip)
    - Extension convention: .tar.gz
    - Example: tar -czf files.tar.gz files/
- Use bzip2 compression method: -j (or --bzip2)
    - Extension convention: .tar.bz2
    - Example: tar -cjf files.tar.bz2 files/
- Use xz compression method: -J (or --xz)
    - Extension convention: .tar.xz
    - Example: tar -cJf files.tar.xz files/

- **Extraction** can *usually* happen automatically:
    - `tar -xf files.tar.gz` will usually work (no `-z`)
    - Best results when:
        - You are obeying filename conventions.
        - `tar` made the archive in the first place.

- **Compression**: no, you have to tell it what to do…
- **SUDDEN REMINDER** to obsessively hit your `tab` key ;)

References

[1] Stephen McDowell, Bruno Abrahao, Hussam Abu-Libdeh, Nicolas Savva, David Slater, and others over the years. "Previous Cornell CS 2043 Course Slides".