

## Solaris/Linux Utilities

### *Exercise*

- Display the file *datafile\_part*, highlighting Sunday logins by changing 'Sun' to 'SUNDAY'
- Repeat the exercise, now keeping the output in neat columns (your first answer probably offset some text)
- Display the file *datafile\_part*, eliminating the following:
  - \* Where logged in from (e.g. tty2)
  - \* The day of the week
  - \* The colon character in the login and logout times
- Display the file *datafile\_part*, replacing all colon characters with spaces, and printing all letters in upper case.
- **If you finish quickly:-**
- Combine Unix utilities to report on the number of logins reported on each day of the week in the file *datafile-part*.
  - \* Your final report should be 7 lines long, and you will NOT need to list the name of each day of the week!

## Solaris/Linux Utilities

### Compression Utilities - compress

- There are several compression utilities, including some which can also archive files, i.e. compress and save multiple files within a single compressed file.
- It is also common to use *tar* to archive file/directories to a single file, then compress the tar file.
- Some typical utilities are:-
- **compress** (*uncompress*, *zcat*) - the original UNIX compress utility, largely superceded by other utilities which are more efficient. Uses Lempel-Ziv-Welch (LZW) encoding (<http://en.wikipedia.org/wiki/Lzw>)

```
$ ls -l a.txt
```

```
-rw-r--r--  1 sal  2001  188594940 Dec 25 21:18 a.txt
```

```
*      a.txt is around 180Mbytes in size.
```

```
*      To compress a.txt and measure how long it takes:-
```

```
$ time compress a.txt
```

```
real  0m23.702s
```

```
user  0m22.510s
```

```
sys   0m0.350s
```

```
$ ls -l a.txt.Z
```

```
-rw-r--r--  1 sal  2001  80739857 Dec 25 21:18 a.txt.Z
```

```
*      a.txt.Z is the compressed filename. a.txt has gone.
```

```
*      a.txt.Z is about 78 Mbytes, 43% of the original file size.
```

## Solaris/Linux Utilities

### Compression Utilities

- To uncompress the file *a.txt.Z* back to its original state, and measure how long it takes:-

```
$ time uncompress a.txt.Z
```

```
real 0m10.247s
```

```
user 0m9.311s
```

```
sys 0m0.308s
```

```
$ ls -l a.txt
```

```
-rw-r--r-- 1 sal 2001 188594940 Dec 25 21:18 a.txt
```

- To compress the file and write to a specified file, leaving the original file in place:-

```
$ compress -c a.txt > newa.txt.Z
```

\* The *-c* option tells *compress* to write to *stdout*. If you try this, please make sure you redirect the output!

- To display the contents of a compressed file use *zcat*. This is actually the same as *uncompress -c*.

## Solaris/Linux Utilities

### Compression Utilities - zip/unzip

- **zip** (*unzip*) - a very common utility found on many platforms. *zip* archives as well as compresses, so you can use it to compress and archive a directory structure into a single file.

```
$ time zip arch1 a.txt
```

```
adding: a.txt (deflated 67%)
```

```
real 0m49.548s
```

```
user 0m43.780s
```

```
sys 0m0.767s
```

```
% ls -l
```

```
total 490900
```

```
-rw-r--r-- 1 sa1 2001 188594940 Dec 25 21:18 a.txt
```

```
-rw-r--r-- 1 sa1 2001 62416684 Dec 26 08:09 arch1.zip
```

```
* zip has created the arch1.zip compressed archive file containing the single file a.txt. The original a.txt file is still in place.
```

```
* zip has taken over twice as long as compress, but the compressed archive is only 33% of the size of the original a.txt file.
```

- To extract and uncompress from the *arch1.zip* archive:-

```
% time unzip arch1.zip
```

```
Archive: arch1.zip
```

```
inflating: a.txt
```

```
real 0m11.252s
```

```
user 0m5.889s
```

```
sys 0m0.880s
```

## Solaris/Linux Utilities

### Compression Utilities - zip/unzip

- *zip* can handle mutiple files, and directories:-

```
$ cd ~/utilities
```

```
$ zip arch1 *
```

```
adding: Christian (stored 0%)
```

```
adding: datafile_full (deflated 83%)
```

```
adding: datafile_part (deflated 81%)
```

```
etc..
```

```
$ ls -l arch1.zip
```

```
-rw-r--r--  1 sal   2001   287154 Dec 26 08:28 arch1.zip
```

```
$ du -k
```

```
1551  .
```

- To *zip* a directory:-

```
$ cd
```

```
$ zip -r arch1 utilities
```

```
adding: utilities/ (stored 0%)
```

```
adding: utilities/vi.tips (deflated 54%)
```

```
adding: utilities/arch1.zip (stored 0%)
```

```
adding: utilities/ex_data2 (deflated 77%)
```

```
adding: utilities/Christian (stored 0%)
```

```
adding: utilities/internet_addresses (deflated 71%)
```

```
etc...
```

## Solaris/Linux Utilities

### Compression Utilities - zip/unzip

- Use *unzip* to extract/uncompress files from an archive; here are some examples:-

**\$ *unzip -l arch1***

Archive: arch1.zip

etc..

- \* The above command lists the archive contents; the use of the *.zip* extension is optional when specifying the archive file name.

**\$ *unzip arch1***

- \* Extract and uncompress all files in *arch1.zip*

**\$ *unzip arch1 -d /var/tmp***

- \* Extract and uncompress all files in *arch1.zip* to the directory */var/tmp*

**\$ *unzip arch1 utilities/stdcode***

- \* Extract and uncompress *stdcode* in *arch1.zip*

**\$ *unzip -c arch1 utilities/stdcode***

- \* Extract and uncompress *stdcode* in *arch1.zip* to *stdout* (i.e. displays the contents of *stdcode*)

- The *zip* and *unzip* man pages have many more useful examples.

## Solaris/Linux Utilities

### Compression Utilities - gzip/gunzip/gzcat

- **gzip** (*gunzip*, *gzcat*) - similar to *zip*, but does not archive, so it is more convenient for compressing individual files.
- The compression time and efficiency are the same as *zip*.

\$ **gzip** *a.txt*

\* Compress *a.txt*, replacing it with *a.txt.gz*

\$ **gzip -9** *a.txt*

\* As above but works harder to achieve (slightly) better compression. (1 is fast, 9 is slow but with maximum compression - default is 6)

- To decompress, use *gzip -d* or *gunzip*.

\$ **gunzip -l** *a.txt.gz*

compressed	uncompressed	ratio	uncompressed_name
62203991	188594940	67.0%	a.txt

\* Lists the properties of the compressed file.

\$ **gunzip** *a.txt.gz*

\* Uncompress *a.txt.gz* to *a.txt*

## Solaris/Linux Utilities

### Compression Utilities - bzip2/bunzip2

- **bzip2** (*bunzip2*, *bzcat*) -Similar in operation to *gzip*, *bzip2* achieves very efficient compression, but at the expense of high CPU usage. (4 to 5 times slower than *gzip*, but quick to uncompress)
- For example, it would reduce the size of the *a.txt* file example (180 Mbytes) to 13 Mbytes.

\$ ***bzip2 a.txt***

\* Compress *a.txt* to *a.txt.bz2*

\$ ***bunzip2 a.txt.bz2***

\* Uncompress *a.txt.bz2* to *a.txt*

- Note for Solaris users: Before you go compressing all your enormous log files using these utilities, bear in mind that the new ZFS file system in Solaris can automatically compress files when they are written - they can then be viewed and manipulated just as though they were ordinary files.

## Solaris/Linux Utilities

### The conversion utility `dd`

- `dd` copies and converts files, according to the specified criteria; conversion includes such things as EBCDIC to ASCII.
- `dd` can read and write data in files and devices, such as tape and disc.
  - \* If it's physically possible to read a tape on your drive, than `dd` can do it. Whether you can work out the format of the data might be quite another matter ...
  - \* See the next command (`od`) if you want to examine your data byte by byte.
- At the end of its run, `dd` reports the number of input and output records.

\$ `dd if=myprog.f of=vax_prog.for conv=ucase`

- \* This example copies `myprog.f` to `vax_prog.for`, converting all characters to upper case.
- Admittedly, `dd` is somewhat old-fashioned, and designed in the days when transfer of data by 1/2" tape was common.
- However, it has unique features which may, on those rare occasions, prove a life-saver!