

INSTRUCTIONS

Adapted from Dan Kinnamon, Ph.D. (University of Miami HGG630)

The purpose of these exercises is to give you experience solving intermediate-level problems with file system maintenance that anyone using Linux is likely to encounter regularly. Major themes of these exercises are *identifying the right Linux utility* and *using several Linux utilities together in a single command line* to achieve the desired result.

Provide the command(s) that you used and the output along with your answers. Except where noted, ***all of these problems could be solved with a single command line (possibly using multiple utilities)***. If you are unable to find a one-line solution, oh well.

And remember to GTS!!!!

EXERCISES

- 1) **Download files.** Start by executing the following command in your home directory (**bold text should be replaced**):

```
wget -nv -r --no-host-directories --ftp-user=anonymous --ftp-  
password=your-email --preserve-permissions  
ftp.ncbi.nih.gov:blast/documents
```

This will download a directory tree of BLAST documentation that you will use for many of the exercises that follow. Redirect the STDERR and STDOUT of this command to a single file and examine it to make sure the download went correctly. Paste the first 7 lines of that file in your answers.

- 1) **Examine the directory tree.**
 - a. Report the number of each of the following in the directory tree:
 - i. Entries of any type
 - ii. Directories
 - iii. Regular files
 - iv. Text (*.txt) files
 - v. Regular files with *all* write bits unset
 - vi. Directory files with *any* write bit set
 - vii. Files modified within the last year
 - b. List the contents of all directories, including their permissions, file sizes, and modification times. Sort the list by modification time, from most recent to oldest. Also, make sure to display file sizes in human-readable format (e.g., K, M, G).
 - c. Calculate the total size of the directory tree in K.
- 2) **Examine and modify permissions.**
 - a. Reset the permissions for all directories to r-xr-xr-x *while leaving all other file permissions unchanged*. What do these permissions prevent owners from doing? What couldn't the owner do with r--r--r-- permissions on a directory?
 - b. Verify that all directories in the tree have the permissions desired in part (a) and all other regular files still have r--r--r-- permissions. Show your output.
 - c. Give all *.txt files rw-r--r-- permissions *while leaving all other file permissions unchanged*. What can the owner, owning group, and others do with these

- permissions? Can the owner delete this file after the changes you made in part (a)?
- d. Verify that *.txt files have the permissions desired in part (c) and all other regular files still have r--r--r-- permissions. Show your output.
 - e. **BONUS:** What are the following permission bits in symbolic form: 755, 315, 621, 414?
- 3) **Create and maintain links.** The “developer” and “xml” subdirectories are buried deep in the directory tree, so you might want to use links to make access to these subdirectories a bit easier.
- a. If you wanted direct access to these directories from your home directory, what type of link would you use?
 - b. Create these links and produce output showing that you have done so successfully.
 - c. Give yourself write permissions on the “xml” subdirectory and rename it “foo”. What happens to your link?
 - d. Remove the links you created and produce output showing that you have done so successfully.
 - e. **BONUS:** What is the number next to the permissions in the `ls -l` output? Why is it 4 for the “documents” subdirectory?
- 4) **Archiving a directory tree for storage.**
- a. Use a utility to print a simple list of “blast” directory tree contents to a flat file in your home directory.
 - b. Use a utility to create a gzipped archive of the “blast” directory tree in your home directory.
 - c. Use a utility to compare the list of archive contents to those in your log file. You may use *two* command lines for this exercise. Provide and interpret the output.
 - d. **BONUS:** Do part (c) with a single command line. You’ll need to look up “process substitution” in BASH.

READING

Start emacs from your terminal command prompt and take the interactive tutorial (Help > Emacs Tutorial using GUI menus, C-h t using key bindings). You can *optionally* read Chapter 7 of Sobell to supplement the tutorial.