COEN 145 : Parallel Computing

Lab 2 : Debugging, Profiling, and Compiler Tricks

Unix

- Becoming familiar with working from the command line interface (CLI) on Unix systems is critical especially if a GUI isn't offered.
- **VirtualBox**, **VMWare**, **Dual-Boot**, Portable-OS, etc

Use the following command to access usage information and optional parameters

$ **man (manual)** *command_name*

$ man man

(Shortcuts - USE THESE)

Tab : autocomplete, view files prior to seeing ls

Home/End : Front or End of line

Common and useful commands

(Navigation)

$ **cd** { . | .. | directory_name | path_to_directory }

$ **pwd** (print working directory)

$ **ls** {-alt} (list directory)

$ **tree** directory_name (list recursively)

(File Creation)

$ **{ vi | vim | touch | emacs | … }** file_name

$ **mkdir** {-p} path/to/directory | ./directory_name

$ **chmod** {(+/-)rwx}{000-777} file_name

(Environment Variables)

$ **export** ENV_VAR=0 (set)

$ **echo** $ENV_VAR (get)

(Pipes (|) and Redirects (> | >>))

$ **echo** "Hello World" { >> | > } file.txt

$ **cat** file.txt (concatenate / print)

$ **ps** (process_status) {-x} | **grep** "string_to_find"

$ **pkill** -9 pid (stop process execution)

Connecting to HPC (WAVE) and transferring files

scp (secure copy), **sftp** (secure-file-transfer-protocol), **rsync**

SSH Configuration

https://linuxize.com/post/using-the-ssh-config-file/
https://linuxize.com/post/how-to-setup-passwordless-ssh-login/

~/ ⇒ /path/to/usr/home
~/ ⇒ /Users/alex (OSX)
~/ ⇒ /home/alex (Ubunutu)
~/ ⇒ /WAVE/users/unix/awhelan (Rocky-Linux/HPC)
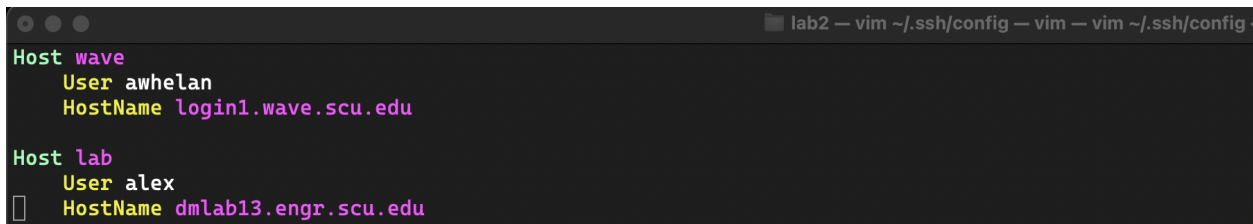
$ **mkdir** -p ~/.ssh && **chmod** 700 ~/.ssh
$ **touch** ~/.ssh/config
$ **chmod** 600 ~/.ssh/config

```
Host wave
    User awhelan
    HostName login1.wave.scu.edu

Host lab
    User alex
    HostName dmlab13.engr.scu.edu
```

Contents of ~/.ssh/config

$ **ssh** wave
$ **ssh** lab

If connecting from home make sure to set up SCU's VPN or if connecting from campus (i.e. eduroam) need to install **ClearGuard** and **Sophos.**
https://www.scu.edu/technology/get-connected/networking/how-to-access-vpn/

SSH Keybased Authentication (Passwordless Login)

(Generate Key for HostName)
$ **ssh-keygen** -t rsa -b 4096 -C **"username@scu.edu"**

Enter file in which to save the key (/home/yourusername/.ssh/id_rsa):

$ [Enter/Return]

Enter passphrase (empty for no passphrase):

$ [Enter/Return]
$ [Enter/Return]

(Copy public key from local to remote machine)
$ ssh-copy-id **"User@HostName"**

Repeat the above steps for any additional Hosts

SFTP

You can now use the **sftp** command in place of **ssh** to login to the remote machine and transfer files. The main syntax used is **get / put**

$ **sftp** wave
$ **put** {-R} current-working-dir/file-to-transfer
$ **get** {-R} current-working-dir/file-to-transfer

sftp also supports wildcard (*) replacement (i.e. globbing)

$ **get** -R *.txt

[X11 forwarding]
ssh -X, is an SSH protocol that enables users to run graphical applications (VSCODE, PyCharm, etc) on a remote server and interact with them using their local display and I/O devices.

GDB / Valgrind & Effective Debugging Techniques

"The first 90 percent of the code accounts for the first 10 percent of the development time. The remaining 10 percent of the code accounts for the other 90 percent of the development time."

—Tom Cargill, Bell Labs

- Binary Search (Execution Flow)

```
1       ...
2       ...
3       ...
4       print("Here")
5       ...
6       ...
7       ...
8       ...
```

If our program prints but still crashes we can **probably** assume the error is somewhere lower. Repeat these steps recursively to at the very least narrow down which *EXACT* line of code is causing the issue.

- GDB (GNU Debugger)

TODO

1. Fix gdb_tut.c using the tricks we learned in lab
2. Submit the Makefile along with the source code of gdb_tut.c
3. Format will be explained in lab