Unix

A strong philosophy evolved around small, modular tools that could work well together:

• Write programs to do one thing and do it well

• Write programs to work together

• Write programs to handle text streams, because that is a universal interface.

Cloud

Some AWS – EC(Elastic Compute)

Ubuntu() – t2.small –

SSH

SSH = “secure shell”

A way to access the (UNIX) command line on another computer

[Public-private key cryptography]

ssh -i COMP598DemoKey.pem ubuntu@52.86.231.158

control d – close

Unix Commands

echo: print

pwd: home dir

ls -a: show hidden files

ls -l: show details

touch: create new file

cat: print entire file

nano:

wget: get file from url

head: print first ten lines; head -n 1: print first 1 line

tail: print last ten lines

more: move down

less: move up and down

clear:

wc: word count

grep:

|

>

PATH

PS1: prompt

PYTHONPATH

adduser yingw: add new user (need sudo sometimes)

su - yingw: change user

logout: back to root user

Create New User

1. root user “ubuntu”

2. own user name

3. CLI

user: yingw

pw: 981005

ssh -i id\_rsa yingw@52.86.231.158

[enable sudo for new user]

* sudo usermod -aG sudo yingw
* groups

tmux:

Control d: exit

different panes?

vim $(fzf)

alias vzf=’vim $(fzf)’

alias czf=’cd $(dirname $(fzf))’

vim .bashsrc (otherwise, lose it when log out)

scp

* ssh is a secure communication protocol
* The “ssh” command line tool uses the ssh protocol to allow you to create a new login session on a server.
* The “scp” command line tool uses the ssh protocol to transfer files from one computer to another.
* Remember: To transfer a file, you MUST be able to log into the remote machine using SSH.

(can only use scp on local because you cannot use ssh to login in on your laptop)

vim

Vim modes

* NORMAL – navigating & giving commands to vim
* INSERT – typing in text (i)
* VISUAL – selecting things (v)
* ESC (escape key) – puts you into NORMAL mode

In NORMAL mode:

* Navigation: • hjkl - your arrow keys • $ - end of the line • 0 – start of the line • : - goto that line
* Editing text: • d<something> - delete • dd – delete line • d0 – delete to the start of the line • d$ - delete to the end of the line • d4l – delete the next(l-arrow key) 4 characters • x – delete the next character • X – delete the previous character • u - undo
* Entering INSERT mode • “i” – INSERT where the cursor is • “a” – INSERT right where the cursor is • “I” – INSERT at the first non-empty character on the line • “o” – insert a line below and INSERT at the start of it • “O” – insert a line above and INSERT at the start of it
* Search: • / - search forwards (hit “n” to go to the next match) • ? - search backwards (hit “n” to go to the next match)

In VISUAL mode:

* Navigation: hjkl
* y – copy selection
* d – delete selection

vim configuration

.vimrc

bash configuration

If you’re using the bash shell, .bashrc is the script run to configure it before it presents you with the first prompt “$” (Use bash\_profile in ssh)

* Set environment variables (alias cd598=’cd ……..’
* Write aliases for commands you use a lot
* Write bash functions for more sophisticated things you need to do
* Give you an inspirational message