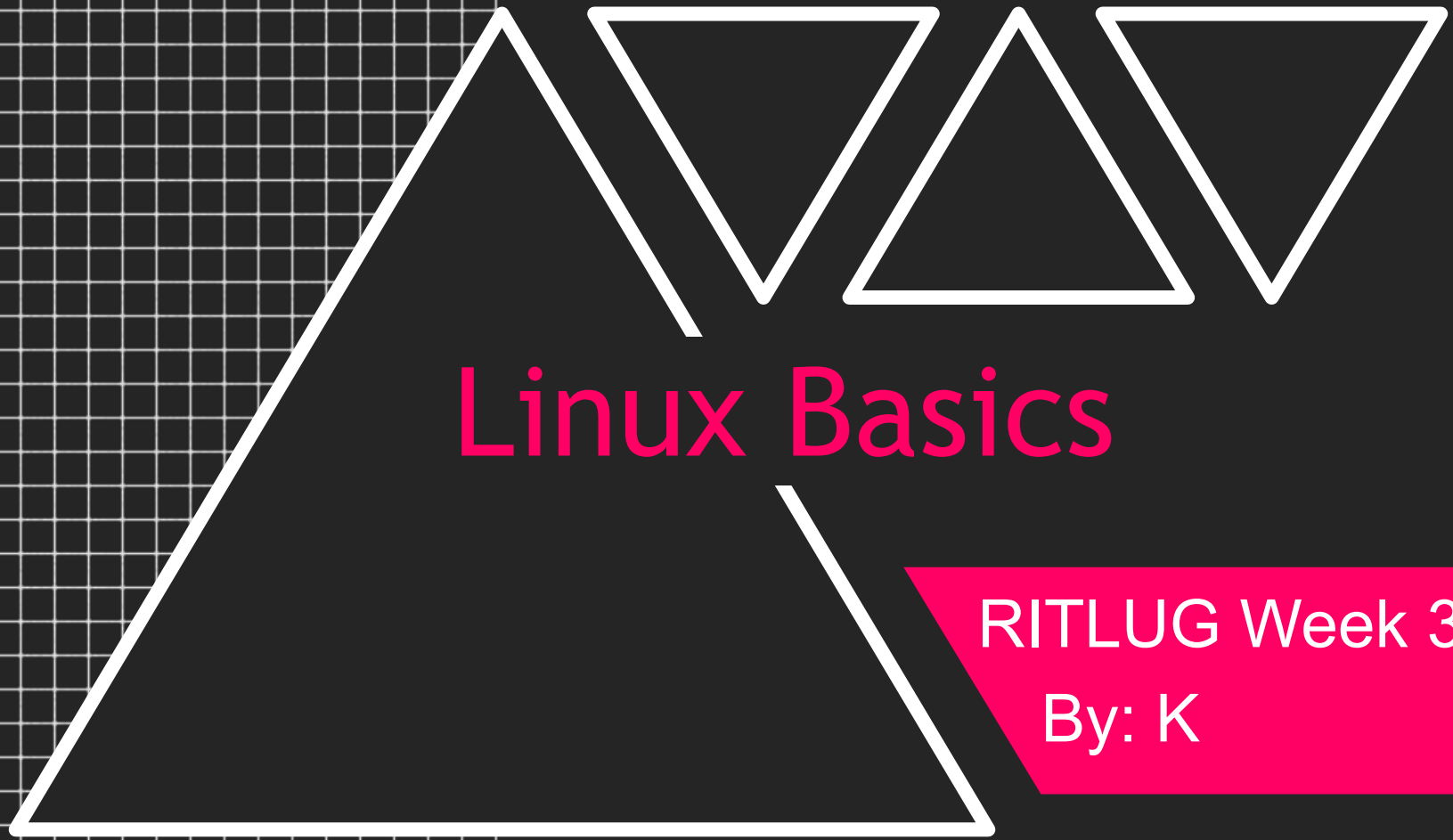
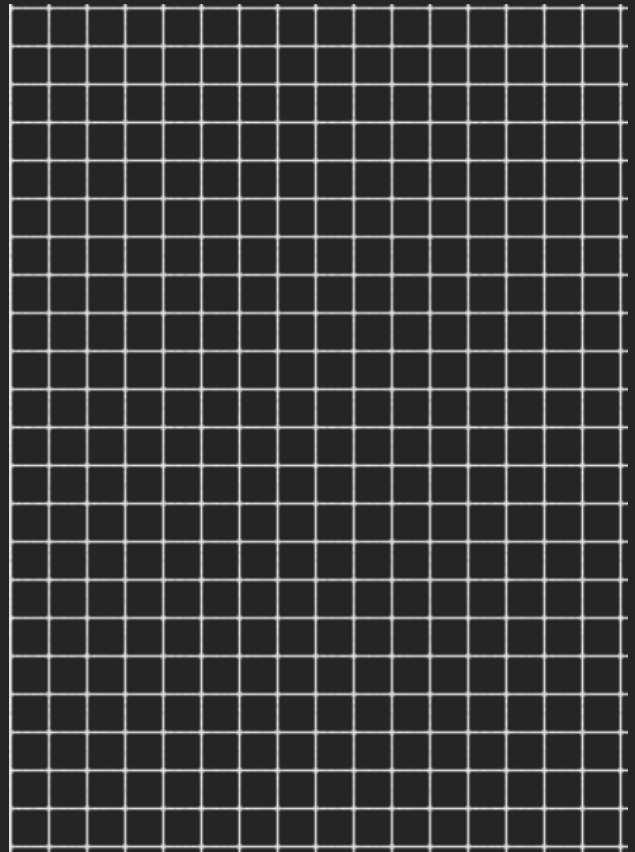


Linux Basics

RITLUG Week 3

By: K

The Command Line



- Also known as the shell, terminal, command prompt, Bash
- Many modern distros come with a graphical interface, but some, like Arch and distros made for servers come with just a command line by default.
- Bash is most common, other varieties such as zsh, fish, ksh, sh
- Powerful
- Open with ctrl+alt+t

Getting Started

```
kyri@tess-s5:~$
```

pwd - Print Working Directory. What directory are you in now?

ls - List contents of directory

clear - Clear the text from the terminal

```
kyri@tess-s5:~$ pwd
/home/kyri
kyri@tess-s5:~$ ls
Desktop    Downloads  Music  Pictures  snap       Videos
Documents  go         opt    Public    Templates
```

Navigation

cd - Change Directory

Directory shortcuts:

/ root directory

.. move up a directory

~ home directory

. current directory

- last directory you were in

Common Directories:

/home - home directories for all users

/bin - system binaries

/dev - devices connected to your system

/etc - configuration files for apps and system

/lib - system libraries and modules

/tmp - temporary files, not permanent

/var - logs, mail, printing spool

Flags

Add options to commands to do more.

Short options are usually a dash and a letter, long options are two dashes and a full word

Examples:

ls -la - common flags used with ls. l is to list in the long format, a is to show hidden files.

uname -a - lists information about your system, like hostname, distro, date and time

Files

“Everything is a file”

- Regular files: text, executables, programs, code, pictures
- Directories
- Other: sockets, named pipes, symbolic links, mounted devices

Commands:

mv - move a file or rename it

```
mv file1 ~/Documents
mv file1 file2
```

cp - copy a file to another directory

rm - remove a file

rm -r - remove a directory

touch - create a new, blank file

cat - print the contents of a file to the shell

Text Editors

Used for creating and editing files.

- vi, vim, nvim - vim is the improved version of vi. A bit of a learning curve, but worth it. Neovim allows much more customization, and you can turn vim into an IDE.
- nano - an easier to use text editor. Tells you the key bindings so you know how to do things.
- gedit - a GUI editor
- emacs - can be both command line or graphical. Also has a learning curve. Can do things beyond editing text, like email and games

Redirecting Output

Some commands will print output right to the terminal, but what if you want it to go into a file?

> - redirect output (overwrite)

>> - redirect output (append)

<,<< - redirect input

| - the output of one command used as input for the next

more - see a file page by page
less - more, but better
echo - print a string to standard output
tail - see the last 10 lines of a file
head - see the first 10 lines of a file

Examples:

```
echo “hi” > file.txt
echo “line 2” >> file.txt

less file
```

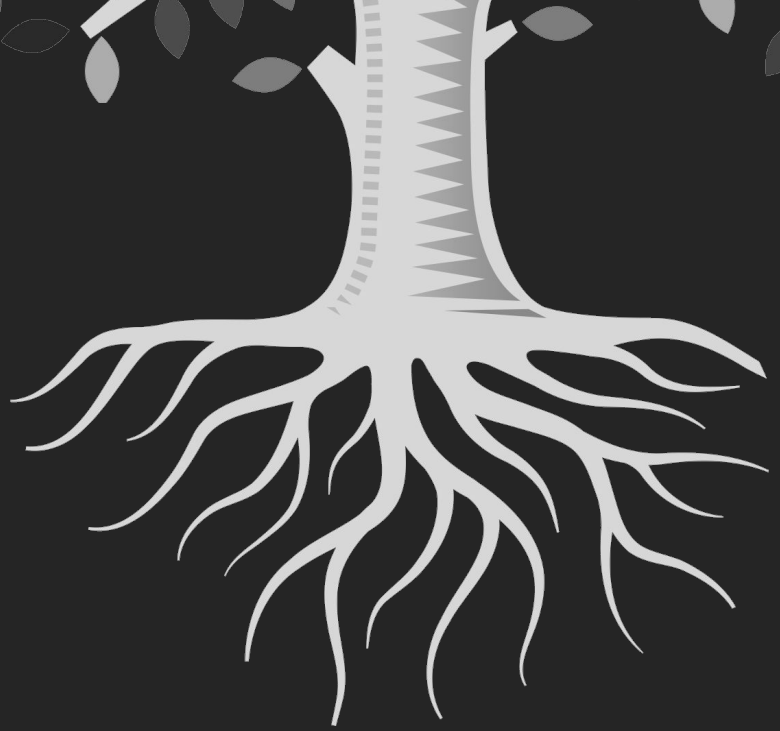
or

```
cat file | less
```

Root

Superuser account on Linux

This is the most powerful account, root can do anything, including delete the whole system



sudo - run one command with root privileges

su - switch user, default will switch to root

Because root is so powerful, it is a best practice to use sudo only when you need it and to not log in as root

Searching and Finding

grep - Global Regular Expression Print. Used to search for a specific string or regular expression in a file and print out matching lines.

ps - show the current running processes

```
ps | grep bash
```

```
tail /var/log/auth.log | grep cron
```

find - search for a file or directory. Can search by filename, type, owner, permissions, and more

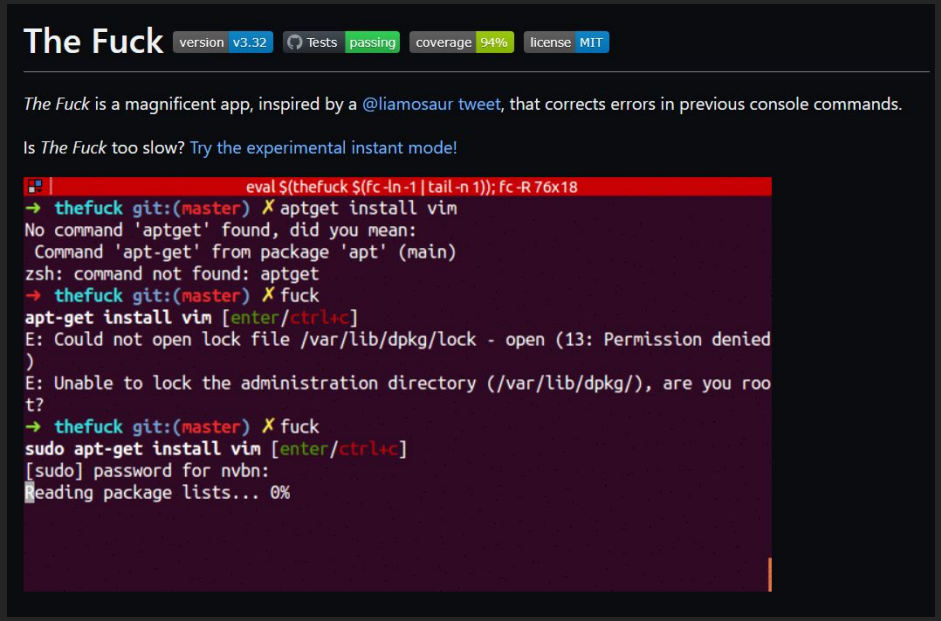
* a wildcard character, can be used in lots of commands

```
find . -name "*.pdf"
```

```
find ~ -empty
```

Resources

man - Manual pages. Gives information, options, examples for every command.

Look up examples for commands on google.

Man pages are also on the internet.