Week 01
Terminal Commands

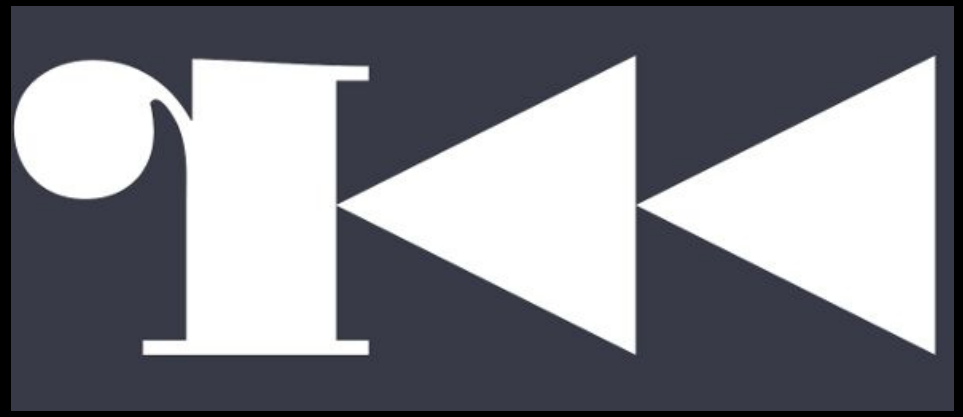
Slides By Thomas Quig & Anusha Ghosh on 8/29/21

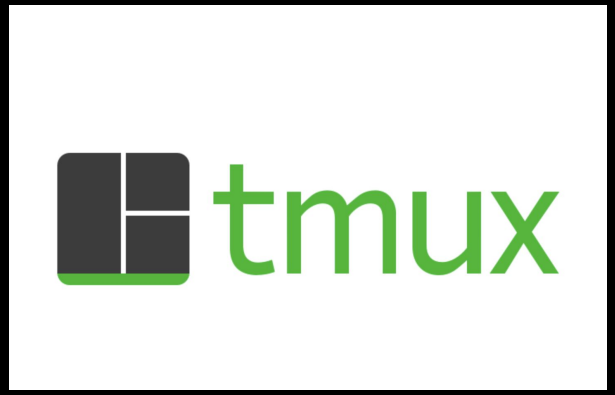
Announcements

- Scoreboard reset

- Fall recruitment event (SEP 25)

- CSAW !!!





And other valuable commands you will use often.

Table of Contents

- Relevant commands :)
- File Commands
  - find
  - grep
  - strings
  - file
- Process Commands
  - gdb
  - r2
- “The other Stuff”
  - git
  - vim
  - tmux

THIS IS THE PART WHEN WE PUT THE FLAG ON THE BOARD

sigpwny{wh@t_1n_th3_shell_is_g0ing_oN!?}

This is here so we don't forget

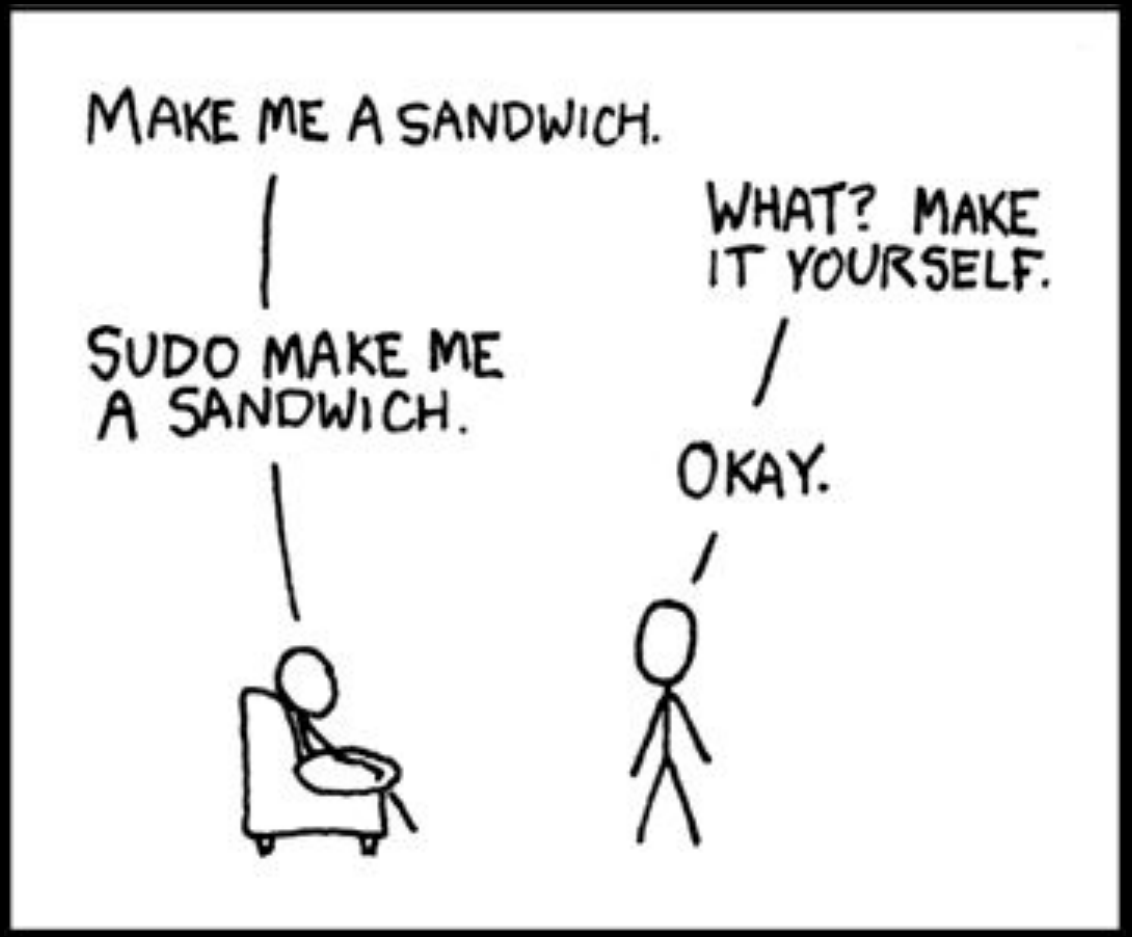
This is not a tutorial on how to get into terminal.

If you are still not sure, that's okay! Go to *sigpwny.com/tutorial* and get set up :)

File Commands

Relevant challenges: “the-library”

sudo



Give yourself more permission

very useful for certain commands

find

Find files! Most often by filename

```
find [directory] -name 'spaghetti'
```

```
find [directory] -name '*.ext'
```

grep

Find text within files!

```
grep -r “text you want to find” .
```

```
grep -A -B -C -r “text you want to find”
```

```
cat | grep
```

`strings`

see what strings exist in a file

really good initial command for RE/PWN

Why does this command exist???

Some files are not human readable, this prints out all the human readable things

There may or may not be a useful challenge for this one.

`file`

Determine the file type of a file (what it really is)

Valuable to know what you are looking at before you start attacking/RE'ing something

Process Commands

Relevant challenges: “how2re, geedeebee”

gdb

Look at executables and slowly step through them.

We can run an entire meeting on gdb, and we will if you are interested

`gdb --args` “is a way to have command line arguments”

b = breakpoint

n = next

s = step

si = step instruction (use this one for binaries)

x = look at the stack

print = print variables

What is r2?

- R2 is a “””great””” free disassembler
  - It is also known as “Radare2”
  - Easy to pick up the basics for
  - Clunky and weird to use, but can be used within your personal terminal
- You can install on your linux environment
- A good starter disassembler

Open terminal

`r2 fileName`

Press v and then enter to activate visual mode

That looks pretty complicated… but we can make it look a lot easier to understand.

Print mode

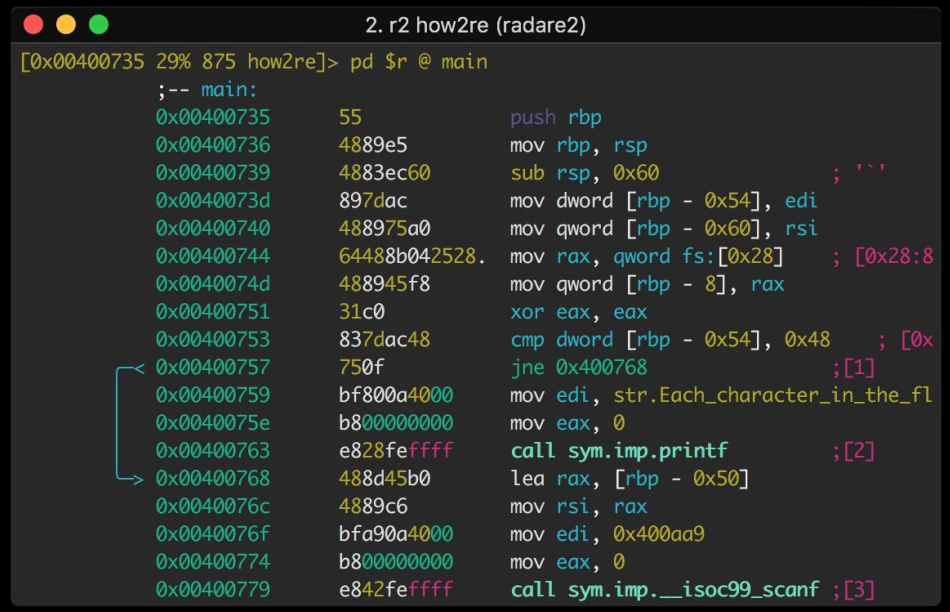
2. r2 how2re (radare2)

```
[0x004005e0 23% 100 how2re]> pd $r @ entry0
           ;-- entry0:
           ;-- section..text:
           ;-- rip:
           0x004005e0      31ed           xor ebp, ebp                ; [14] -r
           0x004005e2      4989d1         mov r9, rdx
           0x004005e5      5e             pop rsi
           0x004005e6      4889e2         mov rdx, rsp
           0x004005e9      4883e4f0       and rsp, 0xfffffffffffffff0
           0x004005ed      50             push rax
           0x004005ee      54             push rsp
           0x004005ef      49c7c0500a40.  mov r8, 0x400a50
           0x004005f6      48c7c1e00940.  mov rcx, 0x4009e0
           0x004005fd      48c7c7350740.  mov rdi, 0x400735           ; main
           0x00400604      e8a7ffffff     call sym.imp.__libc_start_main ;[1]
           0x00400609      f4             hlt
           0x0040060a      660f1f440000   nop word [rax + rax]
           0x00400610      b867106000     mov eax, 0x601067
           0x00400615      55             push rbp
           0x00400616      482d60106000   sub rax, 0x601060
```

Press p to toggle print mode (easier to see instructions)

You can navigate with the arrow keys, but that is slow.

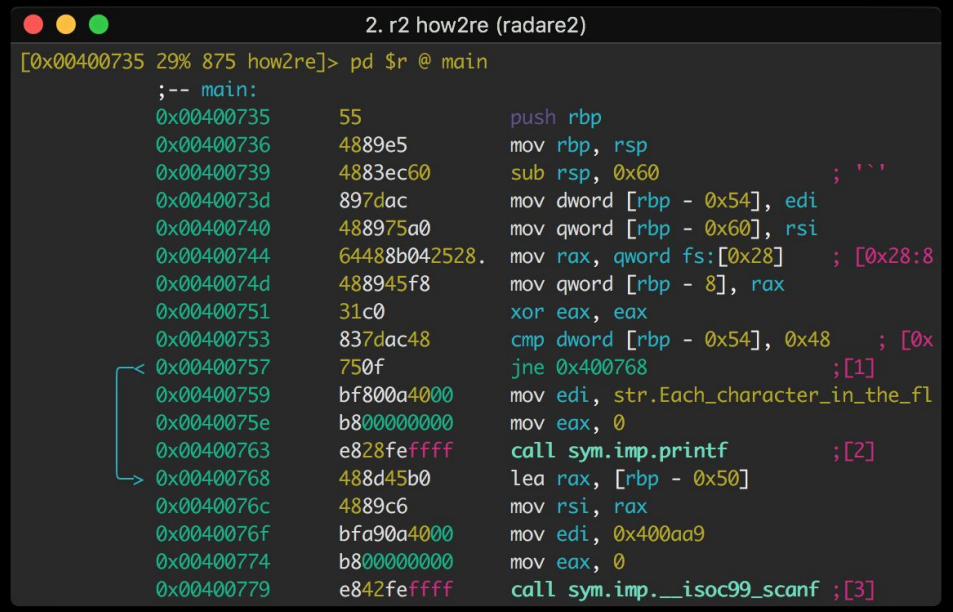
Moving faster



Press **n** and **N** to navigate between sections

You usually want to look for the **main** function, as that is where things are going on.

Making it even easier to read.

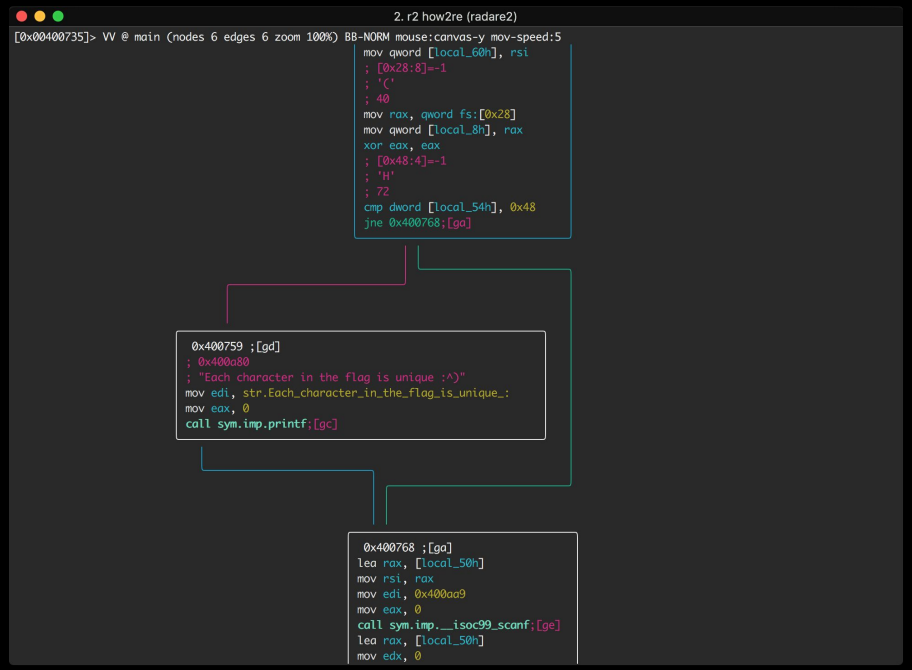


Press d, and then f.

d means define, and **f** is function.

This **defines** main as a **function,** and allows us to look at it in a much easier to understand way (visual representation)

Graphical Representation



Press V to enter visual mode, this allows you to see what is actually going on in the script in a nice visual way.

You can see where jumps go, true or false

Helpful Radare2 Book (From actual website)

https://radare.gitbooks.io/radare2book/content/first_steps/intro.html

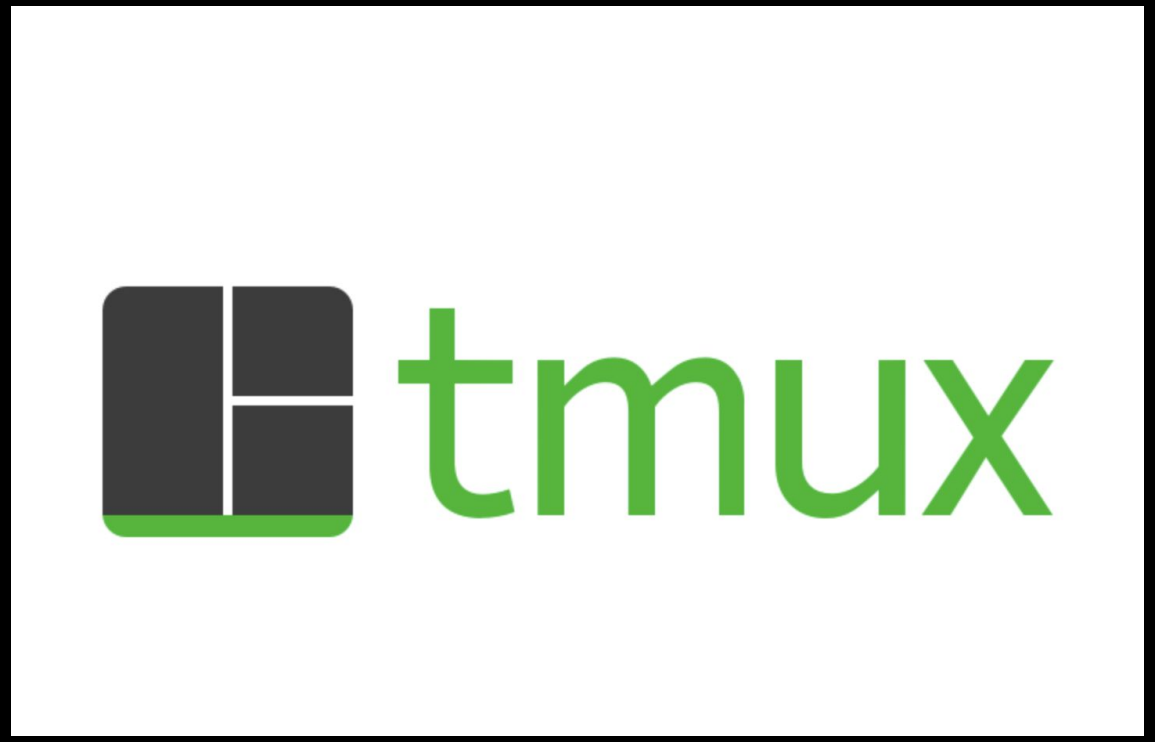
tmux

A really jank way to keep processes running after you close the terminal window

(So basically the thing everyone uses)

So if you want to… keep a ctf up, run a file sharing system, or run a Minecraft Server without needing a terminal window open.

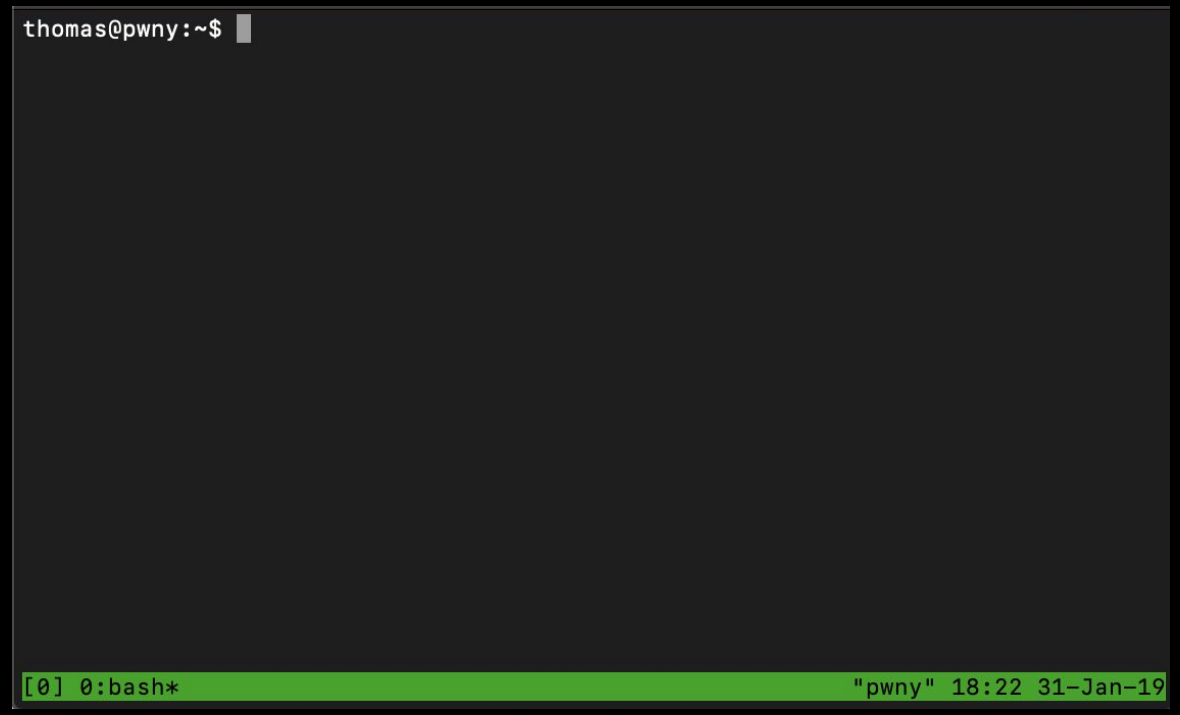
How to install tmux



It would be a good learning experience if you figure out how to install bash extensions on your operating system. So go try to do that.

It may also already be installed, the command is… **tmux**

tmux basics



To create a new window type **tmux**

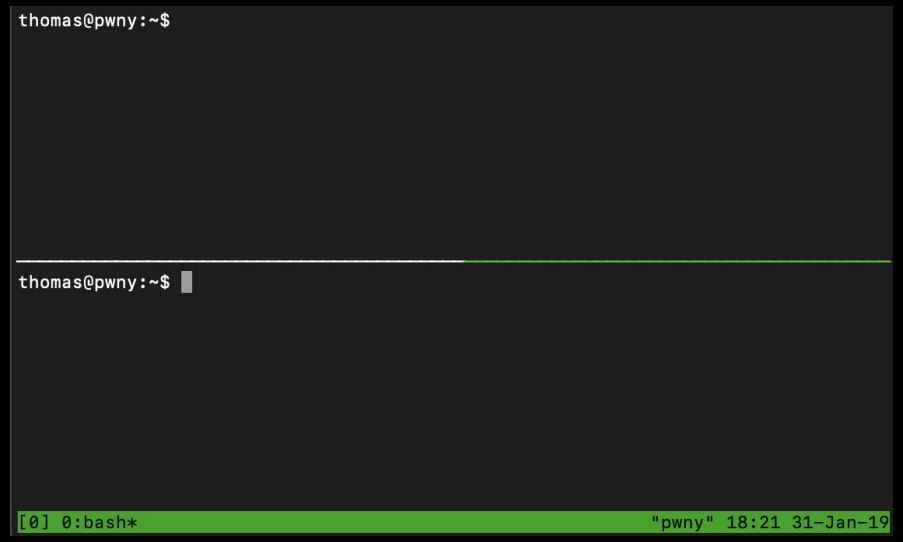
This will open up a new tmux window for you to use.

Mess around with it see what you can do, it is a fully functioning bash window.

NEVER nest multiple tmux windows

Just a bad idea ;P

tmux commands



The command prefix is cntrl + b + ____, some things you can do with this are

- d : detatch your terminal from the tmux window
  - This will keep anything you had running, still running.
  - You can get back to that session with **tmux at -t (number_of_session)**
- “ : horizontal split of terminal
- %: vertical split of terminal
- Force close split
  - Ctrl-d, **exit**

Detaching windows

- Cntrl+b + d to detatch a window
  - Will run REALLY long
- Reattach
  - tmux attach -t NUMBER
  - tmux ls
  - tmux new -s NAME
  - tmux rename-session -t NUMBER NAME

Easy tutorial

https://www.hamvocke.com/blog/a-quick-and-easy-guide-to-tmux/

The other stuff

Challenges are hard for this one, but they are good to know

git

Git is difficult, we could do meetings on meetings on meetings. For this meeting, know the following.

`git clone [url] [folder]`

Clones a repository from a url

`git add -A`

stages all unadded files to the **repository**

`git commit -m "Commit message"`

Commits those stages to your personal **branch**

`git push`

Pushes your branch to the main branch

`git pull`

Pulls the latest changes from the main branch

Vim

- Vim is an in terminal text editor
- It is NOT an IDE
- **vim** to open new vim window

EXIT VIM with :wq (colon + wq)

i = Insert Mode

dd = delete line

p = paste deleted line

:tabnew FILE_NAME = open new tab, gt to navigate tabs

:LINE_NUMBER = jump to line number

Use .vimrc!!!

package managers

apt - standard linux package installer

brew - apt but for mac

pip/npm/etc... -

rc files

rc = run commands

vimrc, bashrc

alias variable=’command’

alias pwny=’ssh thomas@sigpwny.com’

vimrc = number

Questions?

Next Week

Thursday: Web Hacking I!

- Introduction to internet fundamentals
- How to view page source, network, storage, hidden pages, etc.

Weekend Seminar: Web Hacking II!

- Attacks on web frameworks
- xss, CSRF, SSRF, SQL Injection, Packet Injection