Linux Kung-Fu

James Droste

UBNetDef Fall 2016

$ init 1

- GO TO https://apps.ubnetdef.org
- GO TO https://apps.ubnetdef.org
- GO TO https://apps.ubnetdef.org
- GO TO https://apps.ubnetdef.org
- GO TO https://apps.ubnetdef.org
- GO TO https://apps.ubnetdef.org
- GO TO https://apps.ubnetdef.org
- GO TO https://apps.ubnetdef.org
- GO TO https://apps.ubnetdef.org

$ get_server_info

- **GO TO: https://apps.ubnetdef.org/ssh**

- **HOST**: 10.0.0.2
- **PORT**: 22
- **USER**: (you have this)
- **PASS:** (you have this)

$ whoami


I CAN SHOW YOU THE
LINUX TERMINAL

$ whoami

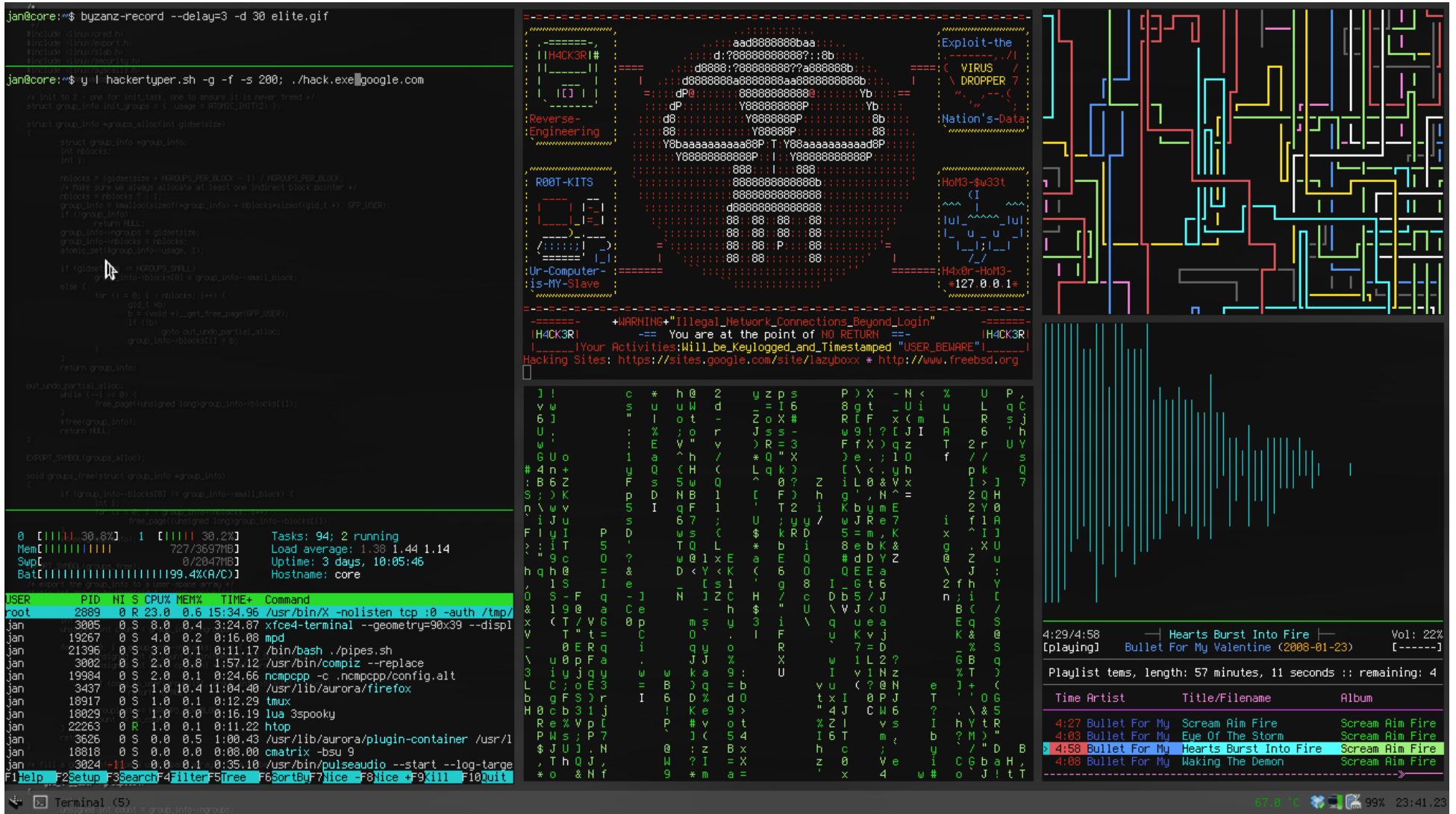
- James Droste
- Started using linux / the terminal at ~13
- Prefers the debian linux distribution (hint: the server you're connected to is running this!)
- Kind-Of System Administrator
- Typical Linux Guy for the UBNetDef Competition Team
- 21 years in dank memes

$ echo $ajenda

- The Basics
- Files Directories - Oh My!
- Processes
- Pipes and Redirects
- Becoming god (root)
- Package Managers
- Services
- Best Practices
- Advanced Stuff

The Basics: Level 1

$ what_is_the_terminal



$ reading_your_prompt

- root@netdef:~#

- User: root
- Computer Name (hostname): netdef
- Current Directory: ~ ← ?????
- Is Super User: Yes

$ what_is_the_terminal

- echo: Outputs (echo's) the data you pass into this function
  - `jamesdro@netdef:~$ **echo** Hello World`
    `Hello World`
- passwd: Allows you to change your own (or another user's) password
  - `jamesdro@netdef:~$ **passwd**`
    `Changing password for jamesdro.`
    `(Current) UNIX password:`
  - `root@netdef:~# **passwd** another-user`
    `Changing password for another-user.`
    `Enter new UNIX password:`

$ getting_help

- `$COMMAND` --help: "**Typically**" shows the help documentation for a command.
  - `jamesdro@netdef:~$ passwd` **`–help`**
    `Usage: passwd [options] [LOGIN]`
    `(snip)`

- man `$COMMAND`: Shows the **MAN**ual for `$COMMAND`.
  - `jamesdro@netdef:~$` **`man`** `passwd`
    `(Press` **`Q`** `to exit)`

$ user_management

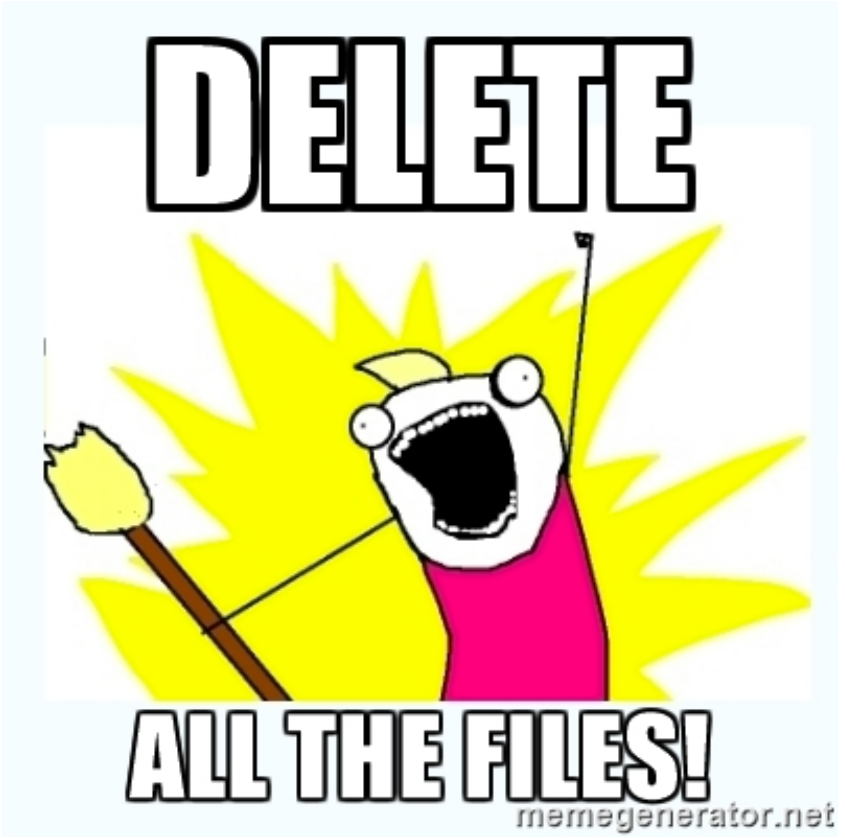
- adduser
- addgroup
- usermod
- groups

- su: Allows you to **S**witch **U**ser.
  - ```
    jamesdro@netdef:~$ su root
    Password:
    ```
    ```

$ user_management

- root@net-def:~# **adduser** test
  Adding user `test' ...
  Adding new group `test' (1006) ...
  Adding new user `test' (1004) with group `test' ...
  Creating home directory `/home/test' ...
  Copying files from `/etc/skel' ...
  (**snip**)
- root@net-def:~# **addgroup** another-test
  Adding group `another-test' (GID 1007) ...
  Done.
- root@net-def:~# **usermod** -G another-test -a test
- root@net-def:~# **groups** test
  test : test another-test

Files, Oh My!



$ moving_around

- ls: **L**i**S**t the files in a directory
  - `jamesdro@netdef:~$ ls`
  - `nuclear_launch_codes.txt secrets`
- cd: **C**hange **D**irectory
  - `jamesdro@netdef:~$ cd secrets`
  - `jamesdro@netdef:~/secrets$`
- pwd: **P**rint **W**orking **D**irectory
  - `jamesdro@netdef:~/secrets$ pwd`
    `/home/jamesdro/secrets`

$ moving_around_ADVANCED

- ls –al: Run the `ls` command, with the arguments "-al". Argument –a tells `ls` to print all files in the directory. –l tells `ls` to do it the "long" way.
  - `jamesdro@netdef:~$` **`ls -al`**

```
total 1536
drwx------  4 jamesdro jamesdro 4096    Oct  6 09:47 .
drwxr-xr-x 23 root root         4096    Aug 22 23:49 ..
-rw-------  1 jamesdro jamesdro 9244    Oct  6 01:48 .bash_history
-rw-r--r--  1 jamesdro jamesdro 570     Jan 31  2010 .bashrc
drwx------  3 jamesdro jamesdro 4096    Oct  5 14:50 .cache
-rw-r--r--  1 jamesdro jamesdro 1524722 May 21 19:30 nuclear_launch_codes.txt
-rw-r--r--  1 jamesdro jamesdro 140     Nov 19  2007 .profile
-rw-------  1 jamesdro jamesdro 1024    Aug 28 00:15 .rnd
-rw-r--r--  1 jamesdro jamesdro 66      Aug 28 01:51 .selected_editor
drwxr-xr-x  2 jamesdro jamesdro 4096    Oct  6 09:47 secrets
```

$ reading_is_important

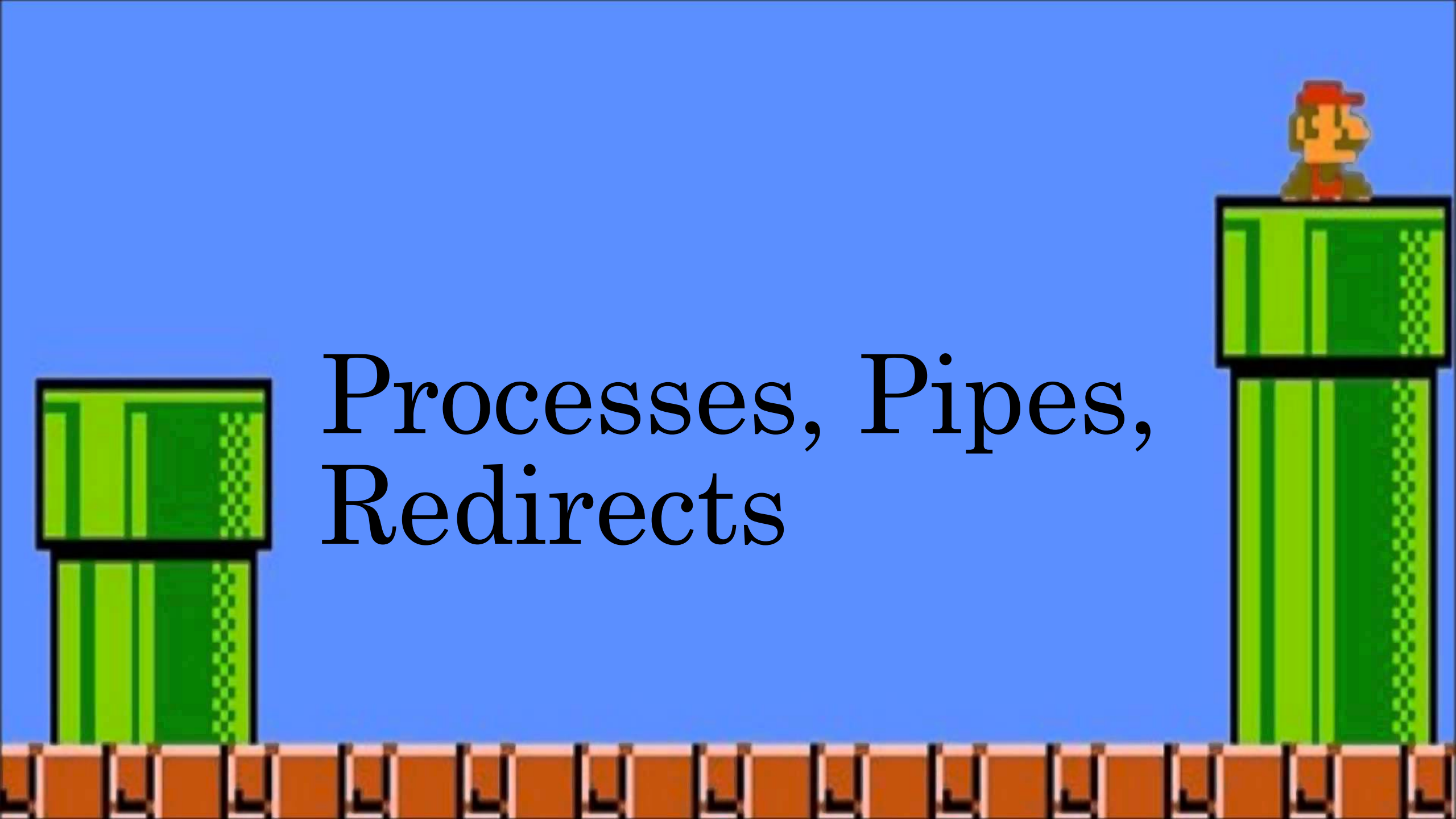
- cat: con**CAT**enate a file (aka: print the file contents to the terminal)
  - `jamesdro@netdef:~$ cat nuclear_launch_codes.txt`
  - `[Now I can't show you that ;-)]`
- less/more: Viewers that let you interactively scroll through a file.
  - `jamesdro@netdef:~$ less nuclear_launch_codes.txt`
- Let's view a file that contains all the user accounts on the system
  - /etc/passwd
  - WHO ARE THEY?

$ writing_is_cool_too

- nano
- vi
- vim
- ed
- emacs
- Magnetic needle

$ note_about_permissions

- Files/Folders have this thing called Access Control Lists (ACLs)
- This is a large topic on it self
- tl;dr:
  - Linux matches based on the file's owner, file's group, then everyone else.
  - Each permission set can have read, write, or execute privilege (any combination)
  - These can be represented with octal notation (1=execute, 2=write, 4=read)
  - Permissions can be changed with the command **`chmod`**
  - Ownership can be changed with the command **`chown`**
  - Group ownership can be changed with the command **`chown`** or **`chgrp`**


Processes, Pipes,
Redirects

$ viewing_processes

- ps: **P**rocess **S**napshot. Shows currently running processes.
  - `jamesdro@netdef:~$ **ps**`
  - `jamesdro@netdef:~$ **ps** aux`
- top/htop: Interactive way to show all the currently running processes
  - `jamesdro@netdef:~$ **top**`
  - `jamesdro@netdef:~$ **htop**`
- kill: Used when we want to ~~MURDER~~ kill a process.
  - `jamesdro@netdef:~$ **kill** <pid>`
  - `jamesdro@netdef:~$ **kill** -9 <pid>`
    - or: `jamesdro@netdef:~$ **kill** –SIGKILL <pid>`

$ pipes_and_redirects

- “|” is the pipe operator
- “>” is the redirect operator

- Examples
  - `jamesdro@netdef:~$ ps aux | less`
  - `jamesdro@netdef:~$ ps aux > some_file.txt`
  - `jamesdro@netdef:~$ who | awk '{ print $1 }' > users.txt`

Here Be Dragons

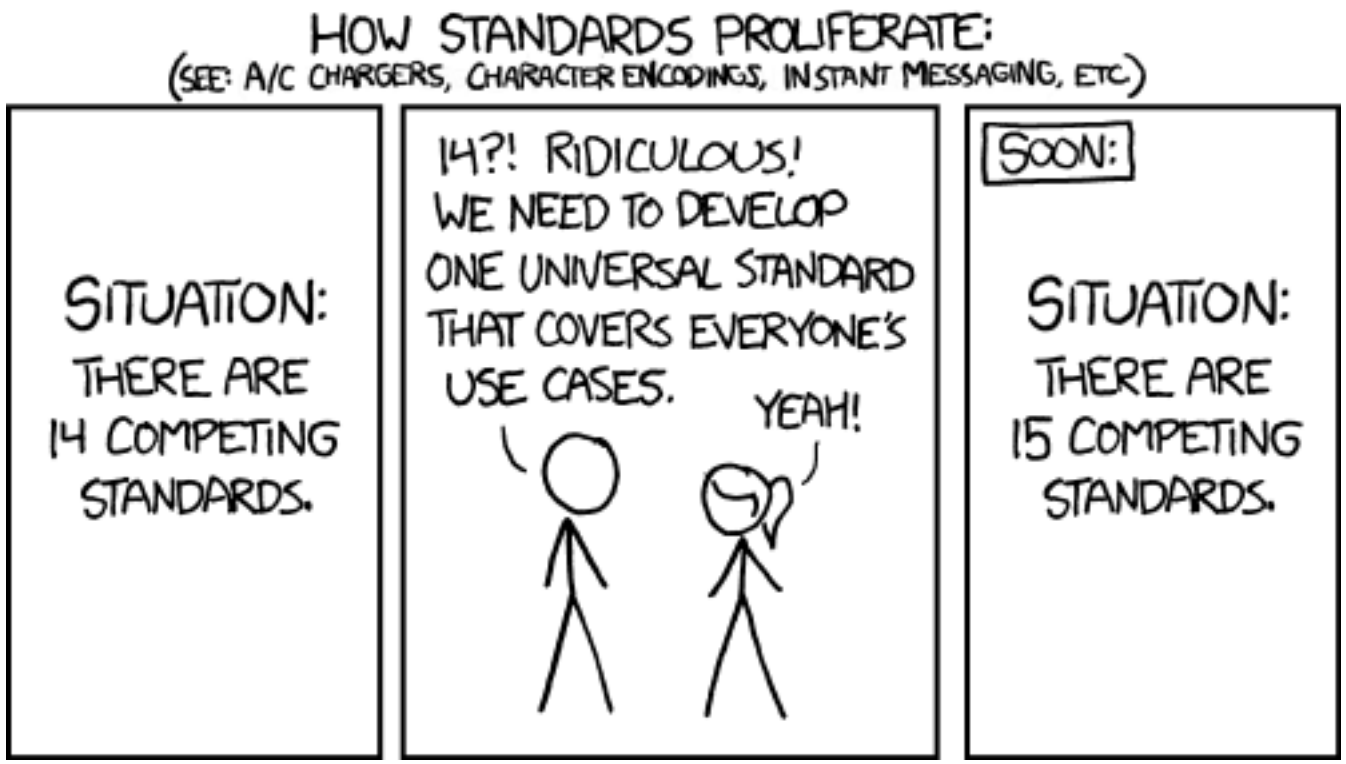
$ woot_woot_got_root

- sudo: **S**uper **U**ser **DO**. Runs a specific command as a super user (uid=0, aka root).
  - `jamesdro@netdef:~$ **sudo** whoami`
    `root`
  - `jamesdro@netdef:~$ whoami`
    `jamesdro`

$ packages

- aptitude: Debian, Ubuntu
- apt-get: Debian, Ubuntu
- dpkg: Debian, Ubuntu
- yum: CentOS
- dnf: Fedora

$ packages_2


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

$ packages_3

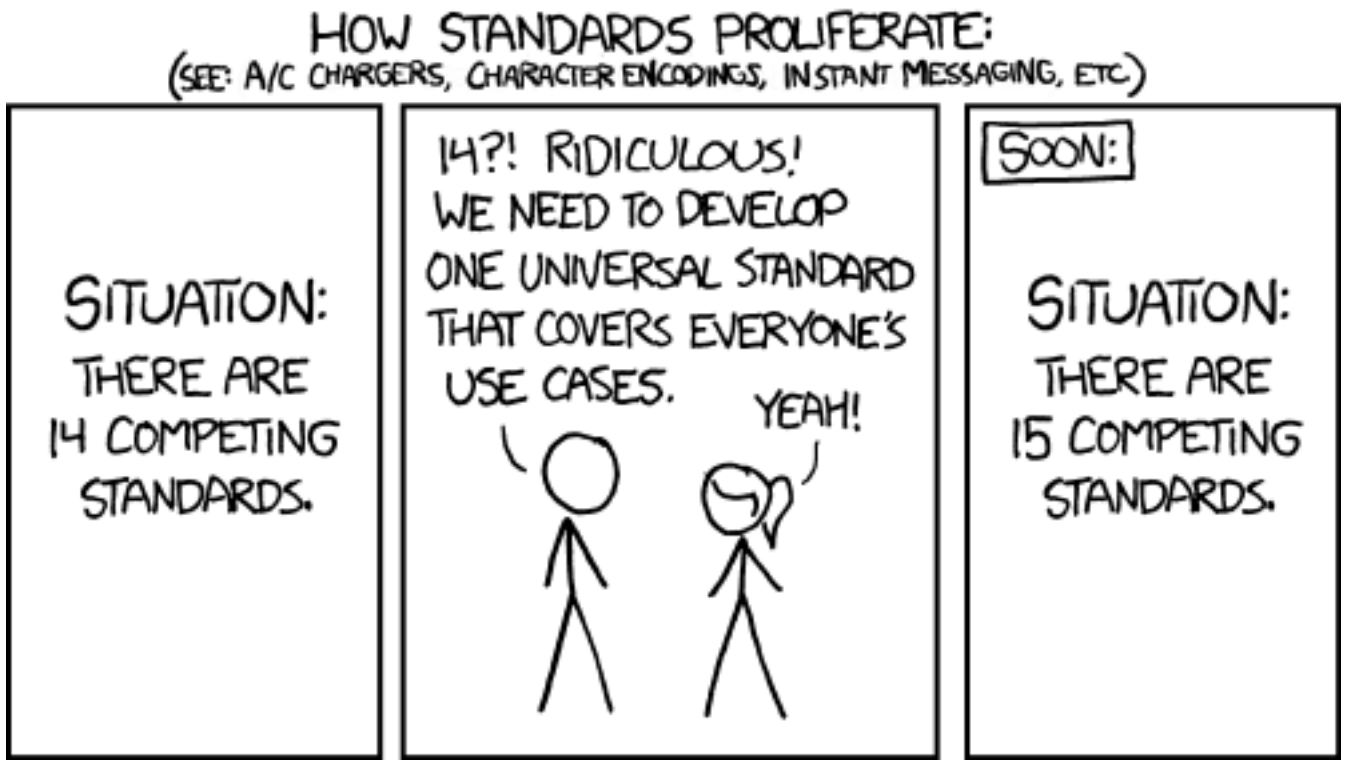
- Installing a package
  - `jamesdro@netdef:~$ sudo apt-get **install** <package-name>`
- Removing a package
  - `jamesdro@netdef:~$ sudo apt-get **remove** <package-name>`
- Updating the local package index
  - `jamesdro@netdef:~$ sudo apt-get **update**`
- Updating a package
  - `jamesdro@netdef:~$ sudo apt-get **upgrade** <package-name>`
- Updating ALL packages
  - `jamesdro@netdef:~$ sudo apt-get **upgrade**`

$ packages_4_this_never_ends

- Typical flow (installing package apache – which is a web server)
  - ```
    jamesdro@netdef:~$ sudo apt-get update
    jamesdro@netdef:~$ sudo apt-get install apache
    ```

- Daily Ritual
  - ```
    jamesdro@netdef:~$ sudo apt-get update
    jamesdro@netdef:~$ sudo apt-get upgrade
    ```

$ services_1


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

$ services_2

- systemctl: Part of this new subsystem that is called “SystemD”
- service: Part of System V init
- /etc/init.d/<service> <start | stop | restart | status>: Part of System V init, deprecated

$ services_3_electric_boogaloo

- Starting a service - apache
  - Using systemd:
    - `jamesdro@netdef:~$ sudo systemctl start apache`
  - Using sysvinit:
    - `jamesdro@netdef:~$ sudo service apache start`

- Stopping a service - apache
  - Using systemd:
    - `jamesdro@netdef:~$ sudo systemctl stop apache`
  - Using sysvinit :
    - `jamesdro@netdef:~$ sudo service apache stop`

$ linux_directories_revealed

- /bin: Contains all binaries that are necessary for the system to function
- /boot: Contains the linux bootloader
- /dev: Contains all raw **DEV**ices
- /etc: Contains configuration files for the system
- /home: Contains user's home directories
- /mnt: Typically used when mounting devices (like a cdrom)
- /opt: Optional, addon packages
- /proc: Here be dragons
- /root: User root's home directory
- /sbin: Contains all the binaries that are necessary for the system to run. Only super users can use these.
- /tmp: Temporary files
- /usr: User install files

$ good_files_to_know

- /etc/passwd
- /etc/shadow
- /etc/crontab (/etc/cron.{d,daily,weekly,hourly,monthly}/*}

$ init 0