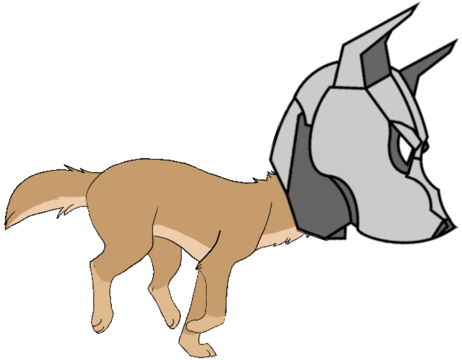
Cyber Defense Organization

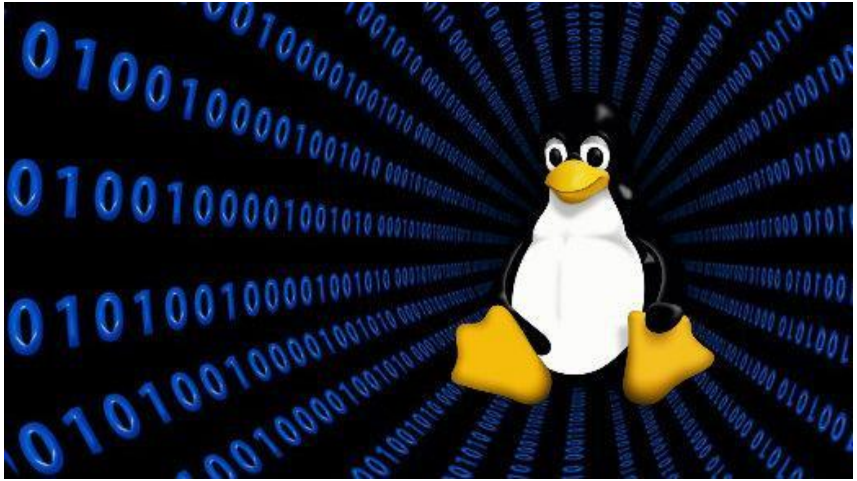
Fall 2019 - Introduction to Linux



Introduction to Linux

- Introduce you to the fundamentals of Linux
- Familiarize you with history of Linux
- Groundwork for future

  workshops/competitions

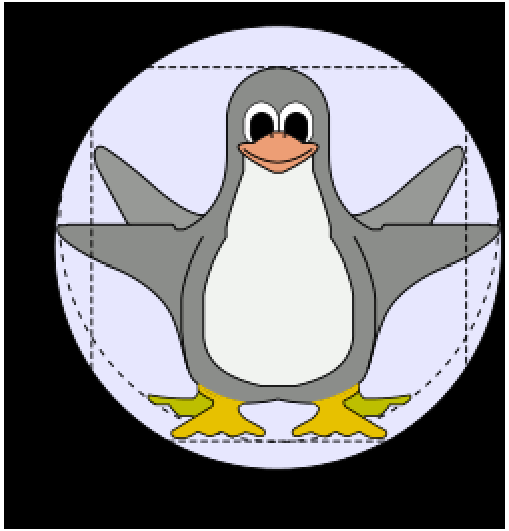


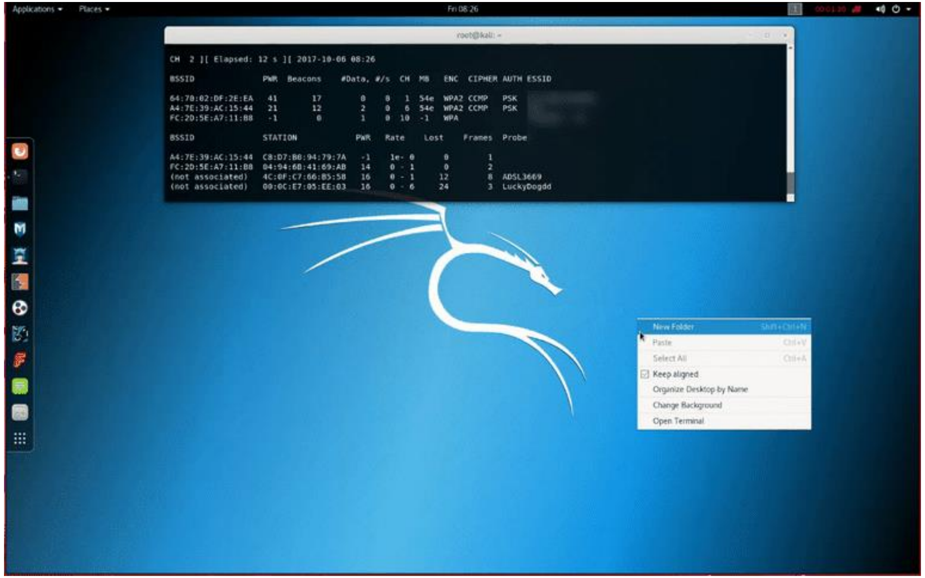
What is Linux?

- Linux is NOT an OS. Linux is a kernel.
- The difference between an OS and kernel is that a kernel interacts directly with the hardware.
- Any OS uses a kernel though.

Types of Linux?

- Ubuntu
- Kali (Used in tv show “Mr Robot”)
- CentOS
- Mint
- Cucumber Linux



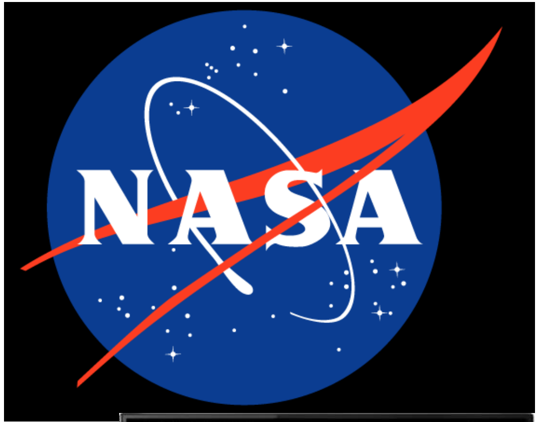


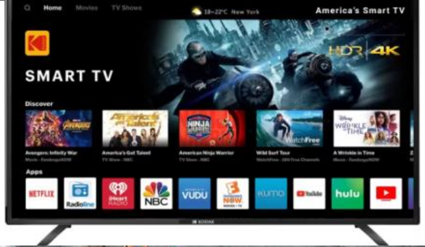
```
CH  2 ][ Elapsed: 12 s ][ 2017-10-06 08:26

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

64:70:02:DF:2E:EA   41       17        0    0   1  54e  WPA2 CCMP   PSK
A4:7E:39:AC:15:44   21       12        2    0   6  54e  WPA2 CCMP   PSK
FC:2D:5E:A7:11:88   -1        0        1    0  10  -1   WPA

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

A4:7E:39:AC:15:44  C8:D7:B0:94:79:7A   -1    1e- 0      0        1
FC:2D:5E:A7:11:88  04:94:6B:41:69:AB   14    0 - 1      0        2
(not associated)   4C:0F:C7:66:85:58   16    0 - 1     12        8  ADSL3669
(not associated)   00:0C:E7:05:EE:03   16    0 - 6     24        3  LuckyDogdd
```

Why Linux?

- Free
- Command line based
- Stripped down
- You can directly modify files

Linux is used in so many things!

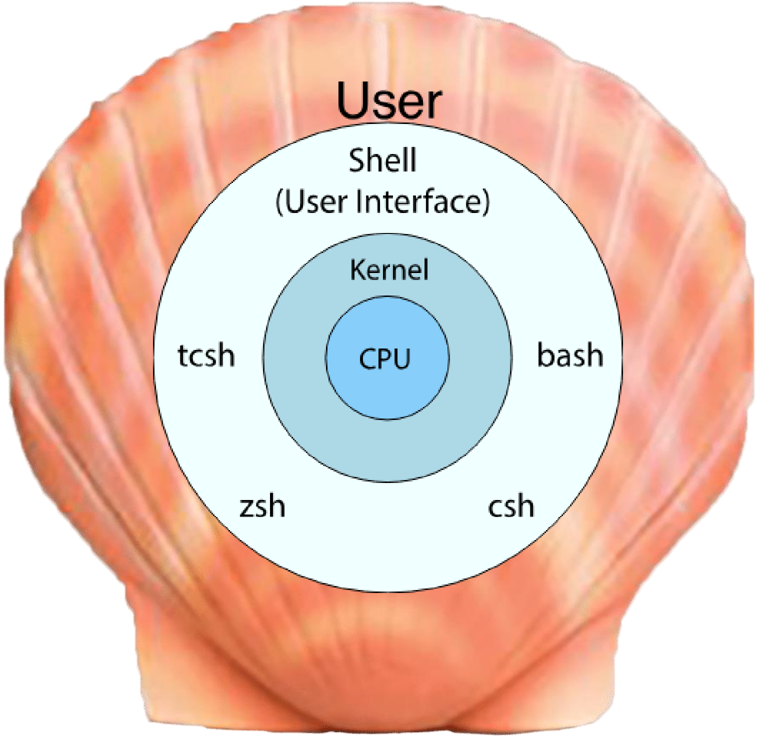




Ur mom

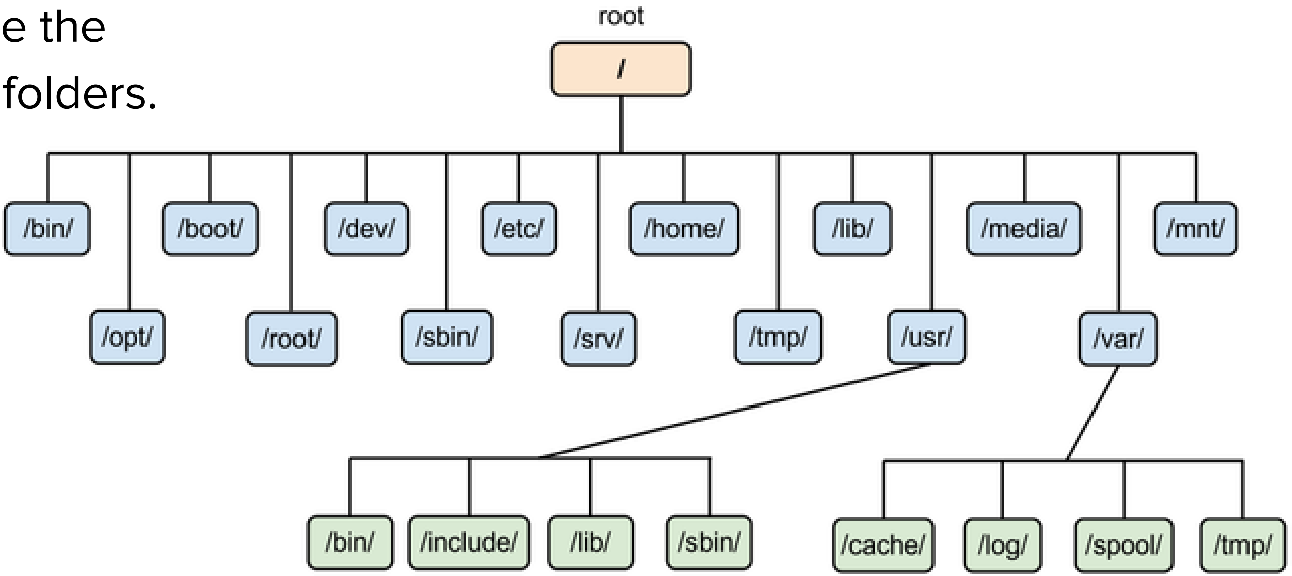
The workings on Linux

- At its core level linux is a kernel that interacts with your computer's hardware.
- Outside the kernel are the shells.
  - The shell is the user interface that communicates with the kernel.



Linux File Structure

- The file systems in linux are broken up into a bunch of directories.
    - Directories are the equivalent of folders.



Linux framework pt2

/ - filesystem root

/bin - contains programs

/sbin - contains programs for admins

/etc - configuration files for programs

/opt - downloaded programs

/home - each user has files live there

/dev - attached devices information (usbs)

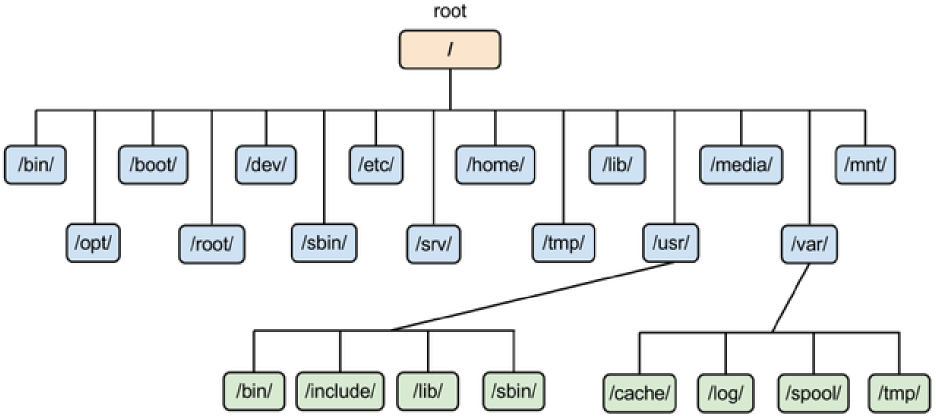
/var - variable files(LOGS!)

/tmp - temporary files

. - current directory

.. - go up one directory

- - go back



Moving around

We use commands to navigate the OS!

pwd - (print working directory) This shows you where you are in the file system

ls - show all the files in the current folder (HINT -a to show hidden files)

Cd <directory name> - Changes the directory (example cd /home moves you to the home directory)

Cat <file name> - shows you the contents of the file

File <filename> - shows you about the file and its type

Follow along with the Hands on Activity!

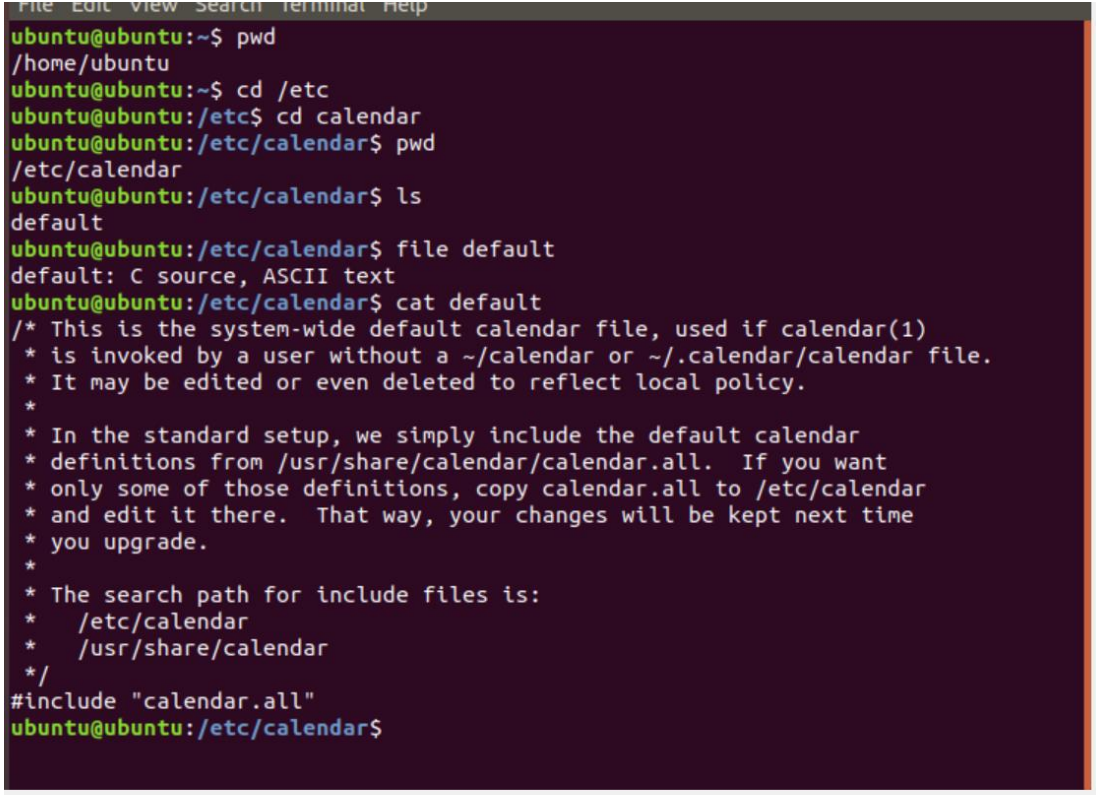
https://tinyurl.com/LinuxIntroCDO

Lets actually start!

- Today we are playing with Ubuntu
- Boot them boxes up!!
- Open the terminal
- Use the PWD command to see your file path
- Cd into the /etc directory
- Now cd into the calendar directory
- Now type the command pwd to see where you are now!
- Ls to see the contents.
- Use the file command to see what the default file is
- Use the cat command to see the contents of default

BOOT!!! GET IT???

Final product!!!



```
ubuntu@ubuntu:~$ pwd
/home/ubuntu
ubuntu@ubuntu:~$ cd /etc
ubuntu@ubuntu:/etc$ cd calendar
ubuntu@ubuntu:/etc/calendar$ pwd
/etc/calendar
ubuntu@ubuntu:/etc/calendar$ ls
default
ubuntu@ubuntu:/etc/calendar$ file default
default: C source, ASCII text
ubuntu@ubuntu:/etc/calendar$ cat default
/* This is the system-wide default calendar file, used if calendar(1)
 * is invoked by a user without a ~/calendar or ~/.calendar/calendar file.
 * It may be edited or even deleted to reflect local policy.
 *
 * In the standard setup, we simply include the default calendar
 * definitions from /usr/share/calendar/calendar.all.  If you want
 * only some of those definitions, copy calendar.all to /etc/calendar
 * and edit it there.  That way, your changes will be kept next time
 * you upgrade.
 *
 * The search path for include files is:
 *   /etc/calendar
 *   /usr/share/calendar
 */
#include "calendar.all"
ubuntu@ubuntu:/etc/calendar$
```

Manipulating files

Touch <name> - creates a file of designated name in current directory

Rm <name> - removes a file of that name in the current directory

Mkdir <name> - create a directory with that name

Rmdir <name> - remove the directory with that name.

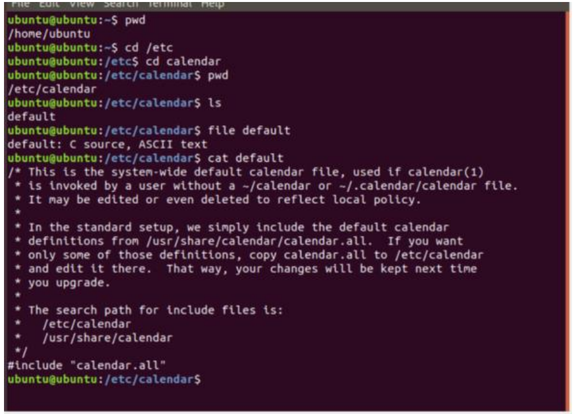
Nano <file name> - edit the text inside of a file

Vi <file name> - the more difficult way to edit text

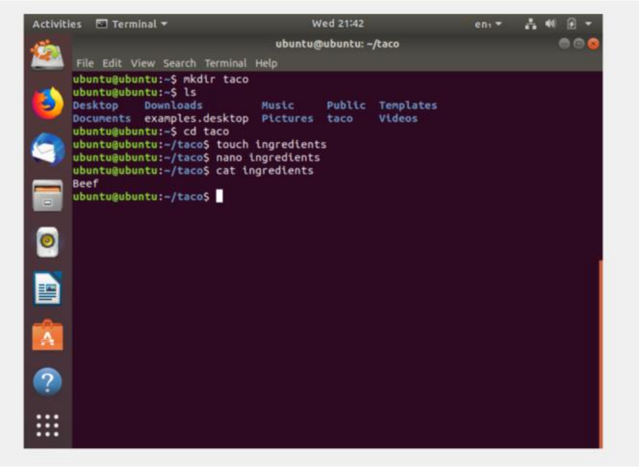
Let's give it a go!

- Type cd to go back to your home directory
- Create a directory called taco with the mkdir command (you can use ls to check if it worked)
- Cd into taco
- Create a file called ingredients using the touch command
- Now you wanna put the ingredients of the taco in there, so use nano to edit the ingredients file
- Write beef in it (use the commands written in the bottom to save and exit [control x])
- Now test to make sure it is in there by using the cat command from earlier
- Remove the file with the rm command
- Now we want to remove taco so use the rmdir command to remove it!

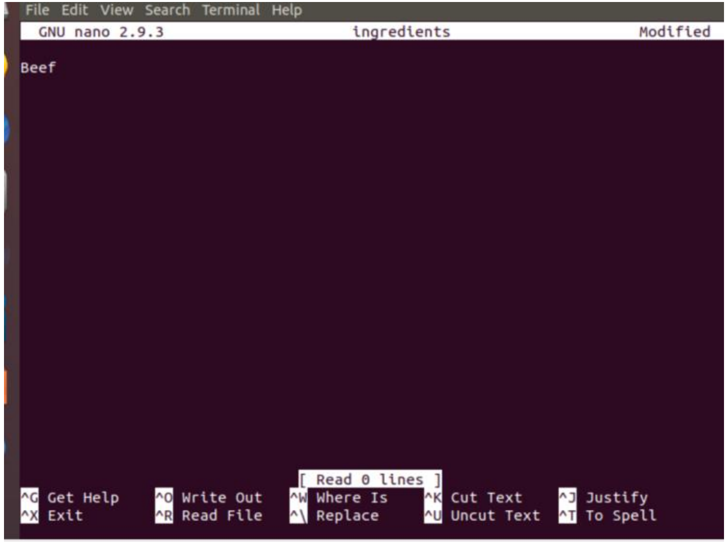
Example!



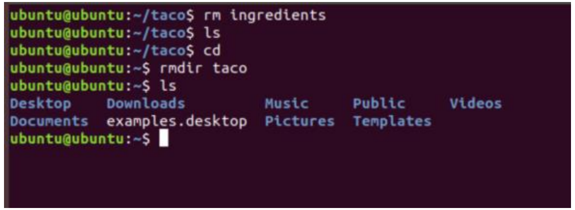
```
ubuntu@ubuntu:~$ pwd
/home/ubuntu
ubuntu@ubuntu:~$ cd /etc
ubuntu@ubuntu:/etc$ cd calendar
ubuntu@ubuntu:/etc/calendar$ pwd
/etc/calendar
ubuntu@ubuntu:/etc/calendar$ ls
default
ubuntu@ubuntu:/etc/calendar$ file default
default: C source, ASCII text
ubuntu@ubuntu:/etc/calendar$ cat default
/* This is the system-wide default calendar file, used if calendar(1)
 * is invoked by a user without a ~/calendar or ~/.calendar/calendar file.
 * It may be edited or even deleted to reflect local policy.
 *
 * In the standard setup, we simply include the default calendar
 * definitions from /usr/share/calendar/calendar.all.  If you want
 * only some of those definitions, copy calendar.all to /etc/calendar
 * and edit it there.  That way, your changes will be kept next time
 * you upgrade.
 *
 * The search path for include files is:
 *   /etc/calendar
 *   /usr/share/calendar
 */
#include "calendar.all"
ubuntu@ubuntu:/etc/calendar$
```



```
ubuntu@ubuntu:~$ mkdir taco
ubuntu@ubuntu:~$ ls
Desktop    Downloads         Music     Public  Templates
Documents  examples.desktop  Pictures  taco    Videos
ubuntu@ubuntu:~$ cd taco
ubuntu@ubuntu:~/taco$ touch ingredients
ubuntu@ubuntu:~/taco$ nano ingredients
ubuntu@ubuntu:~/taco$ cat ingredients
Beef
ubuntu@ubuntu:~/taco$ 
```



```
  GNU nano 2.9.3                 ingredients                     Modified

Beef





                              [ Read 0 lines ]
^G Get Help     ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify
^X Exit         ^R Read File    ^\ Replace      ^U Uncut Text   ^T To Spell
```

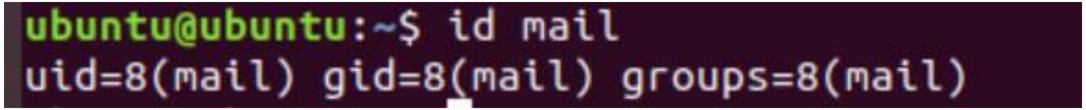


```
ubuntu@ubuntu:~/taco$ rm ingredients
ubuntu@ubuntu:~/taco$ ls
ubuntu@ubuntu:~/taco$ cd
ubuntu@ubuntu:~$ rmdir taco
ubuntu@ubuntu:~$ ls
Desktop    Downloads         Music     Public     Videos
Documents  examples.desktop  Pictures  Templates
ubuntu@ubuntu:~$ 
```

Users and groups!

- Linux uses users and groups to manage access
- Each user gets an identification called a UID
- Each group gets a GID
- Use the id command to see the UID and GID
- Id <user or group>

Courtesy: 20th Century Fox Studios



```
ubuntu@ubuntu:~$ id mail
uid=8(mail) gid=8(mail) groups=8(mail)
```

Modifying users and groups!

useradd <username> - add new user

deluser <username>- delete user

groupadd <groupname>- create new group

groupdel <groupname>- delete group

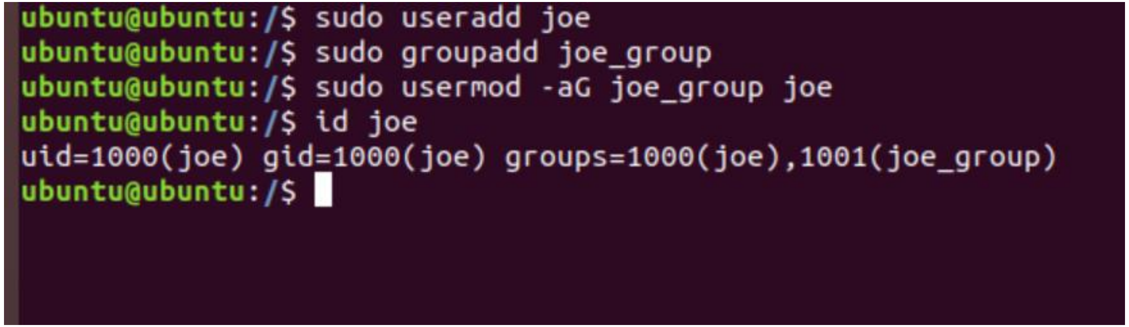
usermod -aG <groupname> <username> - Add user to group.

whoami   - show logged in

who   - who else is logged in

Let’s a go!

- Create a user with the useradd command. Call them “joe”.
- If you are having issues try to put sudo in front of it
- You may want to ask who joe is but that is a question for another time.
- Now you wanna make a group for joe try using groupadd and make the group joe_group
- Next, add joe to joe_group with the usermod -aG command.
- Check it with id joe



```
ubuntu@ubuntu:/$ sudo useradd joe
ubuntu@ubuntu:/$ sudo groupadd joe_group
ubuntu@ubuntu:/$ sudo usermod -aG joe_group joe
ubuntu@ubuntu:/$ id joe
uid=1000(joe) gid=1000(joe) groups=1000(joe),1001(joe_group)
ubuntu@ubuntu:/$ 
```

Ease of use commands

<ctrl-c> : kill current process

<ctrl-z>: put current process in background

<tab>: complete the command

!!: re-run recent command

jobs : view background processes

history - view recent commands

clear or <ctrl-L) - clear the screen

Curious about a command?

Just use the man command!

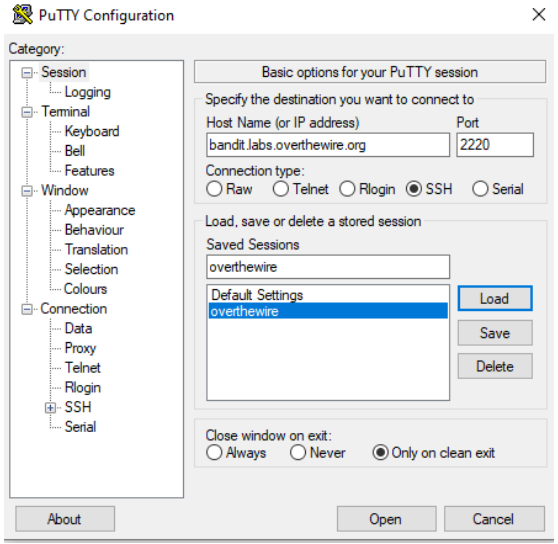
This gives you a manual of what that command is and does!!!

Who doesn’t like free knowledge!

On our own

GO TO: https://overthewire.org/wargames/

OPEN PUTTY:



PuTTY Configuration

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): bandit.labs.overthewire.org

Port: 2220

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

Saved Sessions: overthewire

Default Settings
overthewire

Close window on exit: ○ Always ○ Never ● Only on clean exit

Username and password for level 0 is bandit0

As stated on site look for the first password in a file called readme.

Use that password in a new putty session with the username bandit1

And so on!!

Add us on Social Media!

Twitter: **@ualbanyCDO**

Instagram: **ualbany_cdo**

Website: **uacyber.org**

Myinvolvement: **Cyber Defense Org**

https://github.com/cyber-defense-organization

Cya Next Time!

Next time we have intro to Networking!!!