Study up on this
zero defined networking

Stuff to do on the side if I want to set up a monitoring service on my desktop
https://mitmproxy.org/
https://help.ubuntu.com/community/Router
https://www.elastic.co/downloads/elasticsearch
https://www.elastic.co/guide/en/beats/packetbeat/current/packetbeat-installation.html

```
/etc/sysconfig/network-scripts
ll
vi ifcfg-ens192
```

double tab shows you the way

Linux essentials course

Yum Command

Yum search https
- search options helps find software
yum info httpd
- info option helps query the software
yum install httpd
- install option installs the software

yum update -y
- updates everything

yum list installed httpd
- list installed option checks to see if it is installed

yum deplist httpd
- checks the dependencies

yum remove httpd
- removes the software

yum autoremove httpd
- autoremove option removes the web server and its dependencies

yum repolist
- checks to see where you got the repository

cd *etc*/yum.repos.d
- you can cd into the repos locations

yum clean all
- cleans all options clears all local yum database for more updates

yum update
- update helps improve the software and updates it

yum list installed httpd
- list installed option checks to see if the application is installed

which httpd
- checks to see which directory the application is in

yum provides or yum whatprovides

rpm command – **does not check for dependencies**

rpm – i <rpm file name.rpm>
- installs the package

rpm -q <package name>
- finds the full package name and architecture

rpm -qi <package name>
- helps you find more info regarding the package

rpm -ql <package name>
- displays a list of the stuff installed with the package

rpm -qR <package name>
- shows you what is required with the rpm

rpm – Uvh <rpm file name.rpm>
- updates the package

rpm -qi <package name>
- queries the package to see if you have the latest version

rpm -e –test<package name>
- test to see what software is safe to remove

rpm -e <package name>
- deletes the package from the system

APT Command – for debian systems

cd *etc*/apt
- finds the repo location

less source
- displays a list of repos that are config for this system

apt update
- updates the information for the system

apt full-upgrade

- updates the entire system

```
apt-cache search <package name>
```
- searches to see if there is a specific package installed

```
apt install <package name>
```
- installs the package

```
apt remove <package name>
```
- removes the package

```
apt remove –purge <package name>
```
- removes all of the items that might been left behind when removing the software

```
apt autoremove
```
- helps with removing left overs

dpkg command – The original package manager for debian

```
dpkg –get-selections
```
- helps see if something is installed

```
dpkg-deb -I <debian file.deb>
```
- gathers more information

```
dpkg –get-selections <package name>
```
- checks if the program is installed

```
dpkg-deb –contents <debian file.deb>
```
- checks to see what is contained in this file

```
dpkg -i <debain file.deb>
```
- installs the package

```
apt update
```
- updates the stuff that it does not have

```
apt -f upgrade
```
- download the dependencies needed

```
dpkg -r <package name>
```
- used to uninstall a package

```
dpkg -l <package name>
```
- checks to see if the package is installed

dpkg -p <package name>
- purges the rest of the stuff that came with the package

ls command

ls
– list out of the directory

ls -a
- list out all of the directory (.<file or folder name> - indicates a hidden item)

ls -l
- long listing lets you know more about the directory – shows the type and commissions
- d = directory, -= regular file, group owner, size in bits, m/d/year file was modified, file name

ls -Sl
- list out the size from greatest to smallest

ls -lrS
- list out the previous option but in reverse

ls -l <directory name> or ls <directory name>
- list out the stuff in a specific file in the directory

ls -R
- can look at the content of the directories within directories

ls -lt
- list out the time they were modified with the most recent at the top

echo – print what follows to the content screen
echo $PATH – prints the contents of the path environment variable to the screen
. = current directory
.. = one directory up

Basic commands

whoami

- shows you the username of the user currently logged in

su <username>

- allows you to log in to the other user as long as you know there password

exit

- gets you out of the other users shell environment

su -

- allows you to log in as root

reboot

- reboots the linux system

halt or poweroff

shutdown --help

- will list out the options for shutting down the computer

top

- shows you the top running processes on your computer

to get out of top enter the q

uname commands

uname

– display the name of the system kernel

uname -r

- display the kernel release number

uname -v

- display the kernel build version

uname -m

- display the machine type

uname -o

- display the name of the operating system

uname -a

- display all information that uname can show

cd and pwd commands

pwd
– print working directory, the current directory that you are in
cd
 – change directory
cd ..
- go up one directory
cd -
- returns you to your previous directory
cd ~
- .change to the home directory of the currently logged in

Command history and Command completion

.bash_history
- hidden file within the home directory that contains a log of commands entered at the bash prompt

cat <file name>
- can be used to print the contents of the file to the screen

echo $HISTFILESIZE
- shows the file size of how many commands it can keep

echo $HISTFILESIZE
- shows how you what you history contained

history
- command prints out the commands that have been stored in the .bash_history

!<history number>
- runs that specific command

Globbing

* - matches zero or more characters
Example: ls *.txt
? - matches any single character
Examples:
ls ?.txt
ls Test?.txt – helps by showing you what ends with Test
using more ? Helps improve the search

ls [Ww]<name of file>
- doing brackets in front of the file or inside the file can help you find out if it the letter is cap or not
- it is also helpful in finding what the file name is if your forgot

Quoting

“ ” - double quotes, contains strings and any variables or commands within them get evaluated or acted on
‘ ’ - single quotes, anything within these get treated literally, disables any special characters functionality
\ - backslash, escape character, disables any special character functionality that immediately follows it

locate, find, and whereis command

locate – searches a local database of files and folders looking for items that match the search criteria
Example: locate passwd

find – searches the file system for files that match the search criteria find/path/to/folder -name file

whereis – locates binary, source and/or manual pages for a command

man command

man <content you want to search>
- this will show you all the possible commands for the stuff you would want to enter.

Whatis – command that lists summaries and related man pages based on search term, invoked by entering **whatis <command>**

apropos – command that searches man page for appearances of the keyword provided, invoked by entering **apropos<keyword>**

manman – is the manual for man pages

info command

info<command>
- information command utility command, invoked by entering **info<command>**

Filesystem Hierarchy Standard(fhs)
- this standard was created to allow linux users to know how to find different files

Files and Directories
mkdir <file name>
- makes new folder

mkdir dir1 dir2 dir3
- this is how you make multiple folders at once

mkdir **-**p <dirname>/<dirname>
-makes a parent directory along with the sub-directory

rmdir <dir name>
- removes an empty directory

touch <filename>
- creates an empty file or update a file’s timestamp

cp <dir>/<filename>
- cp copy a file or folder

mv
- move or rename a file or folder

rm <filename>
- removes a file or folder

cp -R
-R copy a file or folder

rm -r
- recursively removes a folder and its contents

Archives and Compression

tar
- manipulate archive files
tar -c
- creates a new archive
tar -z
-passes the archive through gzip compression
tar -z
-pass the archive through bzip2 compression
tar -f
- file name of archive to create
tar -x
- extract an archive
tar -v
- verbose output
zip -r
- recursively creates a compressed file of directory and its contents
example: zip -r for_IT.zip for_IT
unzip
- extract a zip archive
example: unzip for_IT.zip
gzip
- create a gzip archive
gunzip
- extract a gzip archive
bzip2
- create a bzip2 archive
bunzip2
- extract a bzip2 archive

Viewing Text

less <file name>
- view a text file with the ability to scroll through the pages of the file

head <file name>
- view the first ten lines of a file

head – n <number><filename>
- views the first <number> lines of a file specified

tail <filename>
- view the last ten lines of a file

tail -n <number>
- views the last <number> lines of a file specified

tail -f <file location>
- this follows the text file as new data is written to it in real time

Analyzing Text

cut
- remove text from file and print specified fields to screen

cut -d
- specified delimiter to use

cut -f
- specifies which field to print

example: cut -d “ ” 6- anc.txt   - this will check spaces to the 6 space

sort <file name>
- sorts content of file alphabetically based on first character in file

sort -n
- sorts content of file numerically

wc <file name>
- word count, prints number of lines, words and characters in file

wc -l <file name>
- prints the number of lines in a file

wc -w
- prints the number of words in file

example: wc -lw <file name>

>
- redirects standard output to new location, if output goes to file replaces contents of file with output from stdout

example: head -n 1 art_pol.txt > anc.txt

>>
- redirects standard output to new location, appends stdout to file

example: head -n 1 art_pol.txt >> anc.txt

Pipes and regular expressions

grep <pattern><file to search>
- grep shows the lines in a file that match a given pattern
grep -i <string><file>
- performs a case-insensitive search
grep -v
- returns lines that do not contain the pattern
grep -r
- performs a recursive search

Regular Expression:
grep ‘^[AaBb] ‘ <file name>
^ = search the beginning of a line
grep ‘string$’ <file name>
$ = search the end of a line
. = stands in for a single character0
[abc] = search for specified characters
[^abc] search for other characters except for these
* = match zero or more of the preceding characters or expression

| = pipe
- pipe character, used to send output of one command as input to another command
ex: command1 | command2

vi/vim editor

vi = one of the original text editors for unix-like operating systems. Installed by default on most linux distributions
vim = successor to the vi text editor. Offers the same functionality plus extra features.

I= insert text under cursor
A – append text at end of line
u – undo last change in file
v – visual mode
p – past text
y - ‘yank’ or copy highlighted text
shift + g = go to bottom of file
gg = jump to top of editor
shift +a = append text at end of line
dw – delete whole line under cursor
dd – delete whole line under cursor
:w – writes the file out to disk
:wq – wrtie the file to disk then quit
:q! - quit but do not write the file to disk

vimtutor – built in tutorial on using vim, from beginner to advanced
h,j,k,l keys helps you move around vim as well as the arrow keys

helpful website
Vimgolf.com

Shell scripting

#!/bin/bash = the shebang, the first line in a bash script that tells bash what scripting language is being used
# - comment in script

```
if [something]
the
do this thing
else
do this other thing instead
fi
```

chmod +x <file name> - helps make the document executable

hardware

cat/proc/cpuinfo = view the cpuinfo file to gether details on processor
free = view ram stat for system
free -m = shw output in mb
free -g = show output in gb
dmidecode = show details about motherboard, bios, processor and ram on a system
lsblk – view all block devices(such as hard disks) attached to the system
df – view free disk space on a hard disk
df -h = show output in human readable format
top = show stat on processor, ram, and running processes

processes

ps = list the processes currently running on a system
ps -u = list the processes running on the system
ps -e = list all processes for a specific user
ps -H = list all processes with indented output, showing the hierarchy
ps -f = full format listing, including command arguments
top = show statistics on the processor, ram, and running processes

man proc
man signal

Networking in linux

ifconfig or ip addr show
– view ip add

Ip route show or route or netstat -r
- view default route(gateway)

ping
- checks to see if the network is connected

linux user

who = see who is logged into the system
w = see who is logged into the system with more details
id = view user and group ID’s of a specified user, command by itself shows ID’s of current user
sudo = execute a command as another user
etc/password = primary configuration file for all users on a system
etc/group = primary configuration file for groups

Adding users

groupadd = add a new group to the system

useradd (name)= ad new users to a system

passwd = set a password for a user

/*etc*/default/useradd = defines some default behavior for the useradd command

/*etc*/skel = contents within this directory are copied into home directories of newly created users

/etc/shadow = primary configuration file for all encrypted passwords for usres on a system

id – shows you what you are

1-200 = system users for specific redhat processes

201-999 = system users that use system processes but dont own anything on the system

usermod -u (number) (name)

- will specify user

userdel -r (name)

- will delete the user

managing user passwords

vim login.defs

- where you mark how long till the password is going to expire

usermod -s /*sbin*/nologin

- the user will not be able to log into the shell but it will have an active username and passwords

unix epoc

- the start of the calender

chage -l (username)

- shows you the time in which the password will expire

chage -E (date) (username)

- will show you when the account expires

chage -M (day #)

- *this I will change when you get the warning to change your password*

chage -d 0 (username)

- *will change it to where the password must be changed*

chage -I 5 (username)

- *will make to where the account will lock after 5 days of after asking for password change (password inactive)*

chage -W 5(username)

- *will change the number of days of warning before password expires*

vim /etc/login.def

- *will allow you to modify these processes further*

password directory

etc/shadow

- *shows you the last password file change*

usermod -s *sbin/nologin (useraccount)*

- *makes user not be able to log into the shell but allows them to log into mail server*

chmod
sybolic permission
r = read
w = write
x = execute permission
- = no permission

octal permissions
4 = read permission
2 = write permission
1 = execute permission
0 = no permission

chown = change ownership of a file or directory
chmod – change mode of a file or directory, effecting the permissions
ex: chmod o-r <filename>

symbolic links
ln = create a link to files or directories
ln -s = create a symbolic link
ln -s myfile mylink
unlink -remove a link from a file or directory

special files and folders, and the sticky bit

/*var*/tmp – contains files that do not get deleted on reboot
temp – contains temporary files that do get deleted on reboot
sticky bit – a permission that only allows users that create their own files and folder can delete theirs and not another users
applying sticky bit
chmod o+t /path/to/directory
chmod 1777/path/to/directory

Create, Delete, and Modify Local Groups and Group Memberships

```
cat groups
```
- shows all of the groups and the members of that group
```
usermod -aG (file) (user)
```
- gives you permissions to that file
```
groupmod -n class5 class1
```
-renames the primary group to “class 5”
```
groupdel (file name)
```
- deletes the file group

Congiure a system to use an existing Authentication service for user and group information: using realmd

```
Yum install -y realmd
```

```
realm discover ad.linuxacademy.com
```
- discovers the realm then ask you to install the required packages

```
realm update -y
```

```
yum install -y oddjob oddjob-mkhomedir sssd adcli samba-common
```

```
realm discover ad.linuxacademy.com
```

```
vim /etc/ssh/sshd_config
```
- change all kurborse to yes
```
systemctl restart sshd
```

```
 cat /etc/passwd | grep test
```

now you can ssh into it

```
authconfig-gtk
```
- this is how you do the set up with a gui interface

Red Hat Certified systems administrator

cal – brings up the calender
date – brings up the date
daemon - a type of program on Unix-like operating systems that runs unobtrusively in the background, rather than under the direct control of a user, waiting to be activated by the occurance of a specific event or condition.
Sftp – is very important in transferring files to a remote user

umask
- checks permissions

Need more help with
List, set and change standard UGO/RWX Permissions
List, set and change standard UGO/RWX permissions: unmask

important
login shell commands
su -
su -l
su –login

mtr – better way of pinging the network

star -c
- creates a new archive
star -c -f =(filename.tar) (directory and components)

cd /*usr*/share/doc
- holds the information for info and man pages

Boot, reboot, and shutdown a system

init 0 – shutsdown system
init 6 – restarts the system

shutdown -r
- reboot

shutdown -p
- poweroff

shutdown -c
- cancels process

shutdown -h
-shutdown system without reboot

example:
shutdown -r +5 system going down in 5 minutes
- reboots the system in 5 minutes and says its doing that
shutdown -r 00:00

cd /*usr/lib*
- *has a files you can cd into*
such as systemd

Identify CPU/ Memory Intensive Process/ adjust Process Priority and kill processes

ps aux
– list all the processes that are running on the computer
pgrep (name)
- shows all the processes running with that name
pgrep (name) -l
- list out all of the processes and the names associated with them
pgrep -u user -l
- list out all the processes the user has running

example: pgrep -u user -l vim
Just returns the vim process

pgrep -v -u root -l
- list out all the processes that are not owned by the root user

pkill (name)
- kills the processes

kill -l
- list out all the ones in which you can kill.

Pkill -t (tty)
example: pkill -t pts/1
- kills a users process

df -h

renice -n (number) (number associated with process)

uptime
- checks to see how long the process has been running
example:

```
07:10:13 up 6 days, 15:08,  2 users,  load average: 1.27, 1.43, 2.07
                                                    60sec, 5min, 10min
```

cat /*proc*/cpuinfo
- shows you process information of the kernel

Locate and interpret System Log files and journals

first cd into log
cd /*var/log*

tail -f secure
- will list out all users who are trying to access the system in real time

man systemd-journald
- brings up the man pages

journalctl
- shows you all the information into it

journalctl -f
- keeps listening to the information coming in

systemctl status (name of process)
- this will show you the status of stuff listed

journalctl -xn
- shows more information about the file

journalctl   --since=yesterday
- shows you the processes since yesterday

systemd-analyze
- shows information about our boot process

systemd-analyze blame
- shows how long it took each configuration files to load

tail /*var/log/secure*

Start/stop Virtual Machines

Virsh help
- gives you a list of all the commands
list –all
- displays all of your running virtual machines and stopped ones
list
- displays only the running ones

shutdown (vmname)
- shuts it down
start (vmname)
- starts the vm

Another way of doing this
Applications > system tools > Virtual machine manager >right click it and shutdown > shutdown

Start/stop and check the status of network services

systemctl list-units | grep network
- grabs the info for network
systemctl restart network.service
- restarts the networking
systemctl is-enabled network
- shows if it is enabled
systemctl start (name)
- starts it
systemctl stop (name)
-stops it
systemctl enable (service name)
- enables that service at run time
systemctl status network
- checks the status of the network

Securely Transferring files between systems

scp
sftp
port 22 - is how we send encrypted traffic

scp (filename) user@(ipaddress):~/
ex: scp myfile user@54.85.182.116
- transfers file over scp/ you can also put those files in directories

sftp user@(ipaddress)
- transfets file over sftp

when connected type
?
- this list out all of the commands

get (filename)
- downloads the file to your system

put (file name)
- copying file by uploading to the other user

quit – escapes out of sftp

List, Create and Delete Partitions on MBR and GPT Disks

fdisk
- is the tool used to manage MBR based partitions
example: fdisk /*dev*/xvdf

mkfs
- makes a file system
example: mkfs -t xfs xvdf

df -h
- shows you all of the mounted disks currently

blkid
- shows the available block storage devices that are attached to the system

cd /mnt
- best to mount stuff here in the mnt directory

parted

gdisk
- is a text-mode menu-driven program for creation and manipulation of partition tables

Create and Remove Physical Volumes, Assign Physical Volumes to Volume Groups and Create and Delete Logical Volumes

# using the LVM Management on Redhat Enterprise 7 linux
In the lvm-with-redhat pdf

Configure Networking and Hostname Resolution Statically or Dynamically: Troubleshooting

Ipv4:
ping <addr>
ip addr show
Ipv6:
ping6<addr>
tracepath6<addr>
traceroute6<addr>
ss
- allows us to look at listening ports and established connections

ss -a
- show all listening and established connections

ss -t
- allows person to see tcp sockets

ss -u
- allows person to see udp sockets

ss -n
- allows user to see the port number

ip -s
- shows statistic based information regarding this

nmcli dev status
- google it

nmcli dev show
- will show our devices

nmcli con show
- shows all of our connections

nmcli con add (name)
- brings up a new connection

nm-connection-editor
- pops up a gui interface for managing our devices

nmtui
- gives us a text based gui that helps configure networking

cd /*etc/sysconfig/network-scripts/*
- connection configurations are located here

nmcli con add con-name “mycon” autoconnect yes type ethernet ifname eth1 ip4 <ipaddress> gw4

nmcli con down “mycon-static”
nmcli con up “mycon”
nmcli con mod “mycon” connection.autoconnect yes

nmcli con del “mycon-static”
- deletes it

hostnamectl set-hostname (name.com)
- creates a host name
exec bash
- adds it locally
hostnamectl status
- checks to see where it is located
getent hosts
- will display the stuff for that

Schedule Tasks Using at and cron

Getting it set up

Yum install at

systemctl enable atd

systemctl start atd

at now +5 min
- setting a reboot in 5 minutes
example:
>>at 12:00am
>>reboot then ctrl-d
atq
- shows all of the scheduled task
atrm (number)
- removes a specific scheduled task
journalctl -xn
- displays important stuff

create your own
vim uptimelog
>> script
#!/bin/bash
logger “The systems current uptime is $(uptime)”
>> esc
>> wq!
Modify umask permissions
>> chmod +x uptimelog
>> ./uptimelog
>> journal -xn

Cron is a task time based organizer it helps set times for updates and such
system cron is already assigned into your etc directory

cron tab – shows you all you need to know

ls | grep cron
cron.d
- any programs that have their own cron schedule

anacrontab
- allows you to run a command that has not been run for several days

Install Red Hat Enterprise Linux Automatically Using Kickstart

The **Red Hat Kickstart** installation method is used primarily (but not exclusively) by the Red Hat Enterprise Linux operating system to automatically perform unattended operating system installation and configuration.

Vim anaconda-ks.cfg
- you can use this as a starting point

yum install system-config-kickstart

on console
system-config-kickstart
- to pull up the gui

Configure a Physical Machine to Host Virtual Guests

Qemu
- makes the virtualization and also the cpu to hold and host guest
libvirt – helps us interact with qemu and kvm

```
    yum install virt-manager qemu-kvm qemu-img
    yum install libvirt libvirt-python python-virtinst libvirt-client
    Enable libvirt – if you dont you will not be able to enable a vm to boot when your system boot
    systemctl enable libvirtd
    systemctl start libvirtd
```

virsh
- command line for managing virtualization

Install Red Hat Enterprise Linux Systems as Virtual Guests

– have to have physical machine

>> Before you begin have an iso file for the operating system you want to use
>> go to applications
>> system tools
>> virtual machine manager
>> click file – new virtual machine
>> forward
>> browse / browse local – find iso and install

you will not have a network the first time you start
>> ip addr
>> ls /*sys*/class/net
>> cd *etc.sysconfig*network-scripts/
>> ls
>> cat ifcfg-ens3
>> nmcli con up ens3
>> nmcli con mod “ens3” connection.autoconnect yes
– this allows the device to activate on boot

Configure Systems to Launch Virtual Machines at Boot

>> virsh
>> autostart (vm-name)
– starts a vm at runtime

Configure a System to Use Time Services

timedatectl
- provides information about the system configuration time
timedatectl chronyd
timedatectl list-timezones
- lists all the available timezones
tzselect
- shows you a list of places
timedatectl set-timezone (timezone location)
- sets the timezone
timedatectl set-time (time of day)
- allows your to set your current time
systemctl status chronyd

chronyc sources -v
- shows the actual servers that it is communicating with
- google about stratum

chronyc tracking

Now to change the time services
>> cd /etc
>> vim chrony.conf
>> modify the servers
>> systemctl restart chronyd
>> chronyc sources -v

Install and Update Software Packages from Red Hat Network, a Remote Repository or the Local File System: Managing Repositories

cd /yum.repos.d

- if you ls this it will show you a list of repositories

yum repolist all
-shows you all enabled and disabled repos

to disable a repo
>> you can vim into the repo and change enabled to = 0
or
>> yum-config-manage --disable (repo id)

Install and Update Software Packages from Red Hat Network, a Remote Repository or the Local File System: Configuring a Local Repository: Configure the GPG Key

- Download gpg
- learn how to configure it
- why it is important

```
Yum-config-manager --add-repo http://dl.fedoraproject.org/epel/7/x86_64/
```

dl.fedoraproject.org/pub/epel/
- goes to the epel list so you can download the versions of fedora

```
>> cd etc
>> cd pki
>> cd rpm-gpg
>> ls
>> wget http://dl.fedoraproject.org/pub/epel/RPM-GPG-KEY-EPEL-7
>> etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
>> cd etcyum.repos.d
>> ls
>> vim dl.fedoraproject.org_pub_epel_7_x86_64_.repo

>>gpgcheck=1

>>gpgkey=file:/// etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7

>>:wq!

>> yum install python-pip
```

Update the Kernel Package Appropriately to Ensure a Bootable System

Uname
-prints out system information
yum upgrade
- will update the version of the kernel
yum clean all
- cleans up the repos
yumdownloader kernel
- will download the rpm files for the kernel
rpm ivh kernel-3.10.0-299.1.2.el7.x86_64.rpm
yum install linux-firmware
- will update the firmware

cd /boot

Configure Firewall Settings Using firewall-config, firewall-cmd, or iptables

Yum install firewalld firewall-config

systemctl start firewalld

systemctl enable firewalld

firewalld can make runtime changes which rules do not survive a systemctl reboot

firewall-cmd --get-zones
- commandline to open it up

firewall-cmd --get-default-zone

man firewalld
- shows firewall options

firewall-cmd --list-all
- views the rules that are inside of the firewall

firewall-cmd --zone=home --list-all
- changes from default to home

firewall-cmd --zone=home --add-source=(ip address)
- adds the ipaddress range

firewall-cmd --reload
- is created as a runtime change / you have to do this every time you make a firewall change

firewall-cmd –zone=public –add-port=80/tcp

- adds port 80 non permanent
firewall-cmd –zone=public –add-port=80/tcp –permanent

firewall-config
→ really important gui based interface for the firewall

firewall-cmd –panic-on
→ locks down everything and makes it unusable

Configure Key-Based Authentication for SSH

ssh-keygen -t dsa
ssh-keygen -t rsa

→ enter a passphrase

navigate into the ssh file
→ id_rsa = private key

→ id_rsa.pub = public key
permissions on private key = 600

permissions on a public key =644

Introduction to SELinux

https://www.centos.org/docs/5/html/Deployment_Guide-en-US/ch-selinux.html

getenforce
→ Shows that selinux is turned on

setenforce 0
→ puts this into permissive mode

setenforce 1
→ puts it back into enforcing mode

```
cd /etc/selinux
vim config
```