Linux + Certification

Introduction:

* LXO-103 == LPIC 101-400
* LXO-104 ==LPIC 102-400
* 500/800
* 60 questions
* 90 minutes and is valid for 5 years
* $200 to sit for exam.
* Watch video
* Take notes
* Re-watch video and pause at each step
* Read documentation for anything you don’t understand

**About Linux and Installation:**

What is linux?

* Unix-like computer OS for free and open source distribution.
* Originally made for x86 architecture
* Largest install base now with general purpose OS
* Some want to name it GNU/Linux or Linux
* Components
  + **Boot loader** – software that manages boot process
  + **Kernel** – core of OS. Manages CPU, memory
  + **Daemons** – processes that start during the boot that support the system
  + **The Shell** – access to the CLi and control the whole system.
  + **Graphical Server** – X server, sub system which displays graphics
  + **Desktop environment** – what the user interacts with.
  + Applications
* Free
* Stable
* Secure
* Open Source

Linux Distribution

* Collection of software
* **Package management system** – help install, upgrade, and remove software
* Keeps your server up to date
* Redhat
  + CentOS (community version of RedHat)
  + Fedora
* Debian
  + Ubuntu
  + Mint
* SuSE
* Gentoo
* Arch
* Slackwareß

How can we run linux?

* Your own PC
* Someone else’s PC, Shared hosting, Cloud Provider (AWS, GCP, Azure)
* Virtualization

Installing Ubuntu & CentOS

* Not good to put a boot loader in the master boot record. Okay on VM.
* Enabling bridge network mode
  + CentOS
    - Set adapter to bridged
    - Cd /etc/sysconfig/network-scripts
    - Sudo vi ifcfg-enp0s3 (a=append)(:wq) (ESC)
    - Onboot = yes
    - Ip addr is to see IP address
    - Usermod -a -G wheel root
  + Ubuntu
    - Just set adapter to bridged
* **Sudo loadkeys us** – shows keys local to the region.

**System Architecture:**

Boot the System

* **\*BIOS (Basic Input Output System**) – firmware that provides hardware initialization of the boot up of your system. (Delete or f8 to access)
* **MIT (Motherboard Intelligent Tweaker)** – allows you to provide tweaks on the CPU and overclocking.
* **System Settings** – Time and Date
* **Boot Option priorities** – let us select which devices are booted from
  + **UEFI (Unified Extensive Firmware Intelligence)**– UEFI -> EFI Boot Loader -> Kernel -> Operating System
    - Has its own file system
    - Each should have its own boot loader
  + **Legacy booting** – BIOS -> MBR(Master Boot Record) -> BootLoader -> Kernel -> OS
  + **GRUB (Grand Unedified Boot Loader)**
    - Linux – specifies the kernel that will be used to boot the kernel
    - **Initrd** – specifies which ram pkg to use in the boot up process.
    - Not good to install on the master drive with another boot loader
* **MBR (Master Boot Loader)** – identifies where and why a OS is located to boot from.
* **Kernel** – lowest level of replicable software that interacts with the hardware.
* **Ps aux | head** – goes over the processes listed
  + **Sbin/init** – first process ran on linux
* **Sysvinit** – first commercial version of linux available
* **Systemd** – low memory and perform’s better than sysvinit. Easier to read than sysvinit. API
* **Upstart** – Ubuntu developers made by it.
* **Dmesg/dmesg -T(changes date)** – shows kernel messages when the system was booted

Determine and configure hardware settings

* **Udev** – device manager for the kernel of your system
  + Low level access to the linux device tree
  + Handles user space events
    - Loading firmware
  + Provided by a temporary filesystem (tmpfs)
  + Mounted to /dev on startup
  + Rules.d allows you to write custom rules
* **Dbus** – inter-process communication mechanism that allows processes to talk to each other
  + Provides a high-level object oriented programming interface
* **Sysfs** – virtual filesystem that presents various information about kernel subsystems
  + Mounted to /sys
  + Hardware devices
  + Drivers
* **Procfs** – similar to sysfs but presents information about processes and information about system information.
  + Mounted to /proc
  + Can be used to interface with the kernel and change parameters on the fly
  + **Cat cmdline**
  + **Lsmod | less –** page by page version of module
  + **Modprobe –** enable a mod
    - **Rmmod –** remove a mode
  + **Lspci** – shows devices connected to the computer

Run levels, boot targets and how to shutdown and reboot system

* **Run level (inittab)** – number between 1-9 and determines which scripts are used based on the run level
  + 0 – halt or shut down
  + 1 – single user mode
  + 2 – multi-user mode without networking
  + 3 – normal boot
  + 4 – Unused/customizable
  + 5 – Run level 3 + GUI display manager
  + 6 – reboot
  + rcS.d – is run level used during every level
  + **init or telinit** – used to change the run level
  + **man telinit** – manual command
  + **wall** – writes to the shell as to everyone that is logged in.
  + **systemctl** – is a way to control systemd devices or targets
    - can be used like init telinit
* **Scripts** – little programs held on the system that can be used.
* System run level
  + /etc/system/system
  + Package scripts
    - /user/lib/systemd/system
  + System packages take power over packaged scripts
  + **Targets** – are like run levels. Named instead of numbers

**Linux Installation and Package Management:**

Design a hard disk layout

* /usr – user binaries where packages are installed
* /home – user files
* /boot – where the boot loader is found
* /etc
* /var – variable files
* /tmp – unique storage area where everyone can write too
* Partitioning
  + Separation of files
  + Allows dual booting
  + Divide storage into multiple pieces
  + Data organizations
  + System protection
  + Helps ensure data doesn’t overflow into other directories.
  + Swap partition can be used to free up more memory
  + **LVM** – Logical Volume Manager – split physical partitions into pools

Install a boot manager

* Common boot loaders
  + LILO
  + GRUB Legacy
  + GRUB 2 **(grub-probe –version)(grub-install –version)**
    - **Less grub.cfg command**
    - **Sudo vi /etc/config/grub.d** 
      * need to run sudo **update-grub** to update the changes made.
      * Quiet mode will eliminate a lot of messages during bootup
    - Find where a script is located – **which update-grub**
    - Can have systemd boot from another target – like **vg-root ro systemd.unit=rescue.target**
    - **Sudo grub-install /dev/sda**

Manage Shared libraries

* Shared reuseable pieces of code
* **Static linking** – application included in a library
* **Dynamic linking** – two applications use one single library
* **/etc/ld.so.conf** is where the shared libraries would be
* **libc.conf** is in C programming language
* **man ldconfig** does the dynamic linking in a directory
* **environment variables –** customized pieces of text that are specific to your shell
  + **export LD\_LIBRARY\_PATH=/home/nick/lib/**
  + **echo $LD\_LIBRARY\_PATH**
  + **LDD –** prints shared object dependencies

Debian package management

* Dpkg
  + Debian Package Manager
  + Install software, ugrade, remove, low-level tool
  + **Dpkg –l | less**
  + **Wget to pull from a url**
  + **Dpkg –purge dlocate**
  + **Doesn’t find dependencies and doesn’t automatically install them**
* Apt
  + Advanced packaging Tool
  + High level tool
  + Install, upgrade, remove packages, upgrade entire system and handles package dependencies.
  + Uses online repositories
  + Installs dependencies.
  + **Apt-cache depends apache2 | less** – search for packages and looks at dependencies of a package.
  + **Apt-cache search nginx**
  + **Apt-get upgrade** – can’t upgrade all packages
  + **Apt-get dist-upgrade** – upgrades everything and deletes everything that it won’t use
  + **Sudo apt autoremove**

Using RPM and YUM package management

* **RPM (RedHat Package Manager) similar to dpkg**
  + Rpm command
  + Low level tool
  + Install software, remove, update.
    - Cpio files which are files that can be combined into on main location
    - Rpm2cpio wget-1.14-13.el7.x86\_64.rpm | cpio –idvm – good way to look in an rpm
  + Rpm –qf (query format) /etc/protocols
  + Sudo rpm –verify setup – queries everyfile to see if there in the directory
    - Don’t want a response on this which is good.
  + Sudo rpm –Va which is simliar
* **YUM (Yellowdog Updater, Modified) similar to apt-get**
  + Replaced YUP, Yellowdog Updater
  + Utilizes online repos
  + Manages dependencies
  + Conf file /etc/yum.conf
  + Sources url list /etc/yum.repos.d
  + Use head command to look at the top of the file
  + Sudo yum install –downloadonly –downloaddir=/tmp wget
    - Yumdownloader install doesn’t exist anymore

**GNU and UNIX commands:**

**Shell** – program in which you type in linux. command line interpreter. Bash shell, sh, sdsh, kssh

* **Sh** shell doesn’t show directory
* **Pwd** – shows working directory.
* **Echo $HISTFILE –** can show if it uses a different environment variable history
* . files are hidden
* **ls –a** shows all files even hidden ones. Can hide director too.
* **History** – shows number commands by order
* **Export HISTFILE=/home/guru/**.test – changes the history file to change where it writes too.
* **Set** – set values of shell options or positional parameters.
* **Man man** – view details about the man guide.
  + **Man –k (value)** – shows and searchs for commands related to it.
  + **Man 7 time** – shows different values for those commands.
  + **Manpath** – shows the directory of man.
* **Uname** - prints out system information

**Environment variables** – affect the way running processes behave on the terminal

* Temporary only during the time that your logged into the shell
* To keep them constant, have a script that executes every time your login.
  + Can be done by changing the .bashrc script

Process Text streams using filters

* Won’t save to files unless told too
* **Cat** file1.txt
  + **Cat /etc/passwd** shows all those that have passwords
* **Cut** – remove sections from each line of files
  + **Cut –c3** file1.txt
  + **Cut –c3-**5 which includes a range.
  + **Cut –d: -f1** - separate fields but the : -. F1 can choose which can be filtered out of the list
* **Expand** – convert tabs to 8 spaces
  + **^I** means tab
  + **Cat –vet file** – shows the spaces between the characters
  + **$** means end of line
* **unexpand** - convert 8 spaces back to tab
  + **unexpand –a**
* **wc** – tell new line count, word count, byte count for file
  + **wc** –l is just the line count.
  + **Fmt** – can split the lines into even lines
* **Head** – see the top of the page
  + 10 lines each
  + Head –n 2 /etc – changes lines in list
* **Tail** – see the bottom of a page
  + 10 lines each
* **Join** – join two files together that share a common field.
* **Less** – shows page by page version of file
* **Nl** – adds line numbers to a file for you.
* **Od** – dump files in octal formats
* **Paste** – does the same as the join command but combines everyfield.
* **Pr** – converts text files for printing.
* **Sed** – stream editor which can perform text
  + Sed – e – regular expressions
* **Sort** – sorts a file apabetically
* Split – splits files into multiple pieces with 1000 file limits.
* Tr – translate or delete characters.
* Unique – show syou matching json texts back to you.

Use Streams, pipes and redirects

* Streams
  + **STDIN - 0** (file describer) – inpute to programs via key strokes
    - 0 before > shows which stream should be done.
    - < can sort the content with sort.
    - << which manually allows us to input characters
      * sort << END
  + **STDOUT – 1** – print text characters to the terminal (Default)
    - > redirect to file name after it
    - >> is to append more about the file.
    - > overwrites a file
  + **STDERR – 2** - print text characters to the terminal
  + **&** copies to a file describer. Position matters
    - **ls > x 2>&1 > y**
  + **tr –s ‘ ‘ < ls.txt > output.txt**
* **Pipes** – allows us to stream an output from one command to an input of another command
  + **Cat input.txt | sort**
* **Xargs** takes input line by line and puts it input in the command thereafter
  + **Find / -name blah.txt | xargs rm**
* Redirect standard output but watch output at the same time.
  + **Ls | tee lsout.txt**

Create, monitor and kill processes

* **Ping** – tests how long a message takes to get to a computer and back again.
  + **Control z**
  + **Bg** can run a program in the background
    - **&** puts it also in the same background.
  + **Fg** can bring a command to the forfront.
* **#!/bin/bash** – tells the shell what the command is in.
  + **source script.sh** – runs a command without starting a new shell
  + **. script sh**
* **nohop –** shows the command but will run the command in the background.
* **Jobs –** shows jobs in the background.
* **Ps –** shows running processes running on the computer.
  + **Ps a –** shows everything to the shell and computer.
  + **Ps x –** shows commands by all users
  + **Ps aux –** shows in a user name formatted.
  + **Kill (process id) –** kills the process
    - Can send signals to the process.
    - SIGINT – CTRL + C – interrupt signal
    - SIGKILL – kill signal
      * **Killall ping** – can be dangerous – exact matching.
      * **Pkill** terminates by command name.
    - SIGSTOP – pause signal – CTRL + Z
    - SIGTERM – termination signal
      * **Kill –TERM 2384**
  + **Top –** shows continuing information being ran on your computer.
    - **Shift p –** sorting by CPU
    - **Uptime –** shows uptime of the computer.
    - **Free –** gives memory information about the computer.
  + **Screen –** run process and screens at the same time in the same window. Persists after.
    - **Control a d –** detach from the screen.
    - **Screen –r –** reattach screen.
    - **Control a c –** starts a new screen in the same shell and toggle back and forth.

Modify process execution priorities

* **Nice value** – determines how much CPU time a process will get.
  + -20 – Highest priority
  + 19 – lowest priority
  + 0 – default nice value
  + **nice** – execute a value with a different nice value than default
  + (-) is with a –
  + **renice** – change a nice value again.

Regular expressions

* Sequence of symbols and characters
* Express a string or pattern
* Searched within text
* Regex
* Symbols
  + | - or
    - Eg gray|grey
  + () – grouping
    - Eg gr(a|e)y
  + $ - end of a line
  + ^ start of a line
  + . – one character
  + ? – 0 or 1 occurrence of the preceding element
    - Colou?r matches color or colour
  + \* - 0 or more occurrences of the preceding element
    - Eg. Ab\*c matches ac, abc, abbc, etc
  + + - 1 or more occurrences of the preceding element
    - Eg. Ab+c matches abc, abbc, abbc, but not ac
  + {n} – the preceding item is matched exactly n times
  + {min,} – the preceding item is matched min or more times
  + {min,max} – the preceding item is matched at least min times but not more than max times
* **Grep** – globally search for a regular expression and print
  + Uses symbols and regular expressions to search plain text
  + Can search for opposites too
  + **Grep**
  + **Egrep (grep –E)** extended search characters
  + **Fgrep** – searches the entire string without quantifiers
  + **Sed** – can use quantifiers with search and replace

Perform basic file editing operations using vi (VIM improved)

* Editing mode
  + Can edit/delete text
  + I – insert mode
  + Esc gets you out of editing
  + A – types in front of cursor
  + o – lets you start typing after the cursor
  + O – lets you start typing before the cursor
  + Shift + ZZ lets you save and quit.
  + :q! changes aren’t saved and exit
  + :w – saves the file without quitting the file.
* Command mode
  + Move around the file and manipulate the text
  + Can type things new in the file.
  + Can delete text
  + Can use letter keys to move around like with hjkl
  + Can search the text
  + n – repeats the last command
  + ?init searches back in the file for the text
  + dd – delete line you are completely on
    - 3dd – deletes three lines
  + P – paste the entire line after the line your own
  + Shift + P inserts it above the line
  + Yy – copies the line.
  + X deletes characters
  + :3 – jump around on lines

**Devices, FileSystems, and Filesystem hierarchy:**

Create partitions and filesystems

* **df** – shows disk partition and disk usage.
  + **Df** –h – readable disk sizes.
* **Lsblk** – shows us what is going on with partition based on the dev mapper with LVM
* **Fdisk** (m shows all of what can be done with fdisk). Gives us a way to create partitions on a hard disk.
* **Parted** – another way to create partitions.
  + Mkpart creates a new partition
* **MBR (old standard)**
  + Allows 4 primary partitions
  + Max disk is 2 TB. Have to put anything above 4 partitions in an extended mode.
* **GPT**  - let’s us use much bigger disk sizes.
  + Associated with UEFI
  + Basically, unlimited partitions
* Filesystems
  + **Ext2** – default and replaced ext
    - Ext only allowed certain amount of characters. Partition size was to small
  + **Ext3** – allows journaling
    - Journaling assists in recovery lost data.
  + **Ext4** – allows huge files and filesystems. (16 TB starting)
  + **Reiserfs** – resizable, journaling filesystem
  + B**trfs** – b-tree filesystem, supports snapshots, checksums, and pooling
    - Online defragmentation and autodefrag mount
    - Online resizing
    - Online filesystem checks.
  + **Mkfs** – makes a filesystem
    - If you don’t specify filesystem, ext2 will be default.
  + **Mount** – shows all the filesystems that are mounted.
  + **Mkswap** – wipes the drive and creates a swap space
  + **Swapon** – provides a swap disk drive.
    - Free shows disk space.

Maintain the integrity of filesystems

* **Inode** – data structure which describes a file or directory
  + Stores attributes
    - Last time of change/access/modification
    - Owner/permissions data
  + Stores disk location
  + You **can** run out of inodes before running out of disk space
  + Directory of a list of names assigned to inodes
    - Contains entry for itself, its parent, and its children
  + **Ls – I** – shows us the inode for a particular file or directory
  + **Find . –inum 585 | xargs rm**
  + **Df –I –** shows inodes on root filesystem
  + **Du –h –** shows all files in your current directory and the amount of disk it is using.
    - **-s** gives summary
    - **du –**h –max-depth=1 / - shows disk usage in root file system. Won’t show files
  + **fsck** – check and repair errors that may be found with the file system. Don’t run integrity check on a disk when it is mounted.
    - Reboot to rescue mode when working on the root drive.
    - **Fsck tab tab** shows all file system for it and the versions.
    - **Fsck** is a wrapper.
  + Super block – segment of file system that stores metadata about itself
    - **Dumpe2fs –h**
    - **Tune2fs –L –** adjust tunable filesystem variables. Lets us set the filesystem name.
    - **Max-mount-counts** – number of times filesystem is mounted

Control mounting and unmounting of filesystems

* **Mount** – can see mounted file systems
  + Cat /pro/self/mounts
  + **/etc/fstab**
    - dump means to create a backup of the system.
    - Pass is used for filesystem checks. 1 is for root disk. 2 is for other checks
    - **Mount –a** mounts everything in fstab without noauto default.
  + **Fuser** – shows us which process is taking the directory
    - **Ps aufx**

Manage disk quotas

* Etc/fstab edited
  + Defaults,usrquota – enable user quote on hard disk
* Apt-get install quota with inline command tools
  + **Quotacheck** – initialize the disk to start quotas.
  + **Edquota** – edit a quota for a user.
  + Quotaon /mnt/photos enables quotas
  + **Quota** – v shows the quota for the partition/hard disk
  + **Dd if=/dev/zero** of=test.file bs=1 count=5120 – dev zero outputs null characters in a file
  + 7 day grace period to go over the quota.
  + **Repquota** – gives a report on the filesystem

Manage file permissions and ownership

* **Ls –la, ls –lad** (shows permissions of the folder not all content)
  + Rwx rw- r—
    - Rwx is the user
    - Rw is the group
    - R—is for others
    - R – read
    - W – write
    - X - execute
  + D – references a directory
  + L – means link
  + – means file type
* **cmod**
  + u+rwx, g+wx, o+r
  + r = 4, w=2, x=1, -=0
    - add the numbers together and then apply 764
    - 0764 the 0 is implied as that is the sticky bit
  + rwx = 421, rw- = 420, r-x = 401, -r- = 040, -wx = 021, -w- = 020, --x = 001, --- = 000
* **setuid/setgid**
  + used to tell the system to run as the program owner with the owner permission even with you not being the owner.
  + **+s**
* **stick bit**
  + used on folders. Prevents folder deletion even if a user has persmission
  + **+t**
  + S not set, s is set
* **Chown** – used to change the owner of a file
  + **Chown username:group filename**
  + **Chown –R username \***
* **Chgrp** – used to change the group of a file
* **Umask (0002 by default)** shows the default of permissions are not set on a files. List what you want to suppress from default files and default folders.

Create and change hard and symbolic links

* File is only removed when all links to an inode are removed
* **Hard link –** link to another file indoe and can occur on the same filesystem.
  + **Ln –** link command.Ln –s is for a symbolic link.
    - **Ls –**li shows the inode tied to the file.
* **Soft link –** link to another file but can occur across file systems.
  + Similar to a shortcut in windows.

Find system files and place files in the correct location

* **Type** – tells if the command is built in or a shell command
* **Whereis** – locates binary, source, and manual page if possible for a command.
* **Locate** - builds a database on your filesystem and then being able to find things on the filesystem.
  + **Updatedb** – updates current database on device.

**Shells, scripting, and data management:**

Customize and use the shell environment

* **Configuration Files**
  + Allows us to change how our shells work
  + Define path, environment variables
  + Set up aliases
  + System wide - /etc/bash.bashrc and /etc/profile
  + Local - ~/.bash\_profile, ~/.bashrc and ~/.profile
    - If [ -f ~/.bashrc]; then source ~/.bashrc fi
  + Action specific - ~/.bash\_login and ~/.bash\_logout
* Shell commands
  + **Set –o noclobber** – changes how the > works
  + **Set –o xtrace** – show you full command when you run it and all the commands
  + **Aliases** – allow you to customize your commands. Edit bashrc to include new ones.

Customize or write simple scripts

* **#!** – hash bang
  + first line of a script. Provides full path to the interpreter
* **if** – test a condition and then do something
  + if [] then fi
  + –gt for greater than
  + –lt for less than
* **else** – test condition, do something otherwise do something else.
* **For** – end of range do something
* **Seq** – print sequence of numbers. `` - used in script to execute a terminal command and print them.
* **Variable storage** – used to store information from a command.
  + $ current\_date = $(date**).** Arithmetic requires double brackets.
* **while –** test if condition is met and then execute commands.
* **Read –** capture user input. Doesn’t need a $ to read
* **Functions –** used to reuse code and you can pass variables to functions.
  + Functions must go at the top of the script.
* **Return values –** determine the outcome of a script or command
  + $echo$?. 1 means there was an error and didn’t exit gracefully
* ./nick.sh executes the script.

SQL data management

* SQL- Structured Query Language
  + INSERT – put a row of data into the DB
  + UPDATE – update a row of data in the DB
  + SELECT – choose a row of data in a DB based on condition
  + DELETE – delete a row of data
  + FROM – select the table youa re selecting data from
  + WHERE – defines the condition of your select command
  + GROUP BY – lets you group your data
  + ORDER BY – lets you order your data
  + JOIN – lets you join two tables which share a column of data.
* Commands
  + Mysql –u root –p
  + Show databases;
  + Use guru;
  + Describe guru – shows information about the table.

**User interfaces and desktop:**

Install and configure X11

* Steps to install X11
  + Sudo apt-get update
  + Sudo apt-get install xorg
  + **Startx** – starts the window configuration
  + Sudo apt-get install openbox (window manager)
  + Startx
  + Cd /etc/X11/Xsession.d/
* Xorg.conf.d has several configuration files in one specific directory. Also known as xorg.conf
* You can run multiple x servers by export DISPLAY=:0 and then xterm
* **Xwininfo** – click on the screen to gain information about it
  + **Xdpyinfo** provides a little more information about the x server
  + **Xhost** – used to restrict the x server on who can access it.
    - Xhost + adds access
    - Xhost – restricts access

Setup a display manager

* **Display manager** – graphical interface at the end of the boot process which prepares the desktop environment
  + XDM
  + GDM (Gnome)
  + KBM (KDE)
  + LightDM
    - Sudo apt-get install lightdm gnome
    - Generates users.conf
      * Can hide shellers, users and add permissions
    - Sudo service lightdm start
    - Ctl alt f 1 or ctl alt f 7 to go back or forward
    - Sudo systemctl set-default graphical.target – display manager get start by init system upon boot.

Accessibility

* Sudo systemctl lightdm start going into gnome
* **Gok** – gnome onscreen keyboard

**Manage user and group accounts and related system files:**

Manage user and group accounts and related system files

* Root:x:0:0:root:/root:/bin/bash
  + 0 is user and 0 is group. X is the encrypted password in the shadow file
  + /etc/shadow
    - Mail:!:17001:0:99999:7:::
    - ! means the password is invalid and needs reset.
    - 17001 – total number of days when the password was last changed
    - 0 - Next is the minimum number of days when the password needs reset.
    - 999999 – maximum number of days when the password expires.
    - 7 – number of days before password expired to be warned.
    - ::: - # of days when the account will be disabled and deleted
  + /etc/group
    - adm:x:4:syslog.guru
      * adm – name of the password
      * x – password of the group
      * 4 – group id
      * syslog.guru – list of users belong to the group
  + /etc/skel
    - skeleton directy for when users are added to the system
    - useradd – add users to without all the deletes
    - adduser – wrapper but also adds extra options when a new user is created
    - groupadd
    - addgroup
    - **chage** – changes number of days between password changes
    - **getent** – pull information from various files.

Automate System administration tasks by scheduling jobs

* **at** – read commands from standard input to run at a later time.
  + Control D to finish
  + **Atq** – to see a job
  + **Atrm 5** – to remove a job
  + Sends an email with the result
* **Batch** – acts at the same way and executes when the system load drops below a certain threshold 1.5.
  + runs in sequential order
  + at.deny shows the list of users that can use it
* **cron** – run tasks on a schedule. Automate system maintenance. Cron tab is a cron table
  + **crontab –e**
    - 0 18 \* \* \* /bin/ls
  + **Crontab –l**
  + **/etc/crontab** – show users that run the jobs
  + **Cron.daily** – crons run daily
  + **Cron.allow** – users that can use it
* Anacron – similar to cron but assumes the system isn’t always running. Runs with the system is back up
  + 1 5 cron.daily then the command

Localization and Internationlization

* **Date** 
  + Cat /etc/timezone – tells the system where you are located
  + /etc/localtime –la – sysmlink to another zone
  + Can copy or symlink it
* **Tzselect** – view installed time zones and see the time in other zone
* **Timedatectl** – change system clock and arguments
  + Set-ntp
* **Locales** – defines location and language for shell
  + **LC\_ALL** overwrites every other locales
  + C is the default locale
  + **POXIS** – portable operating system interface
  + **Iconv** – convert to different encodings
    - Iconv –f utf-8 –t ISO\_8859-15 <fromutf8.txt > toiso.txt

**Essential System Services:**

Maintain system time

* Keep your computers syncrhonised and aaccurate
* Helps with scheduled jobs Backups
* Helps caching and logging
* Helps identify when issues may occur
* Time-dependent applications require very accurate clocks
* **Date** +%s shows Unix time
* **Ntpdate –** used to query remote time server
  + **Ntpdate pool.ntp.org**
* **Ntpd –** runs a data on the computer to keep time accurate and current
  + **/etc/ntpd.conf**
  + Keeps running when installed
* **Hwclock** – utility to access the hardware clock or RTC. Independent of the system
  + keep it towards UTC time
  + **hwclock –systohc –localtime**
  + **/etc/adjtime** – shows where the time is stored for the hardware clock

System Logging

* **Syslog** – message logging standard. Separates message generator, storage, reports, analyzes
  + Each message has a facility code including software type and severity label
  + Used for System Management, Security auditing, general information, analysis, debugging messages
  + **Syslogd**
  + **Rsyslogd** – advanced version of syslog
  + **Syslogng** – has some features rsyslogd doesn’ thave.
  + **/etc/rsyslog.d**
    - Auth, authopriv.\* - \* shows severity level. , shows both logs. ; separates
    - .none won’t log anything
    - /etc/logrotate.conf
      * Rotating a log would finish a log file and go to a different file.
    - /etc/logrotate.d
    - Can also give logs to a different console window by \*.\* /bin/tty10
* **Journalctl** – used to query display messages from systemd journal
  + **Journalctl –b** – see messages from last boot
  + **Journalctl –since “2 days ago”**
  + **Journalctl –cron –u**
  + /etc/systemd/journald.conf
  + Control alt F2 goes to different console windows

Mail Transfer Agent (MTA) basics

* Running own email server on your OS
  + **Sendmail** – origin mail service. Backwards compatible
  + Postfix
    - Install postfix mutt
    - **Mutt** – email program to send and receive emails.
  + Qmail
  + Exim
* **Alias** – define lists or groups of users
  + **/etc/aliases**
  + **Newaliases –** rebuilds databases for where aliases are sent
  + **$** refreshes email
  + **H** shows the headers
* .**forward** – forward all emails somewhere else
* **Mailq** – show local mail queue before it is delivered
  + /var/mail/log

Manage printers and printing

* **CUPS** – Common UNIX Printing System
  + Standard for printing
  + Manage print jobs, queues, network printing, supports large range of printers, provides web-based configuration and administration tool
  + Apt-get install cups cups-bsd
  + /etc/cups/cups.conf
    - Add port for printer to listen on and authentication (DefaultEncryption IfRequested)
    - Allow all from restriction
* **Lpq** – shows printer queu
* **Lpr file1.txt** prints the file
* **Cupsdisable <Printer>** - print jobs will still be queued by not printed

**Networking fundamentals:**

Fundamentals of internet protocols

* **IP addresses** – number assigned to your computer
  + Allows other computers to communicate to you
  + Public and Private IP available
  + IPv4 and IPv6
    - 32 bit addresses with IPv4. Limit to 2^32 addresses. 4 octets
    - 128 bit addresses for IPv6
    - IPv6 – Neighborhood discovery
      * Improvements over ARP
      * Unreachability detection
      * Improves robustness of packet delivery
    - IPv4 ARP
      * Determines IP via DNS
      * Sends packet to computers asking for MAC address
      * Computer response from unknown IP with MAC address
  + Router -> Router -> Router -> Networks Router -> Your computer
  + Needs a network address and host address to reach the host on the network
  + **Subnet mask** – determine if a host is on a remote or local network
    - **CIDR** – method of splitting ip addresses that replace old class base system. How many bits are being used for the network address
    - Always subtract 2 to get the usable ip addresses
  + Private IP address
    - Not allocated to any specific organization
    - Anyone can use them
    - Requires NAT or proxy to connect to the internet
    - 10.0.0.0 - 10.255.255.255
    - 172.16.0.0 – 172.31.255.255
    - 192.168.0.0 – 192.168.255.255
  + Packet Types
    - TCP (Transmission control protocol) – connection orient protocol. High reliability applications
    - UDP (User Datagram Protocol) – connectionless. Stateless transaction
    - ICMP (Internet Control Message Protocol) – Used by network devices. Send error messages or operational information
  + Ports
    - 21 – FTP
    - 22 – SSH
    - 23 – Telnet
    - 25 – SMTP
    - 53 – DNS
    - 80 – HTTP
    - 110 – POP3
    - 143 – IMAP
    - 442 – HTTPS
    - /etc/services – shows all port numbers
    - < 1024 – privileged ports

Basic Network Configuration

* **Ifconfig** – see current network configurations
  + **Ifconfig –a**
  + **Local loopback –** refers the computer to itself
  + **Sudo inconfig enp0s3 new ip address**
    - Changes are temporary and will reset back.
* **Ifdown enp0s3** – brings the interface down
* **/etc/network/interfaces.d**
* **Ip a** to see all network interfaces
* **Route –** shows routes on the computer.
  + Sudo route add default gw router address
  + Route –n shows ip address than resolved
* **/etc/hostname** – shows the hostname of the computer
* **/etc/hosts** – give host names to ip addresses on the local computer
* **/etc/nsswitch.conf** – name service switch functionality

Basic Network Troubleshooting

* **Ifup and ifdown** to bring the ip up or down.
* **Host –** dns lookup on ip address or dns name.
* **Dig –** dns lookup utility which gives more information about the dns records
* **Netstat | less –** prints network connections and internet statistics
* **Ping –** sends icmp packet to the computer and receives a response
  + **Ping6** for IPv6
* **Traceroute** – traceroute to get from one connection to another connection
  + Install the package
  + Traceroute
* **Netcat** – read and write traffic over a network.
  + **Netcat** –l 12345
  + Can also send files over the network.

Configure client side DNS

* **/etc/resolv.conf** – usually the DNS file where the dns server would be resolved against
  + Won’t let you edit this file due to DHCP
  + **/etc/dhcp/dhclient.conf** – can be used to change the dns server
    - Supersede domain-name-servers 192.168.0.254, 8.8.8.8,8.8.4.4
  + **/etc/hosts**

**Security:**

Perform security administration tasks

* **Find –perm** – search for files where the permission bits are set
  + Sudo find/ -perm /u=s,g=s
* **Find – type** – search for files of a particular type
* **Passwd** – change your own password
* **Nmap** – port scanning application
  + Nmap localhost
* **Netstat –at** – shows active internet connections listening on the computer
* **Lsof** – list open files. Can be useful to find directories open when trying to unmount it
* **Who** – shows who is logged into the system
  + **W** – shows the same thing but more details
  + **Last** – same thing but shows history of who logged in
* **Visudo** – lets you edit the sudoers
  + **Sudo update-alternatives –config editor** – allows you to change the default editor
* **Id command** – shows ids and groups of the user
* **Ulimit** – allows us to set shell resource limit
* **Su** – change user or become super user
  + Can change to other users if you have their password

Setup Host Security

* **/etc/hosts.allow** – list host names or domains that can access the file
* **/etc/hosts.deny** – does the opposite and denies access. Can block everything with the ALL command. If it doesn’t match allow or deny it will be allowed
* **Nologin** – easy way to lock down your system if something is going on
  + **Sudo touch /etc/nologin**
* **Xinetd** – listen for incoming requests which launches programs for those requests. Supports logging
  + **/etc/xinetd.d**
* **Tcpd** – setup to monitor incoming requests. Triggered by xinetd. Tcpd logs the request and runs the appropriate program. Logs sent to syslog

Securing Data with Encryption

* **SSH** – remotely connect to another remote machine in a secure connection
  + **/etc/ssh/ssh\_config** – client side configuration file. Can change authentication methods
  + **/etc/ssh/sshd\_config** – change your own computer for connecting via SSH
  + Ssh keys
    - Passwordless authentication
    - Private key
    - Public Key
    - Ssh-keygen to generate public/private key pair
      * **/home/guru/.ssh/id\_rsa .ssh/authorized\_keys** so everyone can login but use the same access key
      * **Eval ‘ssh-agent –s’** – starts the agent
    - **Ssh-copy-id** – remote to a server and adds a copy of the key to the remote computer
* Ssh –X remotemachine name – enables x forwarding and sets display variable for you
* Gpg – allows you to encrypt data and knows the file came from you.
  + Sudo apt-get install rng-tools
  + Sudo Rngd –r /dev/urandom
  + Gpg –gen-key
  + Gpg –list-keys
  + Gpg –send-keys –keyserver hkp://keyserver.ubuntu.com 77574F15
  + Gpg –search-keys –keyserver hkp://keyserver.ubuntu.com ‘nickt@acloud.guru’
  + Gpg –keyserver hkp://keyserver.ubuntu.com –recv-key id
  + Gpg –encrypt –recipient keytest file1.txt
  + Ls –la .gnupg