Linux Academy

Study Guide

LFCSA

Contents

Related Courses

Linux Foundation Certified System Administrator

Need Help?

Linux Academy Community

... and you can always send in a support ticket on our website to talk to an instructor!

Essential Commands

Create and Edit Text Files

- `vi/vim`
  - » Full screen editors; `vi` is always available since it is a POSIX requirement, `vim` may need to be installed)
    - — `vi filename`
  - » Multiple modes:
    - — *Command mode* allows for navigation and the entering of commands using combinations of one or more letters; these can be prefixed with numeric values for repetition
    - — *Ex mode* allow file manipulation; to enter this mode, type a colon (`:`) followed by the command desired
    - — *Insert mode* to edit text; enter this post by pressing **`i`**
    - — The **`ESC`** key always returns to command mode
  - » `vi` command examples:
    - — **`h`** – Move one character left
    - — **`j`** – Move down line
    - — **`k`** – Move up line
    - — **`l`** – Move one character right
    - — **`H`** – Move to top of screen
    - — **`L`** – Move to bottom of screen
    - — **`G`** – Move to end of file
    - — **`0`** – Move to beginning of line
    - — **`$`** – Move to end of line
    - — **`i`** – Insert at current position
    - — **`I`** – Insert at beginning of line
    - — **`x`** – Delete character
    - — **`dd`** – Delete line
    - — **`o`** – Create a blank line after current line

  - **O** – Create a blank line before current line
  - Additional commands can be found in the `vi/vim` man pages

- Text files can be created by various methods outside of `vi`, `vim` or other editors:
  - `touch testfile.txt`
    - Creates a blank file called `testfile.txt` owned by the current user in the current directory
  - `echo "Some value" > testfile.txt`
    - Creates a file called `testfile.txt`, owned by the current user in the current directory, containing the text following the echo command (in this instance, "Some value")
    - Redirection of almost any command will take the output and place it into a file

Search for Files

- `vi/vim`
  - Moves cursor to a certain location, based on a search
  - In command mode, press **f** and then a character; this moves to the next occurrence of value in the current line
  - To search for a value in the entire file, in command mode, use **/** and then type the value
  - Similar to sed, you can use `vi`/`vim` to replace values:
    - `:%s/oldvalue/newvalues/g`
- `find`
  - Commonly used to find files by name, by user and/or by type
  - `find /start/dir -name "value"`
    - Finds files in /start/dir with the name *value*
  - `-maxdepth`
    - The number of subdirectories to search
  - `-type`
    - `f` – Regular file
  - `-size`
    - `+#[MGT]` – number and type of storage value (mega, giga, terabytes)
  - `-perm`

    - Numeric designation of permissions to look for
  - `-exec`
    - `rm '{}' +`
      - Execute the command `rm` on the values/files within the results of the `find` command
- Many more values and examples in the `man` pages

Use Input/Output Redirection, Compare Text Files, Compare Binary Files

- Redirection is used to direct the output of a command or task into a file, or direct the contents of a file to a command
  - `>` – Redirect output to a file, overwriting the contents if any exist
  - `>>` – Redirect output to a file, appending to the contents if any exist
  - `|` – "Pipe" the output to another command and/or file

Archive, Compress, Unpack and Decompress Files

- `tar`
  - Can be used to archive, unarchive, compress and decompress files
  - Archives values in the `/etc` directory in a file called `etc.tar`:
    - `tar cvf etc.tar /etc`
  - Decompresses archive:
    - `tar xvf etc.tar`
  - Archives values in the `/etc` directory in a file called `etc.tar`, and also compress that file with `gzip`:
    - `tar cvzf etc.tar /etc`
  - List contents of archived file:
    - `tar tvf etc.tar`
  - `c` – Archive
  - `v` – Verbose
  - `f` – Write to a file
  - `x` – Extract
  - `z` – `gzip` compression

  - » `j` – `bzip` compression
  - » `A` – Append files to existing archive
  - » `t` – List contents

List, Set and Change Standard File Permissions

- `chmod`
  - » Used to change the read, write and execute privileges of a file or directory for the owner, the group(s) they belong to, and all others (often called *everyone*)
- `chmod 777 somefile.txt`
  - » This would change the file `somefile.txt` to read/write/execute for the owner, the group belonged to, and everyone else (universal permissions)
- Numeric and character values for the permissions are as follows:
  - » **r** – **4** – Read permissions
  - » **w** – **2** – Write permissions
  - » **x** – **1** – Execute permissions
    - — Adding these values together determines the privilege number
- Example: `chmod 755 somefile.txt`
  - » User: read (4) + write (2) + execute (1) = 7
  - » Group: read (4) + execute (1) = 5
  - » Everyone: read (4) + execute (1) = 5
- Permissions can also be changed by single category using character representation
- Example: `chmod g+rw somefile.txt`
  - » Adds group permissions of read and write to the file called `somefile.txt`
    - — Character values as follows:
      - **u** – User
      - **g** – Group
      - **a** – Everyone
      - **r** – Read
      - **w** – Write

    - **x** – Execute
    - **+** – Add the values indicated
    - **-** – Remove the values indicated
- `chown`
  - » Change ownership to a specific user and/or group
- `chgrp`
  - » Change group ownership to a specific group

Transfer Files Securely via SFTP/SCP

- `scp`
  - » Based on the SSH secure protocol for transfer of files
  - » Transfer files to/from a server:
    - — `scp user@RemoteIP:/remote/dir/myfile.txt /home/user`
  - » Copy the remote file called `myfile.txt` to the local `/home/user` directory:
    - — `scp myfile.txt user@RemoteIP:/remote/dir`
  - » Copies the local file called `myfile.txt` to the remote directory `/remote/dir`:
- `sftp`
  - » SSH-based file transfer program whose behavior is based on the less-secure program, FTP
  - » Same format as the scp in terms of individual files or directories
  - » Add batch processing capabilities:
    - — `sftp -b batch.file`
  - » Similar to `expect` scripts, using FTP commands to run through many transactions in one listing

Operation of Running Systems

Change the Priority of a Process/Identify Resource Utilization by Process

- `jobs`
  - » Displays minimal information about processes associated with the current session
- `ps`

  - » By default, `ps` only displays process that were run from its own terminal
  - » `-A\-e` – Displays all processes on a system
  - » `-u` – Displays processes given by a specified user
  - » `-H` – Groups processes and use indentation to show the hierarchy of relationships between processes
  - » `-w` – Tells `ps` not to truncate to system
- `uptime`
  - » Find uptime and display load average
- `bg`
  - » Restores a job to running status, but in the background
- `fg`
  - » Use **CTRL+Z** to pause a program and, then `fg` to send the program to foreground
- `kill`
  - » Can be used to stop executing processes, uses PID
- `nohup`
  - » Run a command immune to hangups, with output to console or non-tty
- `killall`
  - » Can be used to kill all processes of a certain name
- `free`
  - » Show free memory and swap
- Common kill signals:
  - » `SIGHUP 1 HANGUP`
  - » `SIGINT 2 INTERRUPT FROM KEYBOARD`
  - » `SIGKILL 9 KILL SIGNAL`
    - — This signal is not blockable and causes the program to terminate abruptly; only use if you can't terminate with 15
  - » `SIGTERM 15 TERMINATION SIGNAL`
    - — Asks the program to finish what it is doing, then exits; clean exist; the preferred way of killing processes
  - » `SIGSTOP 17,19,23 STOP THE PROCESS`
    - — When a child process exits from a parent process it sends signal 1

    - Signals in the man page `man -k signal`

Manage the Startup Processes and Services

- Reboot the system:
  - `reboot`
  - `systemctl reboot`
  - `shutdown -r now`
    - `-r` – reboot
    - `now` – Reboot immediately
    - `+5` – Wait 5 minutes and then reboot
    - `+0` – Same as now
    - `01:01` – 1:01 AM shutdown
    - `-c` – Cancel a scheduled shutdown
  - `init 6`
    - The *init* system in Red Hat 7 is depreciated. However, runlevels are still compatible for this current version for backwards compatibility
- Shutdown the system (no reboot/power off):
  - `systemctl halt`
  - `halt`
  - `shutdown -h now` (`-h` means halt)
  - `init 0`
- Physically power off the system:
  - `systemctl poweroff`
  - `poweroff`
  - `shutdown -P`

Managing User Processes

- `nice`
  - Run a program with modified scheduling priority
- `renice`
  - Alter priority of running processes

- `top`
  - » Display Linux processes
  - » While running:
    - — **`K`** – Kills processes
    - — **`Q`** – Quits processes
    - — **`r`** – Change process priority
    - — **`s`** – Change update rate
    - — **`P`** – Sort by CPU usage
    - — **`m`** – Sort by memory usage; can also show uptime, memory info, and load average
  - » From the command line:
    - — `-d` – Specifies delay between updates
    - — `-p` – Lists up to 20 specific PIDs
    - — `-n` – Display certain number of updates then quit
    - — `-b` – Batch mode

Set File Permissions and Ownership

- `chmod`
  - » Used to change the read, write and execute privileges of a file or directory for the owner, the group(s) they belong to, and then all others (often called *everyone*)
- `chmod 777 somefile.txt`
  - » This would change the file *somefile.txt* to read/write/execute for the owner, the group belonged to, and everyone else (universal permissions)
- Numeric and character values for the permissions are as follows:
  - » r = 4 – read permissions
  - » w = 2 – write permissions
  - » x = 1 – execute permissions
    - — Adding these values together determines the privilege number
- Example: `chmod 755 somefile.txt`
  - » User: read (4) + write (2) + execute (1) = 7

  - » Group: read (4) + execute (1) = 5
  - » Everyone: read (4) + execute (1) = 5
- Permissions can also be changed by single category using character representation
- Example: `chmod g+rw somefile.txt`
  - » Adds group permissions of read and write to the file called *somefile.txt*
- Character values as follows:
  - » u – user
  - » g – group
  - » a – everyone
  - » r – read
  - » w – write
  - » x – execute
  - » + – add the values indicated
  - » - – remove the values indicated
- `chown`
  - » Changes ownership to a specific user and/or group
- `chgrp`
  - » Changes group ownership to a specific group

Installing Software Packages (Debian/Ubuntu)

- `/etc/apt/sources.list`
  - » Stores the repository locations that apt uses to search for packages specific to your system
- `apt-get`
  - » Package handling and installation utility for Debian-based distributions
  - » Installs packages by name and includes dependent packages during install
- `dpkg`
  - » Installs .deb package files on Debian-based systems
  - » `-i` – Install as well as configure package
  - » `-r` – Remove package

  - » `-configure` – Configures a package
  - » `-c` – List contents of a package
  - » `-s` – List status of package (installed or not)
- `apt-cache`
  - » Allows searching of named package or shows installed packages
    - — `apt-cache pkgnames` – Show installed
    - — `apt-cache search` – Search for named package
- `dpkg-reconfigure`
  - » Reconfigure an already-installed package
- `aptitude`
  - » High level package management interface for Debian-based distributions

Installing Software Packages (Red Hat/RPM)

- `rpm`
  - » Package installation utility for Red Hat-based distributions
  - » `-nodeps` – Install the package without worrying about installed dependencies
  - » `-i` – Install
  - » `-K` – Check package signature
  - » `-V` – Verify
  - » `-a` – All packages
    - — `-Va` – Verify all packages
  - » `-U` – Upgrades or installs a new package
  - » `-F` – Upgrade an already-installed package
  - » `-q` – Query a package to determine if already installed
  - » `-e` –Erase or uninstall
  - » `-f` – Query package-owning file
  - » `-p` – package
  - » `-l` – List files in a package
  - » `-rebuild` – Rebuilds a source package

  - » `-rebuilddb` – Rebuilds the rpm database
  - » `-qa` – Print all install packages
  - » `-ql` – List files in an installed package
  - » `-qf` – Determine which installed packaged a file belongs to
  - » `-qpl` – List all files in an RPM package
  - » `-checksig` – Same as -K
- `/etc/yum.repos.d`
  - » Directory containing yum source repository files
- `cpio` – Create cpio archive
  - » `-d` – Create leading directories where needed
  - » `-i` – Extract
  - » `-u` – Replace all files without asking
  - » `-m` – Retain previous modification times when creating files
- `rpm2cpio`
  - » Converts RPM packages to CPIO compressed files
  - » Used primarily to extract files from a RPM package without installing the RPM package
    - — `rpm2cpio file.rpm | cpio -dium`

Shell

- `echo - $$` – Displays current shell process
- `$?` – Prints the exit value of the command to the screen
- `Exit code 0` – Means command completed successfully
- `$!` – PID of last job run in background
- `$*` – Expands all parameters passed
- `$$` – Contains current process ID
- `$@` – Each passed parameter expands to a different word
- `$0` – Show the name of the shell or script
- `$_` – Set at shell startup and contains the absolute file name of the shell/script being execute; expands last argument to the previous command

User and Group Management

Create, Delete and Modify Local User Accounts

- `/etc/passwd` – This file contains the list of users on the system.
- Format:
  - `username:password:UID:PrimaryGUID:comment:homedir:defaultshell`
  - Example: `linuxacademy:x:539:100:linux academy act:/home/linuxacademy:/bin/bash`
- User IDs under 100 are reserved for system users
- Normal user accounts have IDs between 500-1000
- Can directly edit the `passwd` file to add, remove or modify users
- Changing the shell section of the format to `/bin/false` will prevent the user shell login access to the system
- `/etc/passwd` permissions must be readable by all, but `/etc/shadow` only should be readable by superusers
- `pwck` – Verifies the integrity of the users and authentication information. Checks entries for `/etc/passwd` and `/etc/shadow` for proper format
- `/etc/skel` – Skeleton file used when creating new users; allows you to set what files and settings need to be configured as default for every user added
- `useradd/adduser` – Creates a user on the system. Some distributions such as Slackware use `adduser` instead of `useradd`. Note: Both username and password are case-sensitive.
  - `-c` – Comment; can be used as a comment, but is currently used for user's full name
  - `-d` – Sets the user home directory; by default it is /home/user, but can be set to anything
  - `-e` – Expire-date; sets the user's expiration date; on this date the account password will "expire" and the user will no longer have access; format: YYYY-MM-DD
  - `-p` – Sets an encrypted password; the pre-encrypted password is added as-is to your /etc/passwd and /etc/shadow files; this is not the advised method of setting the password
  - `-M` – Does not create the home directory, even if the /etc/login.defs has the default set to yes
  - `-m` – Creates a home directory at /home/username if it does not exist; files contained in the /etc/skel directory will be copied into the new user's home directory; useradd creates a user's home directory by default
  - `-G` – Defines all the other groups that the member belongs to; separate each group by a comma

  - » `-g` – Sets the default group for the user; this is the user's group when the user first logs in
  - » `-f` – Defines the number of days after a password expires before an account is permanently disabled; a value of 0 immediately disables the account after password expiration, whereas -1 disables the entire feature
  - » `-k` – Defines which directory skeleton to use when creating a user; allows you to have different default settings for different users; if the option is not set, it uses the /etc/skel format by default

- `/etc/default/useradd` – Location of default settings for the useradd command
- `/etc/shadow` – Contains the encrypted passwords for the user accounts on the system; this and the /etc/passwd file can be directly modified; the useradd and usermod commands are an interface to automatically modify these files.
- Format:
  - » `username:password:days_until_change_allowed:days_before_change_required:days_of_warning_before_expiration:days_between_expiration_activation:expiration_date:special_flag`
  - » Flag names are self-explanatory; however, a value of -1 or 99999 will indicate that the feature is disabled for that user
- `chage` – Changes and manages user expiry information; changes the number of days between required password change, and forces password changes for users after x number days
  - » `-E` – Sets the date that the user's password will expire
  - » `-I` – Sets the number of days of inactivity after a password has expired before locking account
  - » `-m` – Sets minimum number of days between password changes
  - » `-M` – Sets maximum number of days which a password is valid
- `userdel` – Deletes a user account and associated files.
  - » `-f` – Forces the removal of the user account even if the user is still logged in; also deletes the user's home directory and mail; not typically recommended — you can use `kill` to boot a user from your system and then remove the user account
  - » `-r` – Removes the user's home directory, files located in the user's home directory and the user's mail; does not remove files owned by the user outside of their home directory — use the `find` command to find files based off owner.
- `usermod` – Modifies a user account
  - » `-d` – Sets the user's home directory to a new directory
  - » `-e` – Sets date for when the user account will expire; use YYYY-MM-DD
  - » `-f` – Number of days after a password expires until account is permanently disabled

  - `-g` – Group ID/name of the user's new default login group
  - `-G` – List of extra groups the user is a member of
  - `-l` – Changes the login name of the user
  - `-L` – Locks the user's account

Create, Delete and Modify Local Groups

- `groupdel` – Deleted group; if any user has this group as their primary group, then the group cannot be removed until it is removed as the primary group
- `groupmod` – Modify group name or group ID
  - `-g` – Specify a new group ID; returns error if group already exists
  - `-o` – When used with `-g`, allows two groups to share the same group ID
  - `-n` – Specifies a new group name
- `/etc/group` – This file contains a list of groups and all the members associated with the groups
- Example of `/etc/group`:
  - `groupName:Password:GUID:userlist`
- groupadd – Adds a group to the system
  - `-g` – Specifies a group idea; if not specified it will auto-select one for you
  - `-r` – Instructs groupadd to pick a group ID; less than 500 used for system groups
  - `-f` – Forces group creation even if another group already exists

Networking

Configure Network Services to Start on Boot – systemd

- Sysvinit (older service management, CentOS/RHEL 6.x, Ubuntu/Debian prior to 10.04/8)
  - Install an example server service:
    - `yum install openssh-server`
  - Enable the service to start on reboot:
    - `chkconfig openssh-server`
  - Start the service in current session:
    - `service openssh-server start`

- Systemd (modern service management, all recent distributions)
  - » Install an example server service:
    - — `yum install openssh-server`
  - » Enable the service to start on reboot:
    - — `systemctl enable openssh-server`
  - » Start the service in current session:
    - — `systemctl start openssh-server`
  - » Query the status of a service (running or otherwise):
    - — `systemctl status openssh-server`

Implement Packet Filtering

- iptables firewall
- List the existing rules:
  - » `sudo iptables -L`
- Policies generally available:
  - » ACCEPT – Lets the packet through
  - » DROP – Drops the packet quietly
  - » REJECT – Rejects the packet and returns a message to the requestor
- General format for iptables rules:
  - » `iptables -A name_of_chain criteria_to_meet -j target_of_rule`
    - — `-A` – Append the rule to the end of the chain
    - — `name_of_chain` – One of INPUT, OUTPUT or FORWARD
    - — `criteria_to_meet` – The conditions against which all packets are inspected against to determine whether the rule applies
    - — `target_of_rule` – The action or policy to apply (ACCEPT, REJECT, or DROP)
- Example: Drop ICMP ping requests between servers:
  - » `iptables -A INPUT -protocol icmp -in-interface enp0s3 -j DROP`
    - — This causes ping commands to drop with no returned response
- Example: Reject ICMP ping requests between servers:
  - » `iptables -A INPUT -protocol icmp -in-interface enp0s3 -j REJECT`

    - This returns a message to the requestor: “Destination Port Unreachable”
    - This can inadvertently expose that your IP address is valid; this setting should be used only internally when you want the client systems to know the port is filtered

Monitor Network Performance

- Socket connections
  - Use the utility `ss` (replacement for `netstat`)
    - “socket statistics”
  - Show all TCP ports open on a server:
    - `ss -t -a`
      - `-t` – All TCP ports
      - `-a` – All connections
  - Show established connections with their timers:
    - `ss -t -o`
      - `-t` – All TCP ports
      - `-o` – Time established
  - Filter by socket:
    - `ss -tn sport = :22`
      - `sport = :22` – Source port of the established connection
- Identify open ports and active hosts
  - Use the `nmap` utility (defensive scanning of your own network)
  - Scan ports on the system or remote host
    - `nmap -A -sS [IP/Hostname]`
      - `-A` – Deep scan for all discoverable ports and services
      - `-sS` – Use TCP SYN (prevents leaving a logged footprint on the remove system)
- Monitor all IP Traffic
  - `iptraf`
    - Ncurses-based, shows all packets across all interfaces (local, physical, virtual)
  - `dstat`

  - Shows second-by-second monitoring statistics on your system by process, including read/write, PIDs and system usage

Configure the Firewall

- Firewalld is the default firewall daemon
  - » Manage at the command line with firewall-cmd
- Allows working with services, zones and rules, as well as import custom rules from XML files
- Service configurations are stored in XML files located at `/usr/lib/firewalld/services` and `/etc/firewalld/services` (depending on user-defined (in `/usr/lib`) or system (`/etc/firewalld`) level definitions)
- Restart the firewall:
  - » `systemctl restart firewalld`
- Reload the persistent firewall rules:
  - » `firewall-cmd --reload`
- Zone management parameters
  - » `--get-default-zone`
  - » `--set default-zone`
  - » `--get-active-zones`
  - » `--get-zones`
  - » `--list-all`
  - » `--list-all-zones`
  - » `--new-zone`
  - » `--delete-zone`
  - » `--permanent`
  - » `--zone`
- Current vs permanent firewall rule changes
  - » Commands making rule changes of any kind are NOT persistent (in other words, will not persist across reboots or service restarts) UNLESS the `--permanent` option is specified
  - » Commands making rule changes take immediate effect UNLESS specifying the `--permanent` parameter, rules specifying permanent must be applied with a subsequent `--reload`
- Service management options
  - » `--get-services`

  - » `--list-services`
  - » `--query-service`
  - » `--add-service`
  - » `--remove-service`
  - » `--new-service`
  - » `--delete-service`
- Port management options
  - » `--list-ports`
  - » `--add-port`
  - » `--remove-port`
  - » `--query-port`
- When adding ports, the port number must specify TCP/UDP for application ranges can be added with dashes
  - » `--add-port 500-599/TCP for example`
- Rich rule management
  - » `--list-rich-rules`
  - » `--add-rich-rule`
  - » `--remove-rich-rule`
  - » `--query-rich-rule`
- Example: Rich rules to allow inbound HTTP access from a network IP range 10.0.1.0/24, logging each one at the info log level and making the change permanent:
  - » `firewall-cmd -add-rich-rule rule-family="ipv4" source address="10.0.1.0/24" service name="http" log prefix="HTTP Allow Rule" level="info" accept --permanent`
  - » Updates the default zone (public by default) XML rule file in `/etc/firewalld/zones/public.xml`
- Port forwarding
  - » `--list-forward-ports`
  - » `--add-forward-port`
  - » `--remove-forward-port`
  - » `--query-forward-port`
- Example: Forwarding by redirecting SSH from port 22 to port 2222 in the DMZ zone:

  - » `firewall-cmd -zone DMZ -permanent -add-forward-port port=22:proto=tcp:toport=2222`

Service Configuration

Provide/Configure Network Shares via NFS/CIFS

- Installing the server packages (server setup):
  - » `yum install nfs-utils`
- Create a directory for the shared or exported filesystem:
  - » `mkdir /var/myshare`
- Change the permissions to be sure everyone has access:
  - » `chmod -R 777 /var/myshare`
- Enable and start the appropriate services:
  - » `rpcbind`
  - » `nfs-server`
  - » `nfs-lock`
  - » `nfs-idmap`
- Create/edit the `/etc/exports` file, add the filesystem with appropriate options:
  - » `/var/myshare 192.168.1.0/24(rw,sync,no_root_squash,no_all_squash)`
    - — `rw` – Read/write filesystem
    - — `sync` – Keep local and remote filesystem caches in sync
    - — `no_root_squash,no_all_squash` – Do not attempt to do root or user level remapping based on UIDs for security
- Start the NFS Server service:
  - » `systemctl restart nfs-server`
- Install the packages (client setup):
  - » `yum install nfs-utils`
- Create a directory to mount the share:
  - » `mkdir /mnt/shareddrive`
- Enable and start the appropriate services:
  - » `rpcbind`

  - » `nfs-server`
  - » `nfs-lock`
  - » `nfs-idmap`
- Example: Command line mount
  - » `mount -t nfs SERVERIP:/var/myshare /mnt/shareddrive`
- Example: Persistent mounting in `/etc/fstab`
  - » `SERVERIP:/var/myshare /mnt/shareddrive nfs default 0 0`
- Mount automatically on boot or on demand:
  - » `mount -a`

Configure an SMTP Service

- Packages to install:
  - » `dovecot`
  - » `postfix`
    - — CentOS/RHEL – yum install dovecot postfix
    - — Debian/Ubuntu – apt-get install dovecot postfix
- Add email aliases:
  - » `/etc/postfix/aliases`
    - — `email_account: alias_name1, alias_name2`
    - — More than one alias can be added for each account to alias
    - — Refresh the alias table:
      - ▪ `postalias /etc/postfix/aliases`
- Configure postfix:
  - » `/etc/postfix/main.cf`
    - — `myorigin = /etc/mailname` (or domain itself)
      - ▪ The file that contains the domain name of the server
      - ▪ Can contain the domain name directly, filename is older standard still in use
    - — `mydestination =` List of domains the mail server will deliver messages to locally instead of forwarding to another system/mail server
    - — `mydestination = myserver.domain.com, localhost.domain.com, localhost`

(examples)

- `mynetworks = subnet`
  - Indicates that we are serving IPs in the local subnet the server exists on
- `inet_interfaces = all`
  - Accepts connections and messages to/from all defined network interfaces (localhost, physical, virtual, other)
- `mailbox_size_limit = #####`
- `message_size_limit = #####`
  - Self-explanatory, can be set to whatever requirements needed, in bytes

» `/etc/postfix/transport`

- Contains definitions of relationships between domains and the next server that mail messages need to be forwarded to. For example:
  - `example.domain.com local:`
  - `.example.comain.com local:`

» Process the transport file and create/update the values to the mail DB format:

- `Postmap /etc/postfix/transport`

- Restrict access to SMTP

» `/etc/postfix/main.cf`

- `smtpd_helo_required = yes`
  - Require mail client to introduce the mail transaction with standard HELO identification
- `smtpd_helo_restsrictions = permit_mynetworks, reject_invalid_helo_hostname`
  - Permits only defined networks (see above) and hosts that identify with HELO appropriately
- `smtpd_sender_restrictions = permit_mynetworks, reject_unknown_sender_domain`
  - Only accepts email to be sent from defined networks and rejects all others from unknown domains
- `smtpd_recipient_restrictions = permit_mynetworks, reject_unauth_destination`
  - Accept only locally-defined user destinations (see above configuration) and reject any unauthorized

- Enable and restart the postfix service:

  - » `systemctl enable postfix`
  - » `systemctl restart postfix`

- Configure Dovecot:
  - » Initial installation supports IMAP and POP3, need to add IMAPS
    - — `/etc/dovecot/dovecot.conf`
      - `Protocols = pop3 imap imap3`

- Test listening on available ports:
  - » `netstat -npltu | grep dovecot`
    - — Should see entries for ports 110 (pop3), 143 (imap), 993 (imaps) and 995 (pop3 secure)

- Use local `mail` subsystem to test:
  - » `mail email_account`
    - — This is our alias account from earlier in the configuration, should go to two users, as defined in the `/etc/postfix/aliases` file
  - » Enter subject, message; exit to send
  - » Log in as either of the aliased users, email should be available for them to read using local `mail` command

Configure SSH-Based Remote Access Using Public/Private Key Pairs

- Generate a public/private key pair for SSH key exchange utilization:
  - » `ssh-keygen`
  - » Passphrase prompt is optional, designed to add an additional security layer (i.e. both the key AND passphrase would then be required when connecting to a server in this manner)

- Copy the public key from a user to a remote host (note that the referenced account on the remote host must already exist)
  - » `ssh-copy-id user@[servername/serverip]`
    - — Prompts for remote user password the first time
    - — Does not connect to the session on the key copy, copies the key only

- Testing:
  - » `ssh user@[servername/serverip]`
    - — If done correctly, prompts for passphrase (if one was entered) or simply connects you to the remote host as the indicated user if no passphrase was entered during key creation

Configure an HTTP Server

- httpd
- Apache service to install:
- `systemctl enable httpd`
- `systemctl start httpd`
  - » By default, listening on port 80
- Key configuration files and directories:
  - » `/etc/httpd`
    - — Primary http configuration directory, all httpd configuration files and directories root here
  - » `/etc/httpd/conf`
    - — magic – mime configuration file for filetype definitions
    - — httpd.conf – Primary overall Apache configuration, default location for vhosts
  - » `/etc/httpd/conf.d`
    - — User and index configuration file (user directory, welcome message and indexes)
  - » `/etc/httpd/conf.d.modules`
    - — Module configuration files (as referenced in httpd.conf)
  - » `/etc/httpd/logs`
    - — Link to /var/log/httpd by default
  - » `/etc/httpd/modules`
    - — Apache modules, link to /usr/lib64/httpd/modules by default
  - » `/etc/httpd/run`
    - — Temporary run files for httpd process, link to /run/httpd by default
  - » `/var/www/html`
    - — Default directory for website content

Configure HTTP Server Logs

- Log format contained, by default, in the httpd.conf file and looks like, for example:
  - » `LogFormat "%h %l %u %t \"%r\" %>s %b" common`
- Can always be overridden within a specific vhost, using the same directive and format
- Format options are:

  - `%a` – Remote client IP
  - `%A` – Local client IP
  - `%b` – Number of bytes transmitted
  - `%B` – Number of bytes transmitted
  - `%{var}e` – Value of environment variable indicated
  - `%f` – Requested file
  - `%h` – Remote host name (or IP if reverse lookup fails)
  - `%H` – Requested protocol
  - `%{var}i` – Contents of the header line name; for example: (user-agent)i
  - `%l` – Remote log name
  - `%m` – Request method
  - `%{var}n` – Contents of the note named
  - `%{var}o` – Contents of the header named
  - `%p` – Canonical port that serviced the request
  - `%P` – Process ID that serviced the request
  - `%q` – Query string or search argument used (prefaced with "?")
  - `%r` – First line of the request
  - `%s` – Server response status (i.e. HTTP 200 OK)
  - `%t` – Time in common log formatting
  - `%T` – Time (seconds) taken to serve the content
  - `%u` – Named of authenticated user (if any)
  - `%U` – Requested URL path
  - `%v` – Server name servicing the request
  - `%V` – Server name according to UseCanonicalName setting

- Reference names are given custom logging formats (like *common* in the example above)
- `ErrorLog` and `AccessLog` directives can then refer to the reference names (when configured in `httpd.conf`) rather than overriding the LogFormat directive within a vhost
  - Example:

    - `ErrorLog logs/mysite-error.log reference_name`
  - Would format the site error log file, using the `reference_name` format as defined in `httpd.conf`

Configure SSL with Apache Server

- Install the openssl utility
  - `yum install openssl`
  - `apt-get install openssl`
- Generate a certificate key file (standard example):
  - `openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /path/to/cert/dir/site.key`
    - Generates a key for a certificate authority to generate a valid certificate expiring one year from the date of creation with industry standard NoDES, x509 RSA 2048bit encryption
- Generate a certificate key file and use it to generate a self-signed certificate for local use:
  - `openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /path/to/cert/dir/site.key -out /path/to/cert/dir/site.crt`
    - Takes the key created and immediately signs it locally with the information provided during certificate creation
- Install mod_ssl:
  - `yum install mod_ssl`
- Activate mod_ssl:
  - CentOS/RHEL – Restart Apache after mod_ssl installation:
    - `systemctl restart httpd`
  - Debian/Ubuntu – Enable the module and restart Apache:
    - `a2enmod mod_ssl`
    - `systemctl restart apache2`
- vhost directives needed
  - `<VirtualHost *:443>`
    - Enables port 443 for current vhost
  - `SSLEngine on`
    - Enables SSL for the current vhost
  - `SSLCertificateKeyFile /path/to/cert/dir/site.key`

    - Whatever path that the created key file exists in
  - `SSLCertificateFil /path/to/cert/dir/site.crt`
    - Whatever path that the install certificate file exists in
- Restart Apache once vhosts are created to activate them on the server

Set Up Name-Based Virtual Web Hosts

- Create virtual host directory:
  - `mkdir /etc/httpd/vhost.d`
- Add the directory to /etc/httpd/conf/httpd.conf
  - `include vhost.d/*.conf`
- Externalizes vhost configuration; by default, vhost configurations can be added directly to `httpd.conf`
- Sample configuration for domain `myhost.sample.com`:
  - file name: `/etc/httpd/vhost.d/myhost.sample.com_http.conf`
    - `<VirtualHost *:80>`
    - `ServerAdmin admin@myhost.sample.com`
    - `DocumentRoot /var/www/html/myhost.sample.com`
    - `ServerName myhost.sample.com`
    - `ServerAlias myhost`
    - `ErrorLog logs/myhost.sample.com-error_log`
    - `CustomLog logs/myhost.sample.com-access_log common`
    - `</VirtualHost>`
- Test the configuration:
  - `apachectl configtest`
- Apache must be restarted or reloaded gracefully to read the new virtual host:
  - `systemctl restart httpd`
  - `systemctl reload httpd`
- Show a dump of the virtual host configuration for Apache in general:
  - `httpd -D DUMP_VHOSTS`

Set Up Name Based Virtual Web Hosts with SSL

- vhost directives needed:

» `<VirtualHost *:443>`

— Enables port 443 for current vhost

» `SSLEngine on`

— Enables SSL for the current vhost

» `SSLCertificateKeyFile /path/to/cert/dir/site.key`

— Whatever path that the created key file exists in

» `SSLCertificateFil /path/to/cert/dir/site.crt`

— Whatever path that the install certificate file exists in

Storage Management

- Standard File Systems

» Partitioning

— `fdisk/gdisk /dev/drive`

▪ Allows us to create a partition from the available indicated disk

▪ Can set type of partition (GPT or MBT depending on use), list the partition types in the menu

» Format the filesystem partition created:

— `mkfs -t ext4 /dev/disk1`

▪ Formats the disk partition `disk1` as an ext4 partition

» Mount the formatted partition:

— `mkdir /mnt/mount`

— `mount -t ext4 /dev/disk1 /mnt/mount`

▪ Mounts the formatted drive, as an ext4 mount on the created directory

» Persistently mounting the disk above

— Obtain the UUID of the device:

▪ `ls -al /dev/disk/by-uuid`

— Add entry to /etc/fstab:

▪ `UUID=UUDI_OBTAINED /mnt/mount ext4 defaults 0 0`

— Mount the defined entry automatically, if not mounted in current session:

▪ `mount -a`

        - This will scan all defined mount points, and if not mounted, mount them using the definitions in `/etc/fstab`

- Encrypted File Systems
    - » System support encrypted file system query:
        - — `grep -I config_dm_crypt /boot/config-$(uname -r)`
    - » Determine if module is loaded:
        - — `lsmod | grep dm_crypt`
    - » Load module if needed:
        - — `modprobe dm_crypt`
    - » Partitioning
        - — Handled the same way as a typical disk and drive as defined above in the "Standard File Systems" section
    - » Install the cryptsetup utility:
        - — `yum install cryptsetup`
    - » Default encryption key setup
        - — Luks – Linux Unified Key Setup
    - » Set up the partitions with passphrase:
        - — `cryptsetup -y luksformat /dev/disk1`
            - Prompts for passphrase for unencrypting drive during mount/use
            - Large drives will take a long time to encrypt
    - » Open partition for use:
        - — `cryptsetup luksOpen /dev/disk1 reference_name`
    - » Format the filesystem partition created, using the mapper overlay created above
        - — Handled the same way as a typical disk and drive as defined above in the "Standard File Systems" section
            - Example: `mkfs -t ext4 /dev/mapper/reference_name`
    - » Mount the drive:
        - — `mount /dev/mapper/reference_name /mnt/mount`
    - » Close the partition once used and unmounted:
        - — `cryptsetup luksClose reference_name`

Virtualization

Configure a Hypervisor to Host Virtual Guests

- Install and configure the necessary packages:
  - `yum install -y qemu-kvm qemu-img`
- Install and configure the virtual manager GUI:
  - `yum install -y virt-manager`