Operating System

Operating system is act like a bridge between user and hardware, which provides a platform where user can run their software application.

Operating system divided between two parts

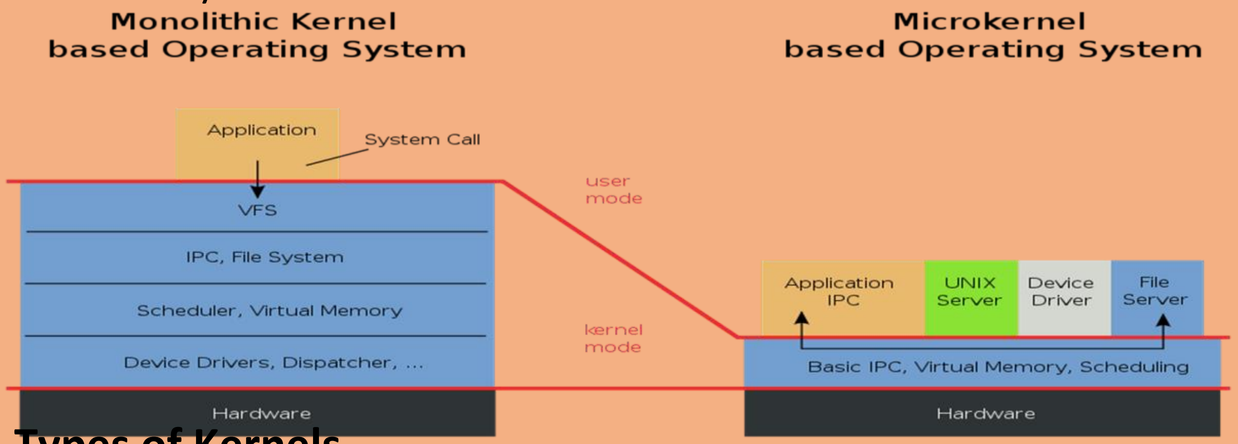
1. **Kernel**
2. **Shell**

Kernel

A kernel is a central component of an operating system. It acts as an interface between the user applications and the hardware. The sole aim of the kernel is to manage the communication between the software (user level applications) and the hardware (CPU, disk memory etc).

The main tasks of the kernel are:

- Process management
- Device management
- Memory management
- Interrupt handling
- I/O communication
- File system...etc...



Types of Kernels

Kernels may be classified mainly in two categories

1. Monolithic

Earlier in this type of kernel architecture, all the basic system services like process and memory management, interrupt handling etc were packaged into a single module in kernel space. This type of architecture led to some serious drawbacks like:

a) Size of kernel, which was huge.

b) Poor maintainability, which means bug fixing or addition of new features resulted in recompilation of the whole kernel which could consume hours

In a modern day approach to monolithic architecture, the kernel consists of different modules which can be dynamically loaded and un-loaded. This modular approach allows easy extension of OS's capabilities. With this approach, maintainability of kernel became very easy as only the concerned module needs to be loaded and unloaded every time there is a change or bug fix in a particular module.

2. Micro Kernel

This architecture majorly caters to the problem of ever growing size of kernel code which we could not control in the monolithic approach. This architecture allows some basic services like device driver management, protocol stack, file system etc to run in user space.

So, what the bare minimum that microkernel architecture recommends in kernel space?

- Managing memory protection
- Process scheduling
- Inter Process communication (IPC) Apart from the above, all other basic services can be made part of user space and can be run in the form of servers.

Shell

The shell is a piece of software that provides an interface for users. A Shell is a command interpreter and it can execute no of commands with in single command in logical order known as shell script. The shell acts as an interface between the user and the kernel. When a user logs in, the login program checks the username and password, and then starts another program called the shell.

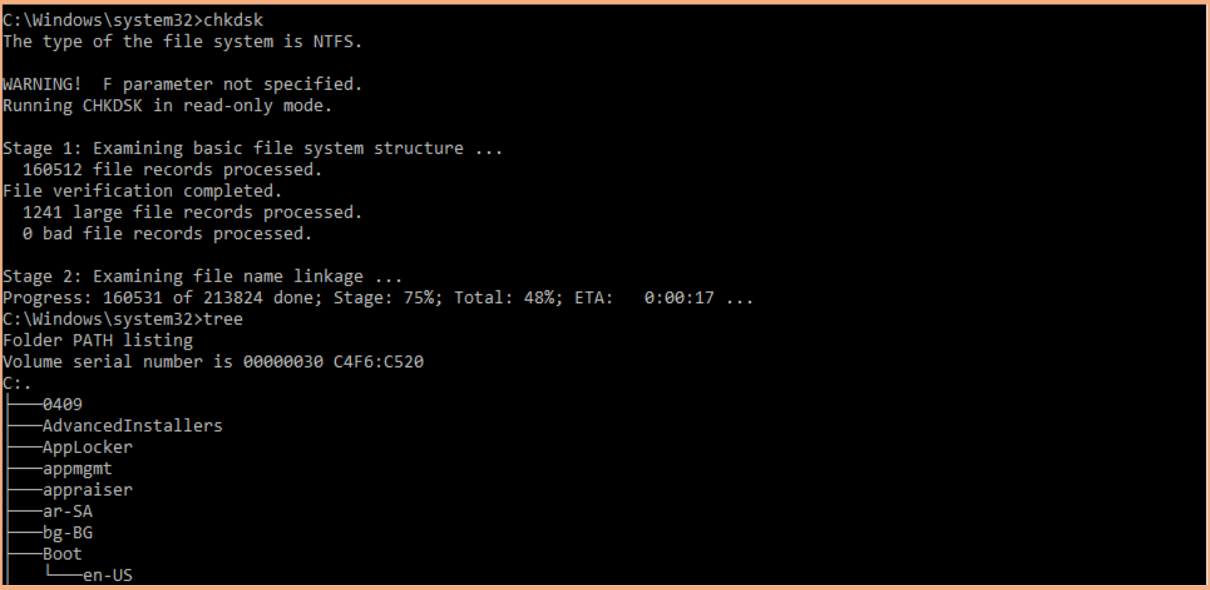
Type of Shell

1. CUI
   CUI stands for character user interface.
   In CUI user has to interact with the applications by making commands, in CUI only one task can run at a time.
   Everything is done by using commands.
   Examples (DOS, UNIX)

2. GUI
   GUI stands for graphical user interface.
   It is a user interface which user interact with applications by making use of graphics. In GUI more than one task can run simultaneously. The user interacts by pointing the applications using devices like mouse. It is a very user friendly interface
   Examples (Windows, Linux)

DOS (Disk Operating System)

Dos (Disk Operating System) is an operating system that runs from a hard disk drive. A disk operating system must provide a file system for organizing, reading, and writing files on the storage disk. It is CUI based Operating system. It's originally written by Tim Paterson and in August 1981 Microsoft introduce this OS and now DOS change to MS-DOS.



```
C:\Windows\system32>chkdsk
The type of the file system is NTFS.

WARNING!  F parameter not specified.
Running CHKDSK in read-only mode.

Stage 1: Examining basic file system structure ...
  160512 file records processed.
File verification completed.
  1241 large file records processed.
  0 bad file records processed.

Stage 2: Examining file name linkage ...
Progress: 160531 of 213824 done; Stage: 75%; Total: 48%; ETA:   0:00:17 ...
C:\Windows\system32>tree
Folder PATH listing
Volume serial number is 00000030 C4F6:C520
C:.
├───0409
├───AdvancedInstallers
├───AppLocker
├───appmgmt
├───appraiser
├───ar-SA
├───bg-BG
├───Boot
│   └───en-US
```

Some important information about popular Operating System

- In 80's, Microsoft's DOS was the dominated OS for PC
- Apple MAC was better, but expensive
- UNIX was much better, but much, much more expensive. Only for minicomputer for commercial applications
- People was looking for a UNIX based system, which is cheaper and can run on PC
- Both DOS, MAC and UNIX were proprietary, i.e., the source code of their kernel is protected
- No modification is possible without paying high license fees

History of UNIX & Linux

UNIX is one of the most popular operating systems worldwide because of its large support base and distribution. It was originally developed as a multitasking system for minicomputers and mainframes in the mid-1970's with its high performance and stability to support high cost computer with MIT and GE.

In **1965 Bell Labs** was adopting third generation computer equipment and decided to join forces with General Electric and MIT to create Multics (Multiplexed Information and Computing Service).
In **1969 AT&T** made a decision to withdraw Multics and go with GECOS (General Electric Comprehensive Operating Supervisor / System), with AT & T in Bells Lab when Multics was withdrawn some of the programmers named Ken Thompson and Dennis Ritchie decided to rewrite operating system in order to support low cost computer. Later on in **1973 UNIX** was rewritten in C programming language.

In **1984** Richard Stallman announces the GNU (Gnu is Not a Uniux) project to develop the GNU operating system a complete Unix like operating system which is a freeware for both Kernal and Program under FSF (Free Software Foundation) which licensed under GPL (General Public License).

In **1987**, under GPL license Andrew Tanenbaum developed inexpensive minimal Unix like operating system named MINIX which is limiting it to educational use. Hence in **1991** Linus Torvalds (16 yrs old) a Finnish student began to developed its own kernel as MINIX (GNU PROJECT) + NEW UNIX KERNAL(LINUS) = LINUX. Which is complete operating system. He launch this operating system on internet under GPL license for freeware distribution. So anyone can obtain this source codes and they can rewrite their own operating system.

First Article About Linux From:

torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds) Newsgroups: comp.os.minix Subject: What would you like to see most in minix? Summary: small poll for my new operating system Message-ID:<1991Aug25.205708.9541@klaava.Helsinki.FI>

Date: 25 Aug 91 20:57:08 GMT

Organization: University of Helsinki

Hello everybody out there using minix I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things). I've currently ported bash (1.08) and gcc(1.40),and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. A Any suggestions are welcome, but I won't promise I'll implement them :-) Linus (torvalds@kruuna.helsinki.fi) PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-(.

GNU & GPL

GNU Project: Focused on creating a UNIX like operating system that could be freely distributed. Established in 1984 by Richard Stallman, who believes that software should be free from restrictions against copying or modification in order to make better and efficient computer programs

GPL: Global Public license (Copyleft)

The GNU General Public Licence (GPL) allows anybody to:

- use the software at no charge, without any limitations,
- copy, and distribute or sell unmodified copies of the software in the source or binary form,
- use the software with propriatory (e.g., your own) modifications, free of charge, as long as you do not distribute or sell the modified version,
- modify, and distribute or sell a modified version of the software as long as the source code is included and licenced on the same terms as the original you received (the GPL),
- sell support for the software, without any limitations.

Linux Distros

A Linux distribution, often simply distribution or distro, is a member of the Linux family of Unix-like computer operating systems.

Distros mainly based on ‘Look and Feel’ and Applications

Major Linux Distributors

1. Red Hat Linux : One of the original Linux distribution. The commercial, non-free version is Red Hat Enterprise Linux, which is aimed at big companies using Linux servers and desktops in a big way. (NJIT)
   Free version: Fedora Project.
2. Debian GNU/Linux : A free software distribution. Popular for use on servers. However, Debian is not what many would consider a distribution for beginners, as it's not designed with ease of use in mind.
3. SuSE Linux : SuSE was recently purchased by Novell. This distribution is primarily available for pay because it contains many commercial programs, although there's a stripped-down free version that you can download.
4. Mandrake Linux : Mandrake is perhaps strongest on the desktop. Originally based off of Red Hat Linux.
5. Gentoo Linux : Gentoo is a specialty distribution meant for programmers.
6. Slackware Linux
7. Turbo Linux
8. Vector Linux& many more……..

The benefits of Linux

Linux can give you:

- A modern, very stable, multi-user, multitasking environment.
- Advanced graphical user interface. Linux uses a standard, network-transparent X-windowing system with a "window manager" (typically KDE or GNOME but several are available).
- The graphical desktop under Linux can be made to look like MS Windows (or probably ANY other graphical user interface of your choice).
- Freedom from viruses. Linux has no viruses because it is too secure an operating system for the viruses to spread with any degree of efficiency.

Red Hat

Red Hat Linux : One of the original Linux distribution.
The commercial, non-free version is Red Hat Enterprise Linux, which is aimed at big companies using Linux servers and desktops in a big way. (NJIT)
Red Hat Enterprise Linux The leading open source platform for modern data centers Red Hat® Enterprise Linux® delivers military-grade security, 99.999% uptime, support for business-critical workloads, and so much more. Red Hat® Enterprise Linux® Server fulfills core operating system functions and includes additional capabilities that provide an infrastructure.

Red Hat run some Global certification:-

Red Hat System Administrator – I (RH124)
Red Hat System Administrator – II (RH134)
Red Hat System Administrator – III (RH255)

Red Hat Enterprise Linux 7.0 Examination with Exam Code.

1. RHCSA (RedHat Certified **System Administrator**) – (EX200- 2 & half hours.)
2. RHCE (RedHat Certified **System Engineer**)-(EX300- 2hrs.)
3. RHCVA (Red Hat Certified **Virtualization Administrator**) – (EX318- 3hrs.)
4. RHSS (RedHat Certified **Security Specialist**) – (EX333- 6hrs. EX423- 4hrs. EX429-3 and half hours.)
5. RHCA (Red Hat Certified **Architect**) – (EX333- 6hrs. EX401- 4hrs. EX423 or EX318- 3hrs. EX436- 4hrs. And EX442- 4hrs.)
6. JBCAA (JBoss Certified **Application Administrator**)- (EX336- 4hrs.,)
7. RHCDS (Red Hat Certified **Data Center Specialist**) - (EX401- 4hrs. EX436- 4hrs. EX423 or EX318-3hrs.)

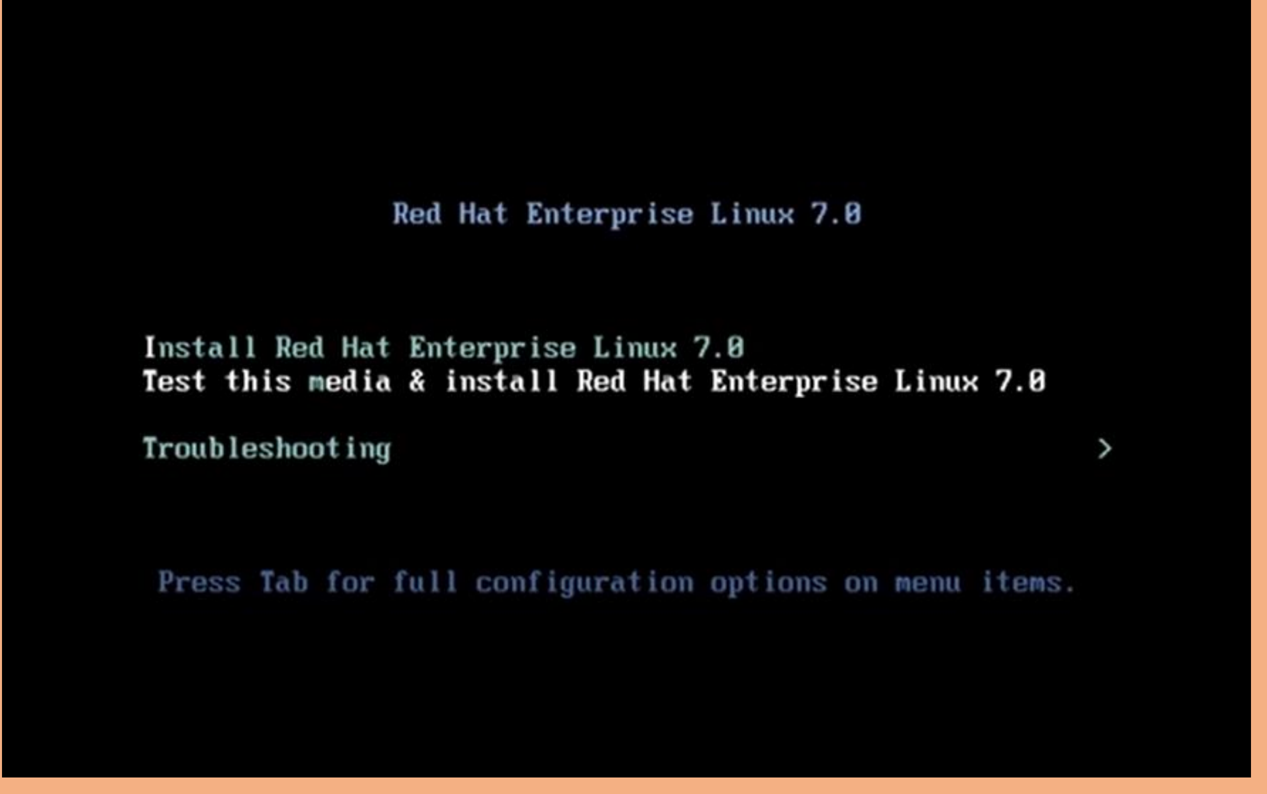
Training Path

1. Windows Admin = RHSCA = (RHCSA-I (RH124) + RHCSA-II (RH135))
↓
RHCE = (RHCSA-III with RHCSA and (RH255))
2. Linux/Unix Admin = RHCSA (Rapid Track Course with Exam RH200)
↓
RHCE = (RHCSA-III with RHCSA and (RH255))
3. Sr. Linux Admin = RHCE (Rapid Track Course with Exam (RH300))
4. Solaris Admin = Red Hat Enterprise Linux for Solaris Administrators (RH290)
↓
RHCE (Rapid Track Course with Exam (RH300))

System Administration 1 (RH-124) & System Administration 2 (RH-134)

Installation of Red Hat Enterprise Linux-7

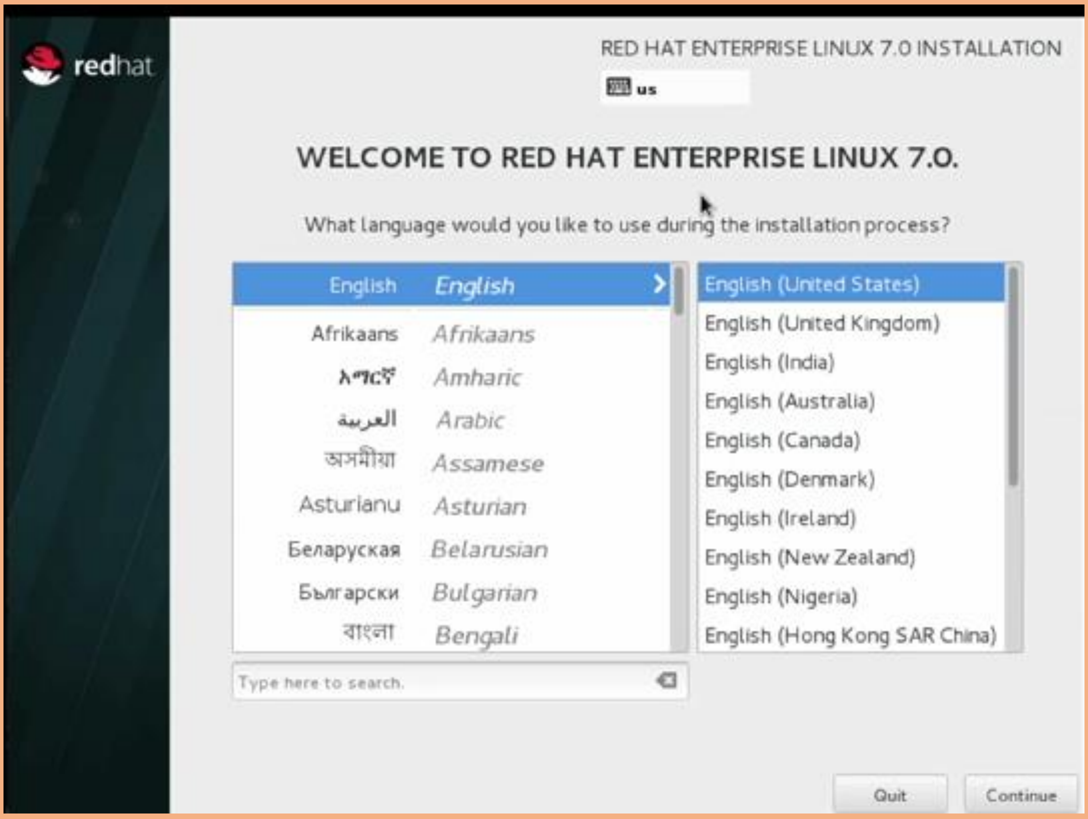
1. Insert Your media and restart your system.
2. Go to the Boot menu and select media type for installation.
3. Now Installation screen will display.



```
Red Hat Enterprise Linux 7.0

Install Red Hat Enterprise Linux 7.0
Test this media & install Red Hat Enterprise Linux 7.0

Troubleshooting                                        >

Press Tab for full configuration options on menu items.
```

4. Select “Install RedHat Enterprise Linux 7.0” by pressing up & down arrow and enter.

5. Select language for the installation process.



RED HAT ENTERPRISE LINUX 7.0 INSTALLATION

us

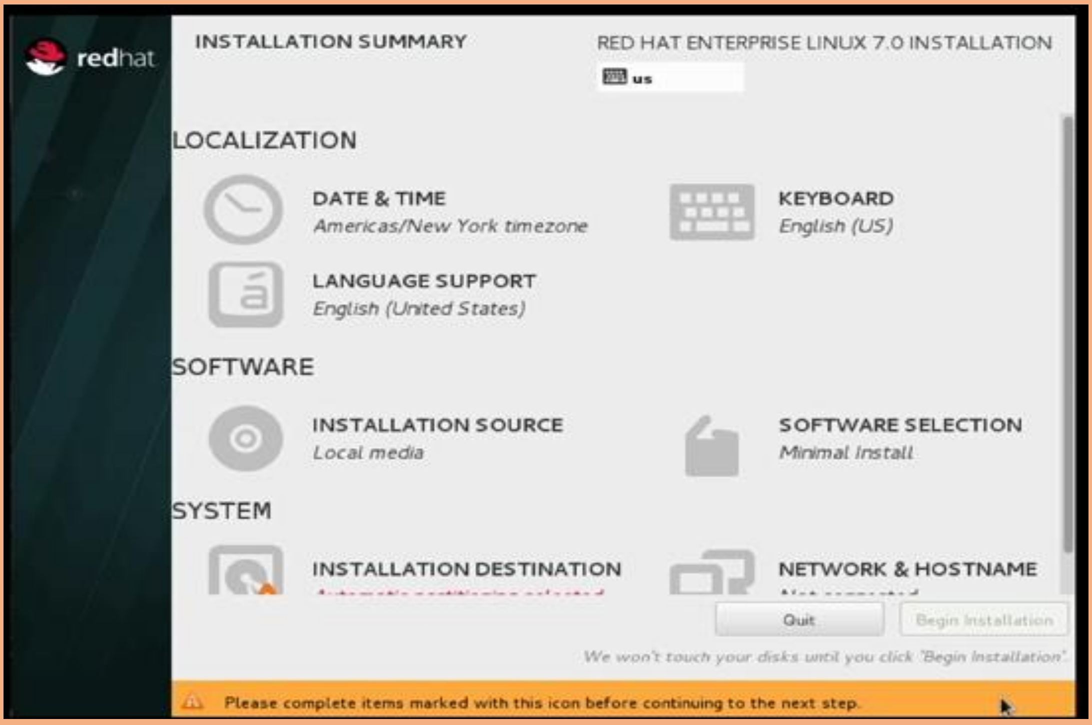
WELCOME TO RED HAT ENTERPRISE LINUX 7.0.

What language would you like to use during the installation process?

English	English
Afrikaans	Afrikaans
አማርኛ	Amharic
العربية	Arabic
অসমীয়া	Assamese
Asturianu	Asturian
Беларуская	Belarusian
Български	Bulgarian
বাংলা	Bengali

English (United States)
English (United Kingdom)
English (India)
English (Australia)
English (Canada)
English (Denmark)
English (Ireland)
English (New Zealand)
English (Nigeria)
English (Hong Kong SAR China)

6. Give Date & Time, Keyboard, Language Support setting.



INSTALLATION SUMMARY

RED HAT ENTERPRISE LINUX 7.0 INSTALLATION

us

LOCALIZATION

DATE & TIME
Americas/New York timezone

KEYBOARD
English (US)

LANGUAGE SUPPORT
English (United States)

SOFTWARE

INSTALLATION SOURCE
Local media

SOFTWARE SELECTION
Minimal Install

SYSTEM

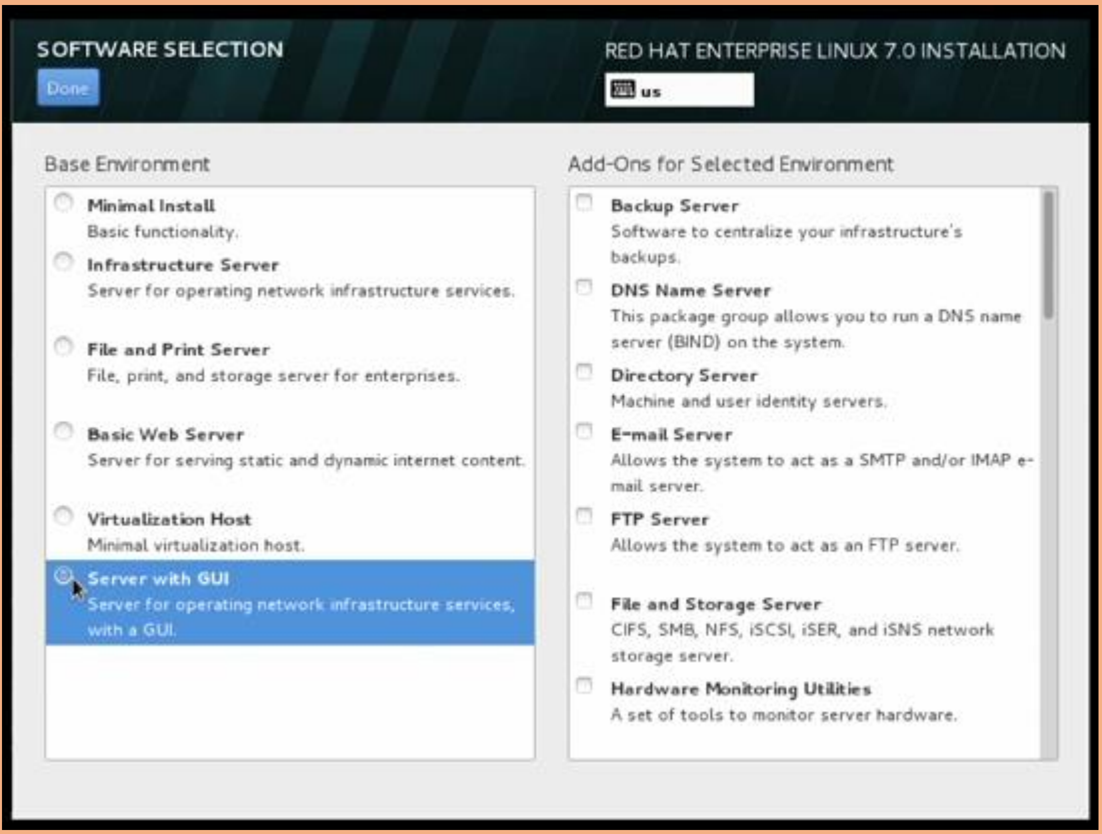
INSTALLATION DESTINATION

NETWORK & HOSTNAME

We won't touch your disks until you click 'Begin Installation'.

⚠ Please complete items marked with this icon before continuing to the next step.

7. Give Software selection and create partition



SOFTWARE SELECTION

RED HAT ENTERPRISE LINUX 7.0 INSTALLATION

Base Environment

○ **Minimal Install**
Basic functionality.

○ **Infrastructure Server**
Server for operating network infrastructure services.

○ **File and Print Server**
File, print, and storage server for enterprises.

○ **Basic Web Server**
Server for serving static and dynamic internet content.

○ **Virtualization Host**
Minimal virtualization host.

● **Server with GUI**
Server for operating network infrastructure services, with a GUI.

Add-Ons for Selected Environment

□ **Backup Server**
Software to centralize your infrastructure's backups.

□ **DNS Name Server**
This package group allows you to run a DNS name server (BIND) on the system.

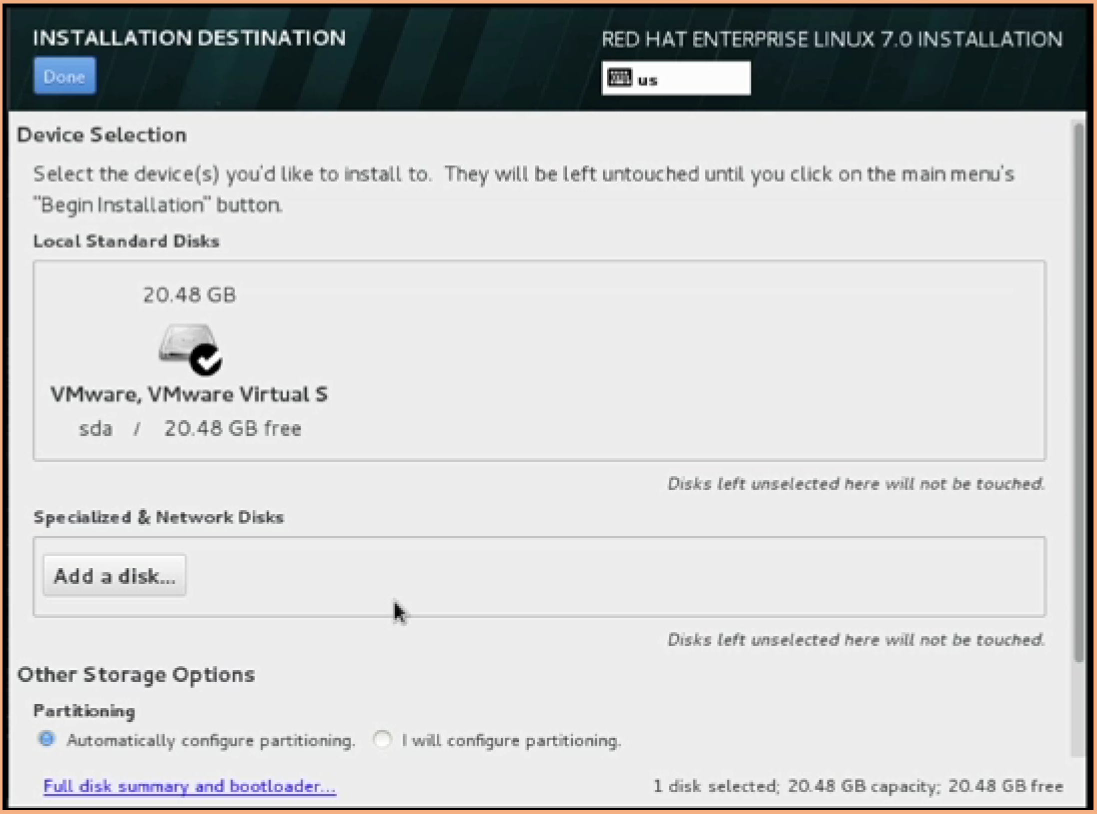
□ **Directory Server**
Machine and user identity servers.

□ **E-mail Server**
Allows the system to act as a SMTP and/or IMAP e-mail server.

□ **FTP Server**
Allows the system to act as an FTP server.

□ **File and Storage Server**
CIFS, SMB, NFS, iSCSI, iSER, and iSNS network storage server.

□ **Hardware Monitoring Utilities**
A set of tools to monitor server hardware.



INSTALLATION DESTINATION

RED HAT ENTERPRISE LINUX 7.0 INSTALLATION

Device Selection

Select the device(s) you'd like to install to. They will be left untouched until you click on the main menu's "Begin Installation" button.

Local Standard Disks

20.48 GB

VMware, VMware Virtual S
sda / 20.48 GB free

Disks left unselected here will not be touched.

Specialized & Network Disks

Disks left unselected here will not be touched.

Other Storage Options

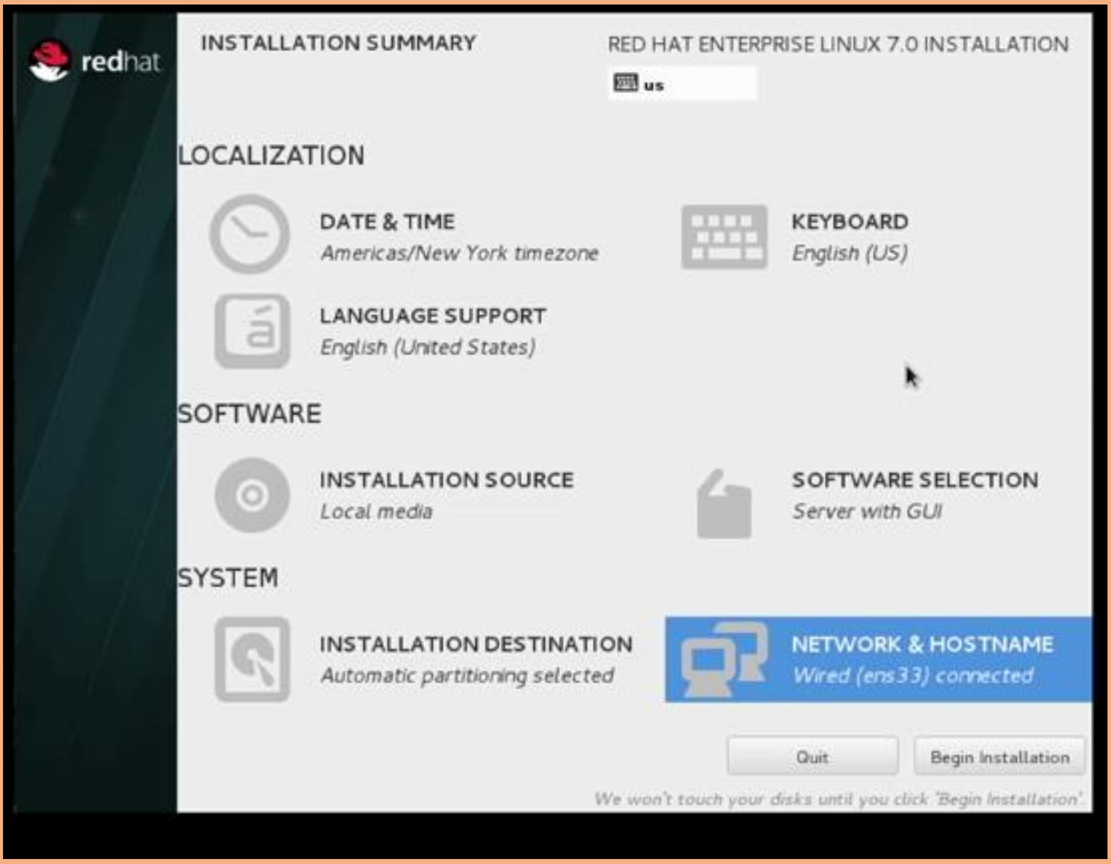
Partitioning

● Automatically configure partitioning. ○ I will configure partitioning.

Full disk summary and bootloader...

1 disk selected; 20.48 GB capacity; 20.48 GB free

8. Now click on “Begin installation” button for begin the installation.



INSTALLATION SUMMARY

RED HAT ENTERPRISE LINUX 7.0 INSTALLATION

LOCALIZATION

DATE & TIME
Americas/New York timezone

KEYBOARD
English (US)

LANGUAGE SUPPORT
English (United States)

SOFTWARE

INSTALLATION SOURCE
Local media

SOFTWARE SELECTION
Server with GUI

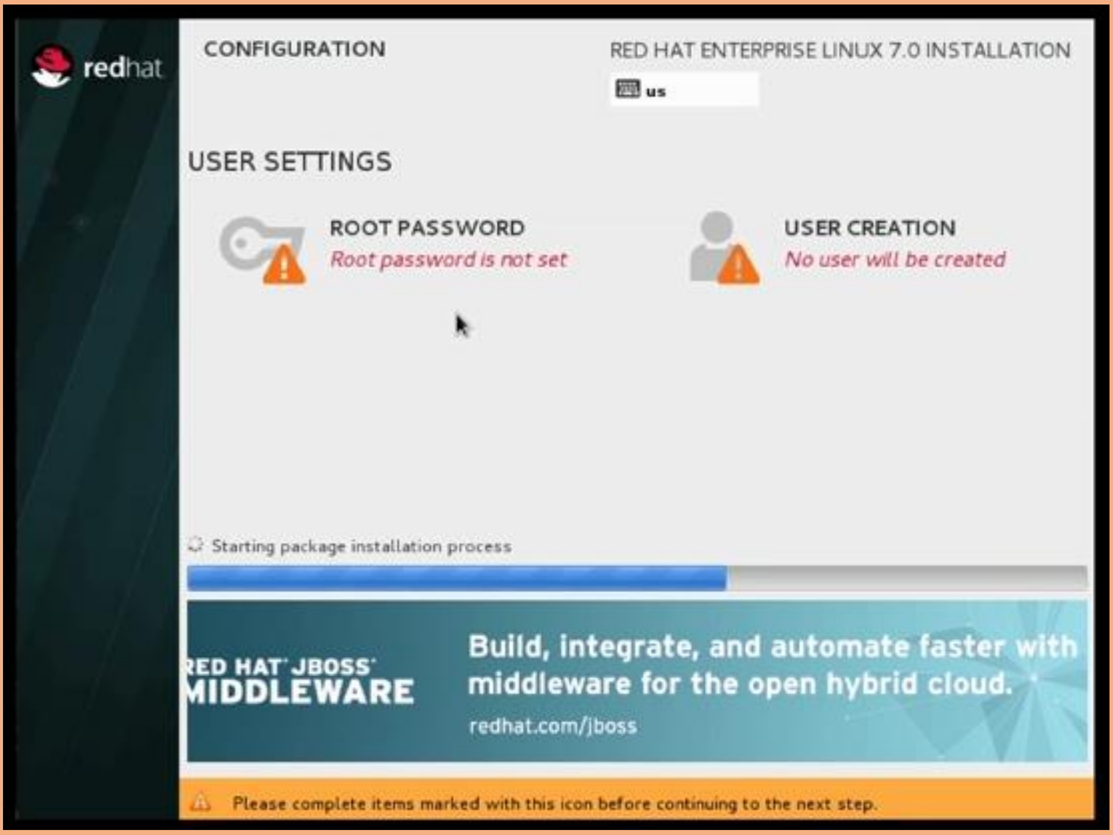
SYSTEM

INSTALLATION DESTINATION
Automatic partitioning selected

NETWORK & HOSTNAME
Wired (ens33) connected

We won't touch your disks until you click 'Begin Installation'.

9. Give Root password and create user



CONFIGURATION

RED HAT ENTERPRISE LINUX 7.0 INSTALLATION

USER SETTINGS

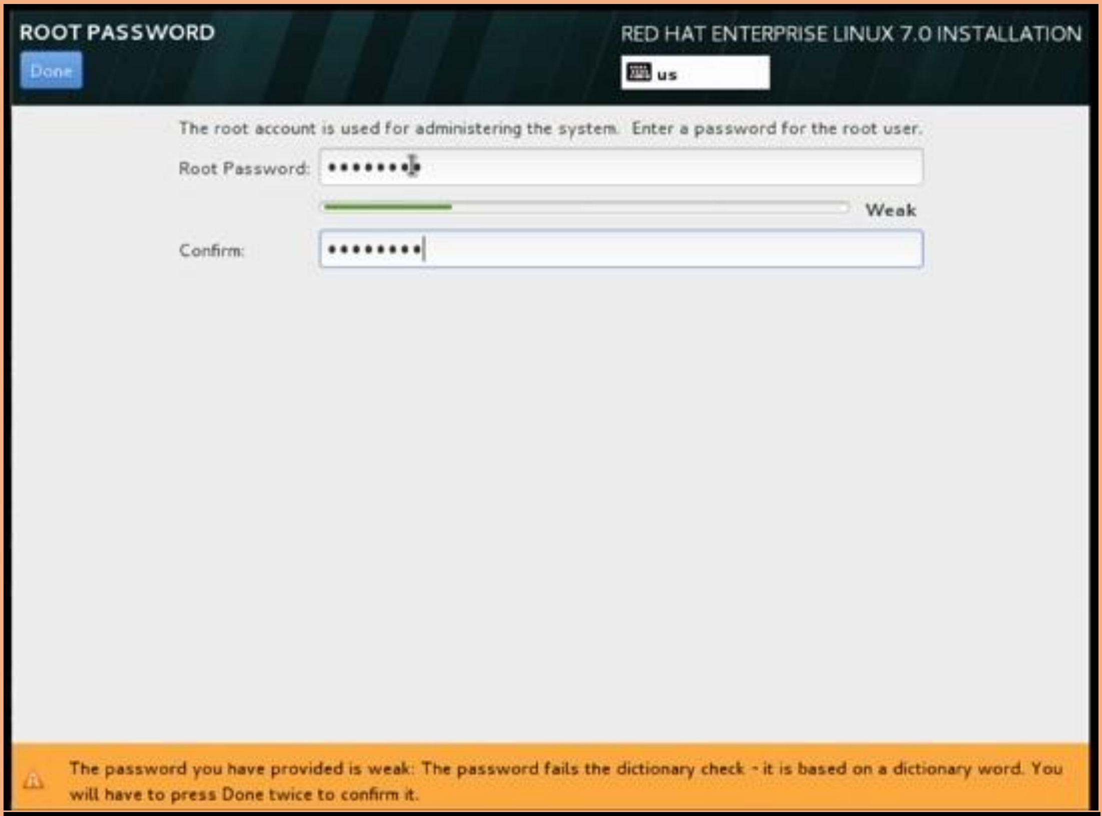
ROOT PASSWORD
Root password is not set

USER CREATION
No user will be created

Starting package installation process

⚠ Please complete items marked with this icon before continuing to the next step.

redhat
us
RED HAT JBOSS MIDDLEWARE
Build, integrate, and automate faster with middleware for the open hybrid cloud.
redhat.com/jboss



ROOT PASSWORD

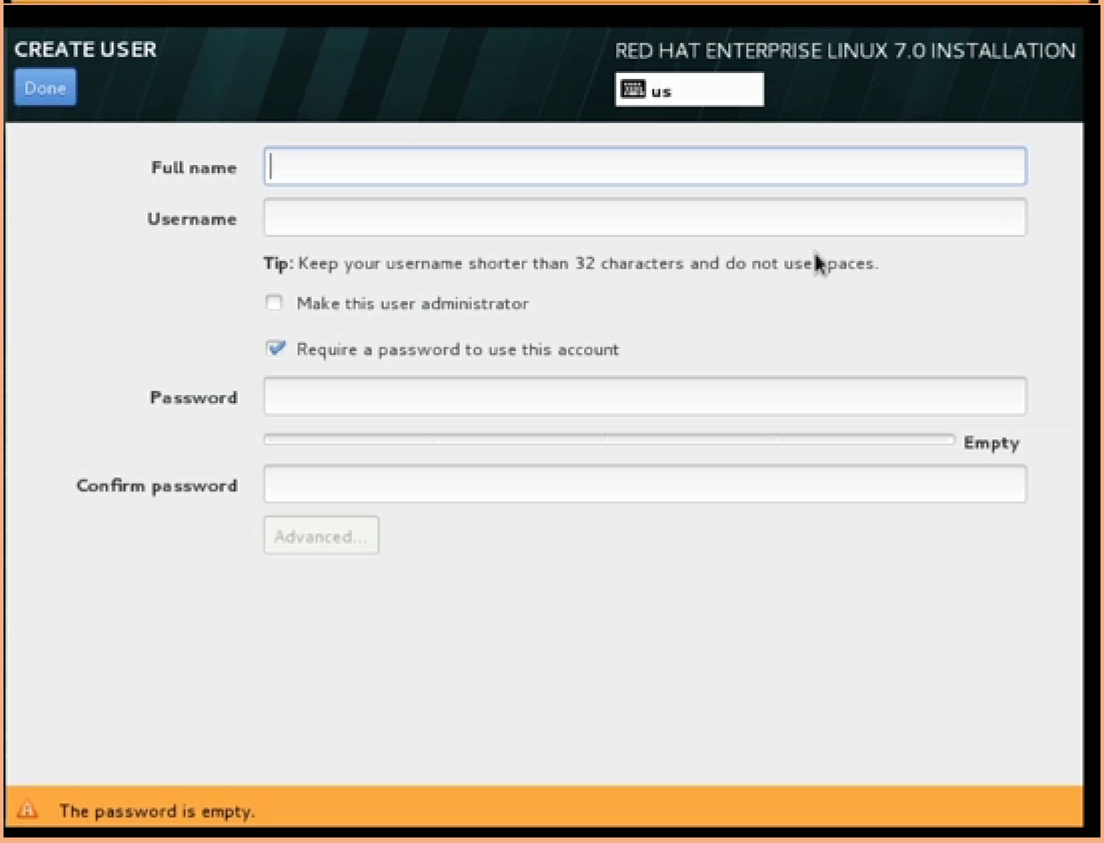
The root account is used for administering the system. Enter a password for the root user.

Root Password: ••••••••

Weak

Confirm: •••••••••

⚠ The password you have provided is weak: The password fails the dictionary check - it is based on a dictionary word. You will have to press Done twice to confirm it.



CREATE USER

Full name

Username

Tip: Keep your username shorter than 32 characters and do not use spaces.

□ Make this user administrator

☑ Require a password to use this account

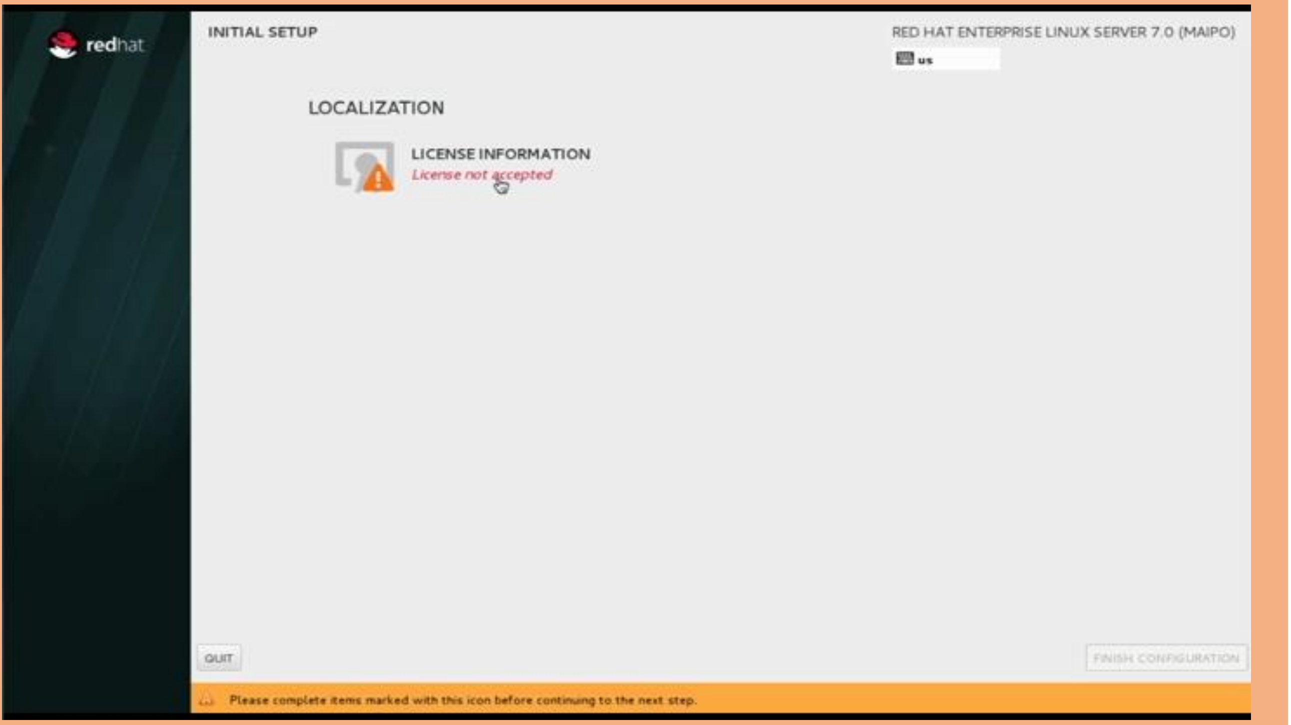
Password

Empty

Confirm password

⚠ The password is empty.

10. Accept the license agreement, and click on finish configuration



INITIAL SETUP

RED HAT ENTERPRISE LINUX SERVER 7.0 (MAIPO)

us

LOCALIZATION

LICENSE INFORMATION
License not accepted

QUIT

FINISH CONFIGURATION

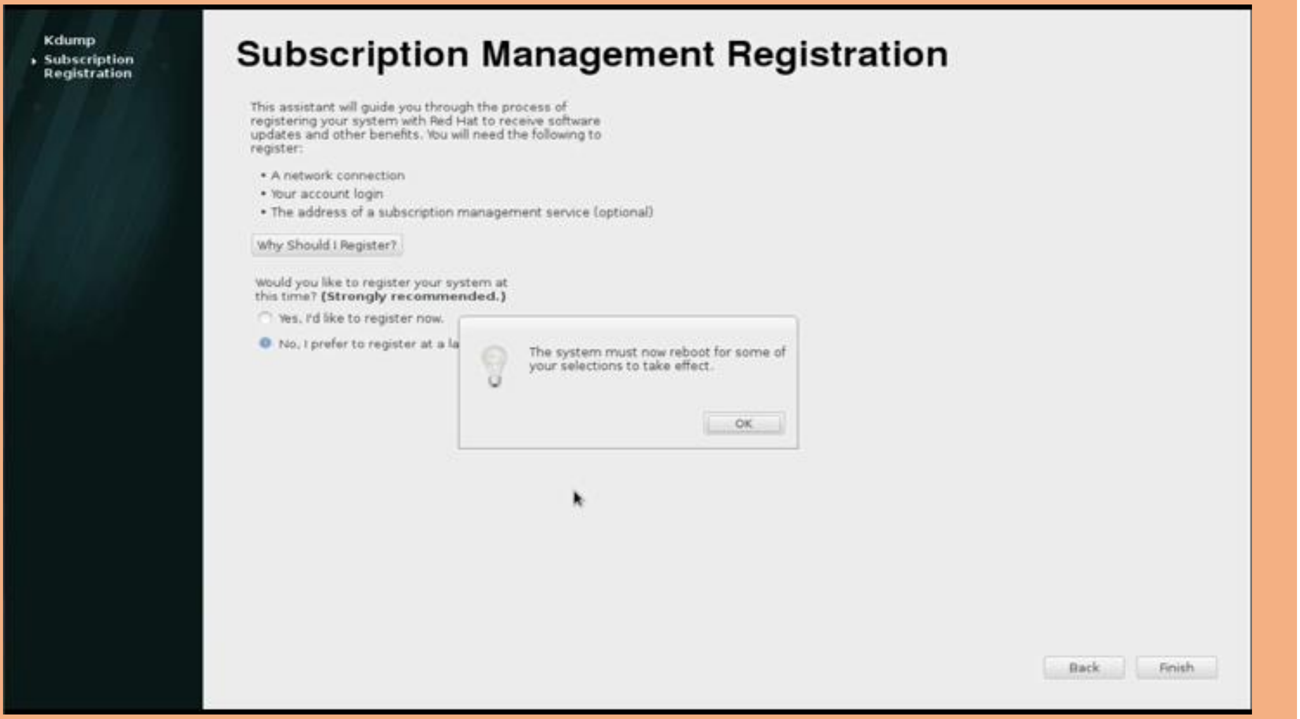
Please complete items marked with this icon before continuing to the next step.

11. Now if you want paid license so Register Red Hat , so click on

◯Yes, I'd like to register now,

Otherwise click on

◯ No, I prefer to register at a later time. And click on finish.



Kdump
Subscription Registration

Subscription Management Registration

This assistant will guide you through the process of registering your system with Red Hat to receive software updates and other benefits. You will need the following to register:

- A network connection
- Your account login
- The address of a subscription management service (optional)

Why Should I Register?

Would you like to register your system at this time? **(Strongly recommended.)**

○ Yes, I'd like to register now.

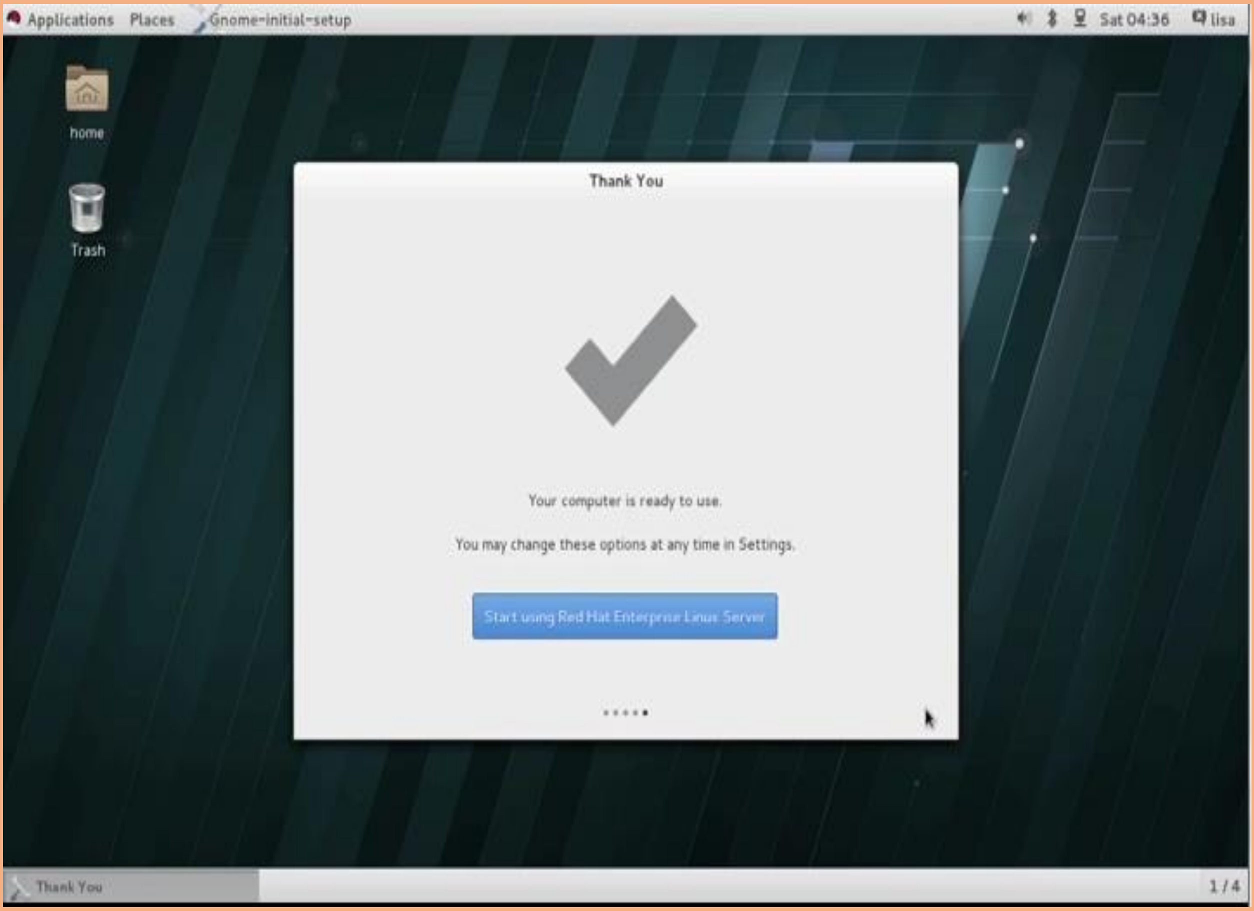
● No, I prefer to register at a la

The system must now reboot for some of your selections to take effect.

OK

Back Finish

12. Your RHEL 7 is ready for use.



Thank You

Your computer is ready to use.

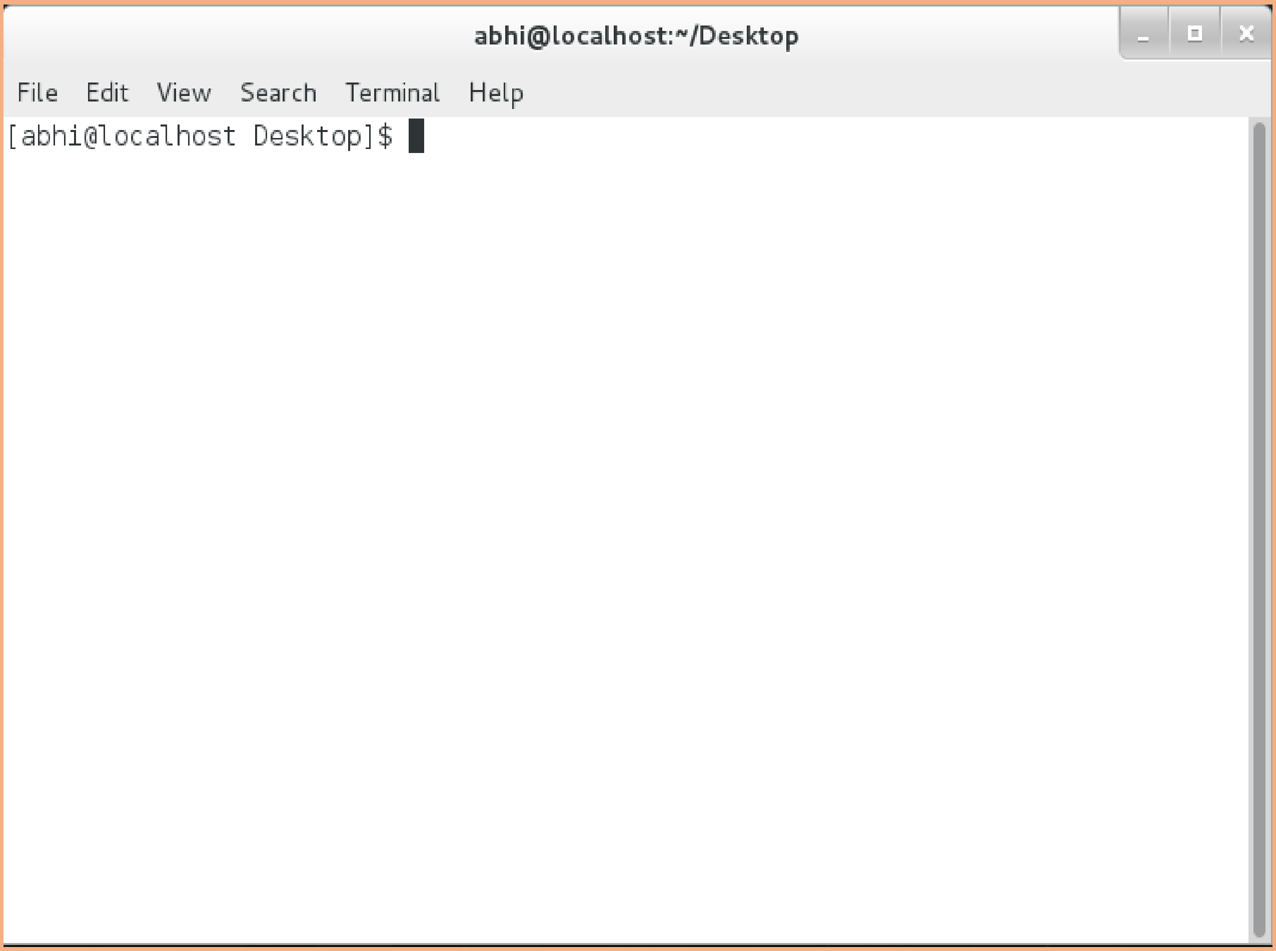
You may change these options at any time in Settings.

How to access CLI Interface

CLI is a text Based interface which can be used to input instructions to a computer system. The linux command line is provided by a program called the shell. The default shell for users in Red Hat Enterprise Linux is the GNU Bourne-Again Shell (bash). Bash is an improved version of one of the most successful shells used on UNIX-like systems, the Bourne Shell (sh).

For Access bash go to Application menu ➡ Utilities ➡ Terminal

Users access the bash shell through a terminal.



```
[abhi@localhost Desktop]$ 
```

GNOME-Shell (GNU Network Object Model Environment)

Graphical shell of Red Hat By default is GNOME-Shell. It is a GUI interface on a Red Hat Linux. We using latest version of GNOME is GNOME 3.

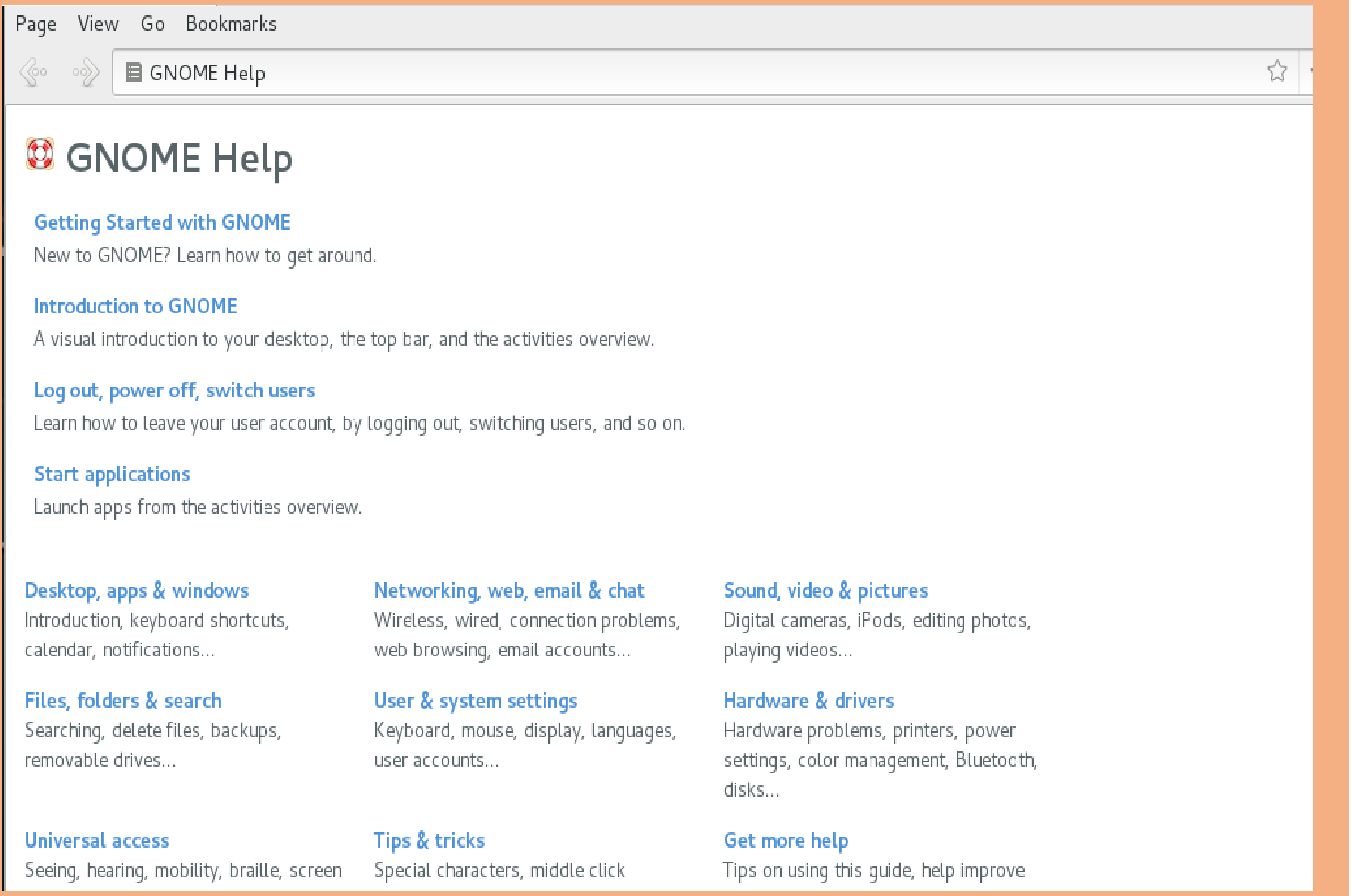
For GNOME Help press F1

Or

Go to Application ➡ Documentation ➡ Help.

Or

By running yelp command on terminal.



GNOME Help

Getting Started with GNOME
New to GNOME? Learn how to get around.

Introduction to GNOME
A visual introduction to your desktop, the top bar, and the activities overview.

Log out, power off, switch users
Learn how to leave your user account, by logging out, switching users, and so on.

Start applications
Launch apps from the activities overview.

Desktop, apps & windows
Introduction, keyboard shortcuts, calendar, notifications…

Networking, web, email & chat
Wireless, wired, connection problems, web browsing, email accounts…

Sound, video & pictures
Digital cameras, iPods, editing photos, playing videos…

Files, folders & search
Searching, delete files, backups, removable drives…

User & system settings
Keyboard, mouse, display, languages, user accounts…

Hardware & drivers
Hardware problems, printers, power settings, color management, Bluetooth, disks…

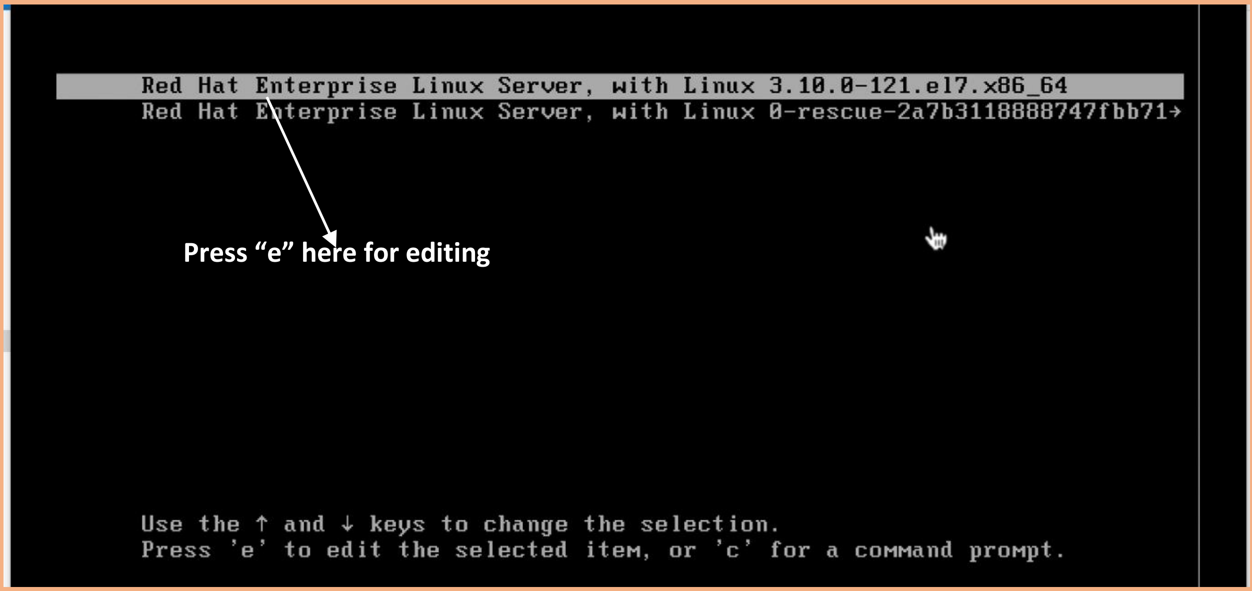
Universal access
Seeing, hearing, mobility, braille, screen

Tips & tricks
Special characters, middle click

Get more help
Tips on using this guide, help improve

Password Cracking through rd.break

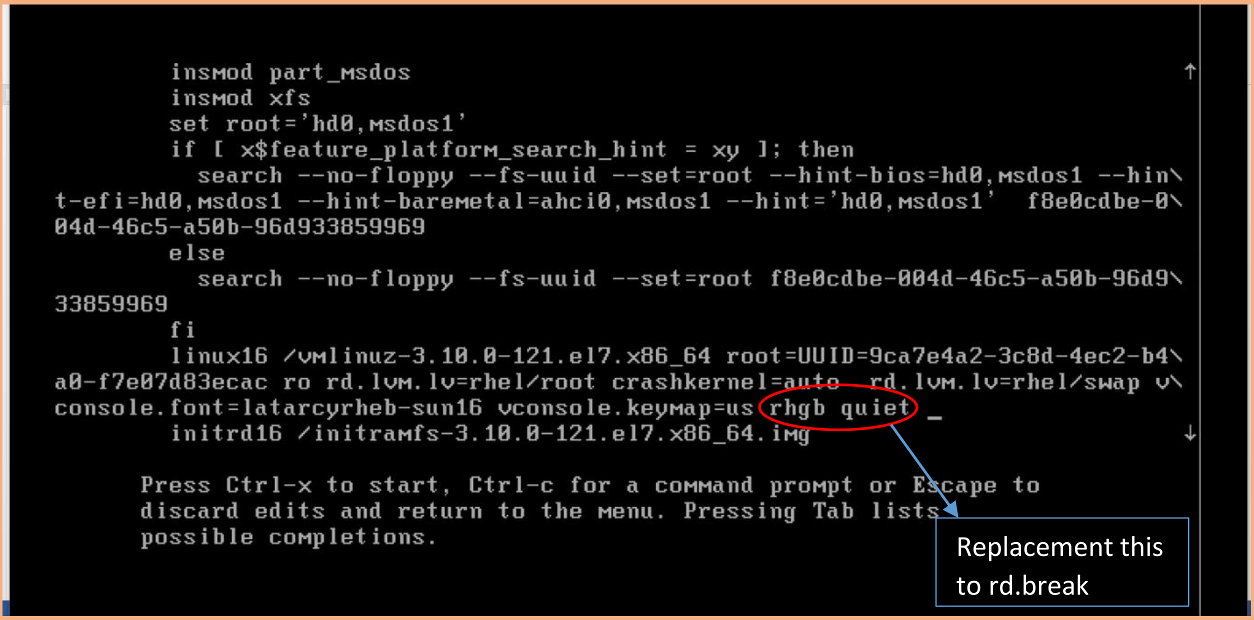
1. Start your system
2. And press “e” on normal mode



```
      Red Hat Enterprise Linux Server, with Linux 3.10.0-121.el7.x86_64
      Red Hat Enterprise Linux Server, with Linux 0-rescue-2a7b3118888747fbb71→






      Use the ↑ and ↓ keys to change the selection.
      Press 'e' to edit the selected item, or 'c' for a command prompt.
```

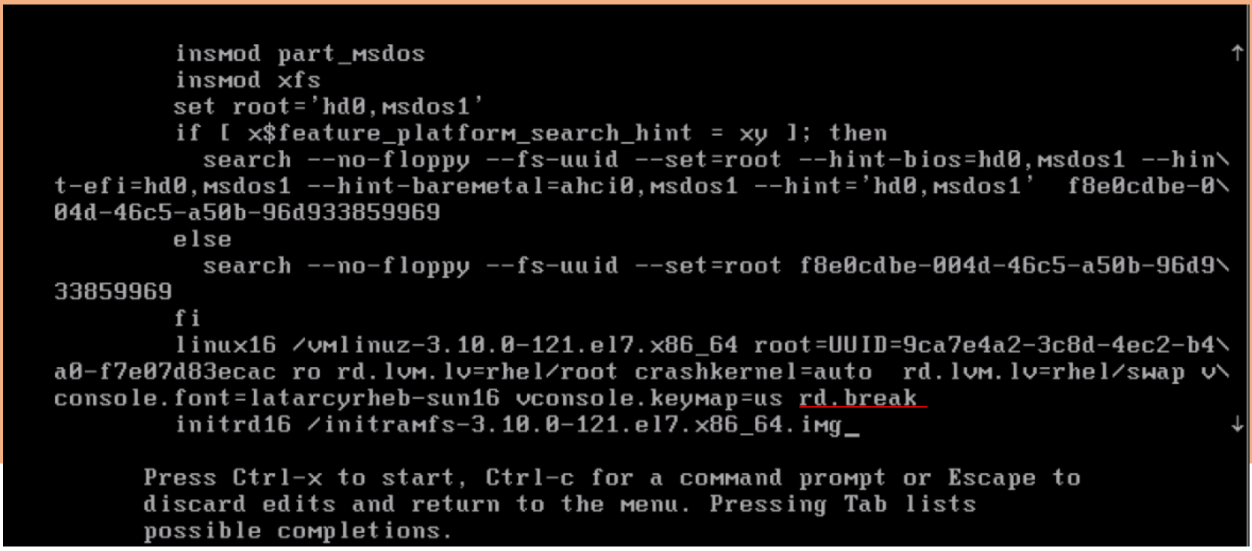
Press “e” here for editing

3. And put rd.break



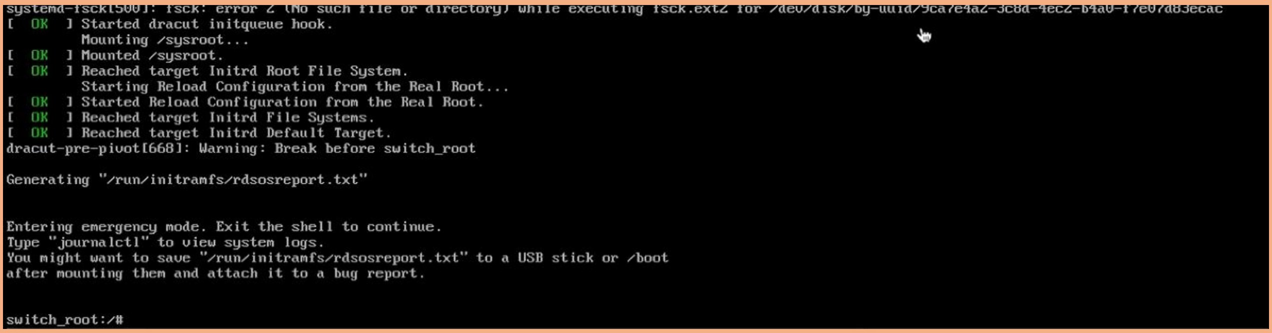
```
        insmod part_msdos
        insmod xfs
        set root='hd0,msdos1'
        if [ x$feature_platform_search_hint = xy ]; then
          search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hin\
t-efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1 --hint='hd0,msdos1'  f8e0cdbe-0\
04d-46c5-a50b-96d933859969
        else
          search --no-floppy --fs-uuid --set=root f8e0cdbe-004d-46c5-a50b-96d9\
33859969
        fi
        linux16 /vmlinuz-3.10.0-121.el7.x86_64 root=UUID=9ca7e4a2-3c8d-4ec2-b4\
a0-f7e07d83ecac ro rd.lvm.lv=rhel/root crashkernel=auto  rd.lvm.lv=rhel/swap v\
console.font=latarcyrheb-sun16 vconsole.keymap=us rhgb quiet _
        initrd16 /initramfs-3.10.0-121.el7.x86_64.img

      Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
      discard edits and return to the menu. Pressing Tab lists
      possible completions.
```

Replacement this to rd.break



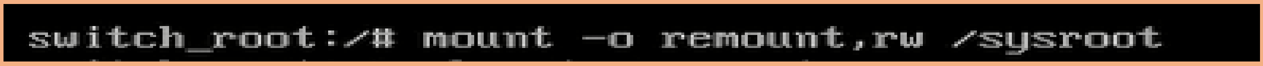
```
        insmod part_msdos
        insmod xfs
        set root='hd0,msdos1'
        if [ x$feature_platform_search_hint = xy ]; then
          search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hin\
t-efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1 --hint='hd0,msdos1'  f8e0cdbe-0\
04d-46c5-a50b-96d933859969
        else
          search --no-floppy --fs-uuid --set=root f8e0cdbe-004d-46c5-a50b-96d9\
33859969
        fi
        linux16 /vmlinuz-3.10.0-121.el7.x86_64 root=UUID=9ca7e4a2-3c8d-4ec2-b4\
a0-f7e07d83ecac ro rd.lvm.lv=rhel/root crashkernel=auto  rd.lvm.lv=rhel/swap v\
console.font=latarcyrheb-sun16 vconsole.keymap=us rd.break
        initrd16 /initramfs-3.10.0-121.el7.x86_64.img_

      Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
      discard edits and return to the menu. Pressing Tab lists
      possible completions.
```

4. Now press ctrl+x



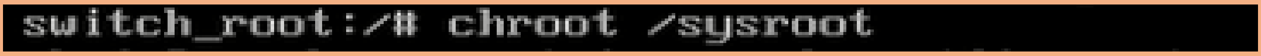
```
systemd-fsck[500]: fsck: error 2 (No such file or directory) while executing fsck.ext2 for /dev/disk/by-uuid/9ca7e4a2-3c8d-4ec2-b4a0-f7e07d83ecac
[  OK  ] Started dracut initqueue hook.
         Mounting /sysroot...
[  OK  ] Mounted /sysroot.
[  OK  ] Reached target Initrd Root File System.
         Starting Reload Configuration from the Real Root...
[  OK  ] Started Reload Configuration from the Real Root.
[  OK  ] Reached target Initrd File Systems.
[  OK  ] Reached target Initrd Default Target.
dracut-pre-pivot[668]: Warning: Break before switch_root

Generating "/run/initramfs/rdsosreport.txt"


Entering emergency mode. Exit the shell to continue.
Type "journalctl" to view system logs.
You might want to save "/run/initramfs/rdsosreport.txt" to a USB stick or /boot
after mounting them and attach it to a bug report.


switch_root:/#
```

5. Now run this command “mountØ-oØremount,rwØ/sysroot”



```
switch_root:/# mount -o remount,rw /sysroot
```

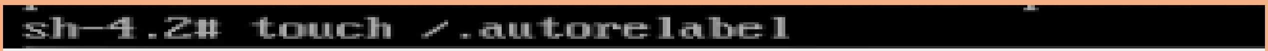
6. Now run this command to change shell “chrootØsysroot”



```
switch_root:/# chroot /sysroot
```

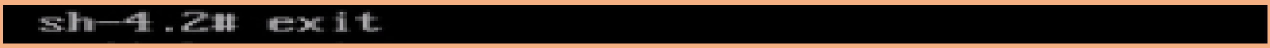
7. Now change the password of root
   Sh-4.2# passwdØroot

8. Now relabel selinux



```
sh-4.2# touch /.autorelabel
```

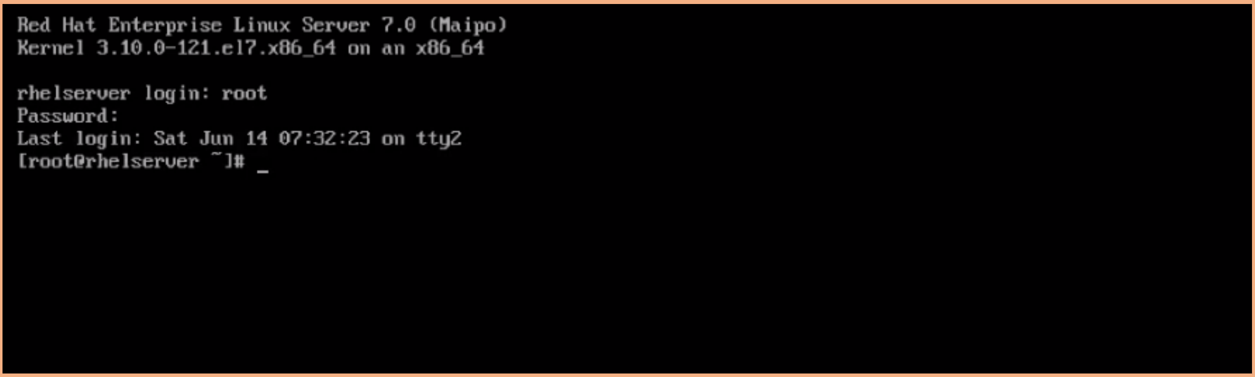
9. Now exit from shell



```
sh-4.2# exit
```

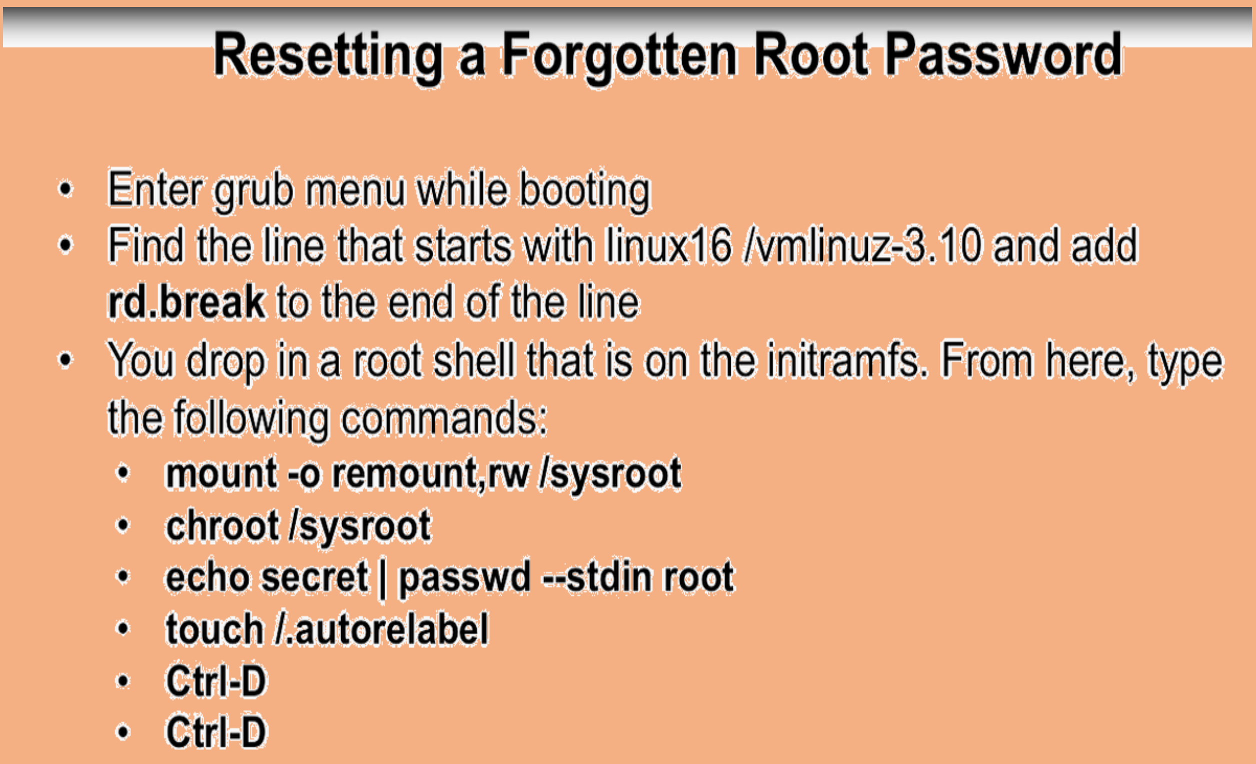
10. Now logout from the user

11. And relogin to root from new password



```
Red Hat Enterprise Linux Server 7.0 (Maipo)
Kernel 3.10.0-121.el7.x86_64 on an x86_64

rhelserver login: root
Password:
Last login: Sat Jun 14 07:32:23 on tty2
[root@rhelserver ~]# _
```

Summary



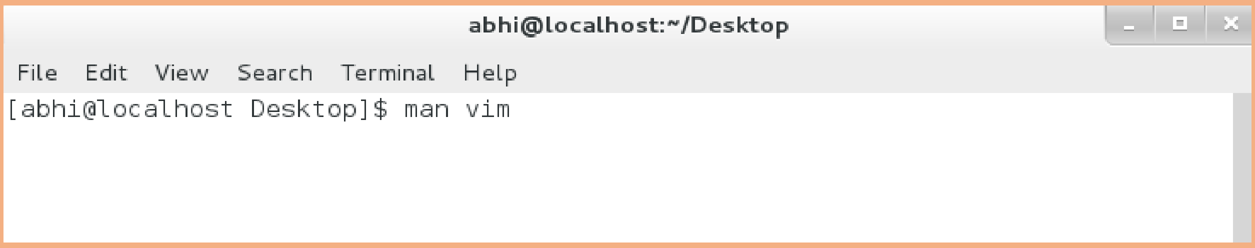
Resetting a Forgotten Root Password

- Enter grub menu while booting
- Find the line that starts with linux16 /vmlinuz-3.10 and add **rd.break** to the end of the line
- You drop in a root shell that is on the initramfs. From here, type the following commands:
  - **mount -o remount,rw /sysroot**
  - **chroot /sysroot**
  - **echo secret | passwd --stdin root**
  - **touch /.autorelabel**
  - **Ctrl-D**
  - **Ctrl-D**

Getting Help in Redhat Enterprise Linux

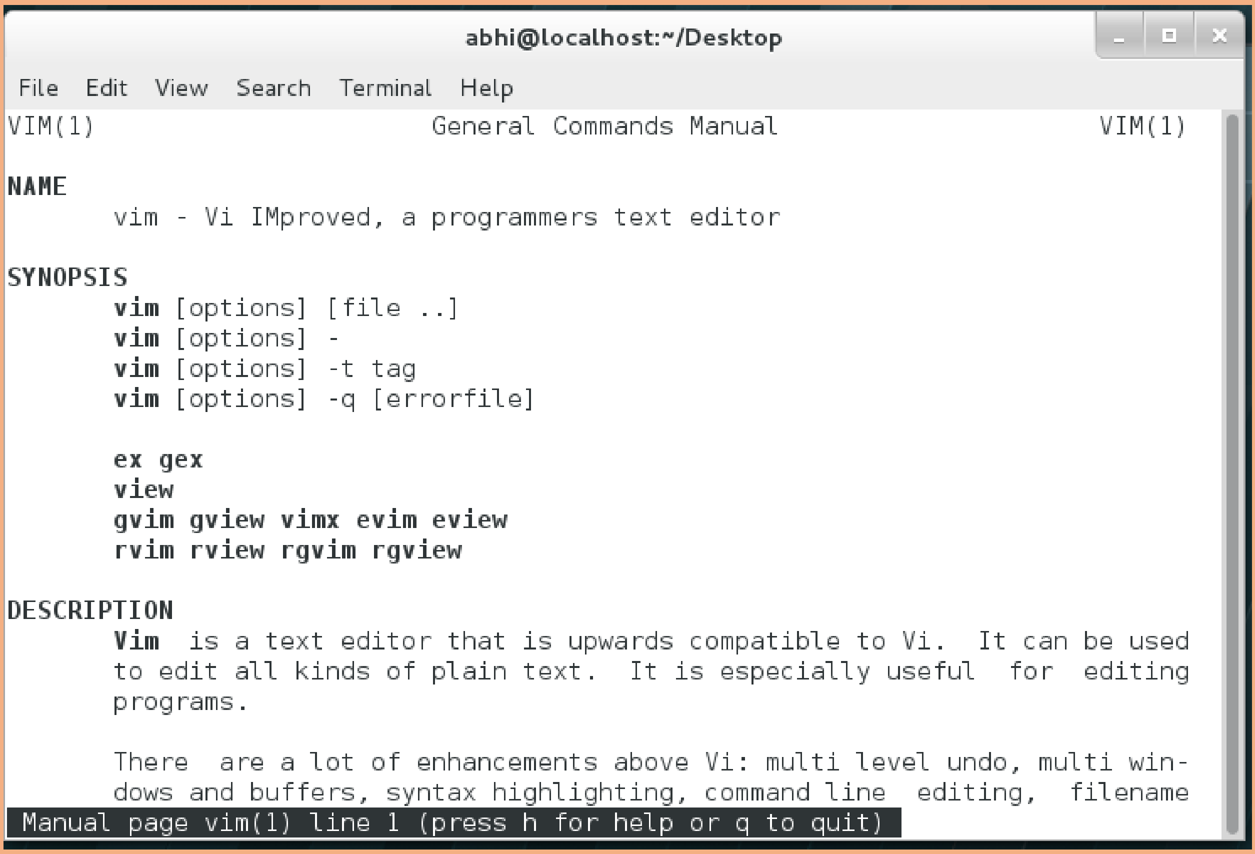
Reading Documentation Using man command

man command is work as a manual guide and it is use for get information about any packages ,service or command. Means how to use any command for execute any task, like if you want know what is “vim” and how to use, so run below command



```
[abhi@localhost Desktop]$ man vim
```

Your Answer:

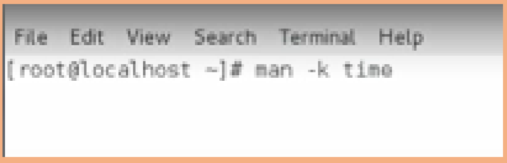


```
VIM(1)                    General Commands Manual                   VIM(1)

NAME
       vim - Vi IMproved, a programmers text editor

SYNOPSIS
       vim [options] [file ..]
       vim [options] -
       vim [options] -t tag
       vim [options] -q [errorfile]

       ex gex
       view
       gvim gview vimx evim eview
       rvim rview rgvim rgview

DESCRIPTION
       Vim  is a text editor that is upwards compatible to Vi.  It can be used
       to edit all kinds of plain text.  It is especially useful  for  editing
       programs.

       There  are a lot of enhancements above Vi: multi level undo, multi win-
       dows and buffers, syntax highlighting, command line  editing,  filename
 Manual page vim(1) line 1 (press h for help or q to quit)
```

If you want set time and don’t what is right command to set time, so can get information by keyword with man command.

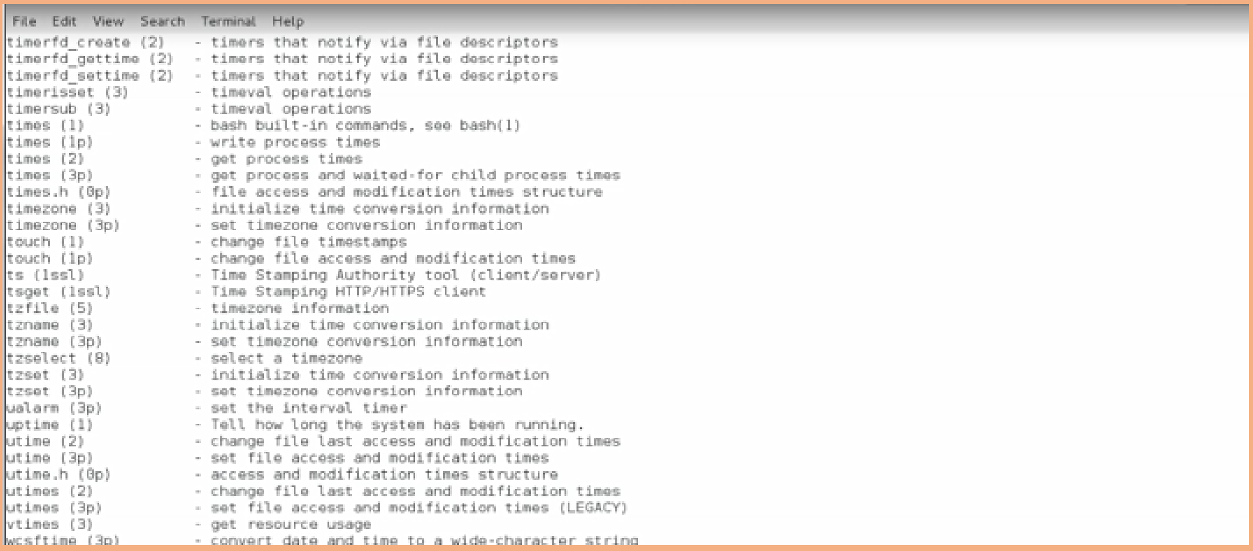
# man Ø –k time

-k is a option for keyword



```
[root@localhost ~]# man -k time
```

Your Answer:



```
timerfd_create (2)    - timers that notify via file descriptors
timerfd_gettime (2)   - timers that notify via file descriptors
timerfd_settime (2)   - timers that notify via file descriptors
timerisset (3)        - timeval operations
timersub (3)          - timeval operations
times (1)             - bash built-in commands, see bash(1)
times (1p)            - write process times
times (2)             - get process times
times (3p)            - get process and waited-for child process times
times.h (0p)          - file access and modification times structure
timezone (3)          - initialize time conversion information
timezone (3p)         - set timezone conversion information
touch (1)             - change file timestamps
touch (1p)            - change file access and modification times
ts (1ssl)             - Time Stamping Authority tool (client/server)
tsget (1ssl)          - Time Stamping HTTP/HTTPS client
tzfile (5)            - timezone information
tzname (3)            - initialize time conversion information
tzname (3p)           - set timezone conversion information
tzselect (8)          - select a timezone
tzset (3)             - initialize time conversion information
tzset (3p)            - set timezone conversion information
ualarm (3p)           - set the interval timer
uptime (1)            - Tell how long the system has been running.
utime (2)             - change file last access and modification times
utime (3p)            - set file access and modification times
utime.h (0p)          - access and modification times structure
utimes (2)            - change file last access and modification times
utimes (3p)           - set file access and modification times (LEGACY)
vtimes (3)            - get resource usage
wcsftime (3p)         - convert date and time to a wide-character string
```

Understanding “vim”

Vim Editor:

vim is a file editor. It is a improved version of “VI”, it use to create, read and edit any file.

1. Command Mode :- In this mode you can edit the files using basic commands.
2. Insert Mode :- In this mode you can edit the file normally.
3. ESC Mode :- this mode is used to escape from certain mode. Eg. Escape from insert mode and go to command mode.

Command Mode Commands

1. x	To delete the character.
2. alt+u	To undo the changes.
3. $	To take the cursor end of the line.
4. ^	To take the cursor start of the line.
5. {	To take the cursor start of the paragraph.
6. }	To take the cursor end of the paragraph.
7. (	To take the cursor start of the sentence.
8.)	To take the cursor end of the sentence.
9. w	To take the cursor next word.
10. b	To take the cursor back word.
11. k	Used for up arrow key.
12. j	Used for down arrow key.
13. h	Used for left arrow key.
14. l	Used for right arrow key.
15. yy	To copy a specific no.’s of lines.
16.p	To paste copied lines.
17. dd	To delete specific no.’s of lines.
18. shift+r	To replace the line
19. r	To replace the character.
20. o	To create a blank line below the cursor.
21. O	To create a blank line above the cursor.
22. I	Go to Insert mode
23. :q	To exit without saving the file.
24. :q!	To exit without saving the file forcefully.

25. :wq	To save and exit the file.
26. :wq!	To save and exit the file forcefully.

VI editor Tutorial

1. To create file using Vim

```
[root@server1~]# vim Øfilename
<PRESS I>to inter into insert mode and type the following text in it

Hello!!!!!!!!
Good Morning Everyone
Welcome to Aegis

TO EXIT FROM INSERT MODE
:wq!   TO SAVE AND EXIT
[root@server1~]# cat Ø filename
Hello!!!!!!!!
Good Morning Everyone
Welcome to Aegis
```

2. To Copy and Paste the lines using Vim editor

```
[root@server1~]# vimØfilename
Hello!!!!!!!!
Good Morning Everyone              move your cursor to specific line
                                   which you want to copy and type
                                   “yy” and move your cursor were you
                                   want to paste that line
Welcome to Aegis                   and type p
Good Morning Everyone

PRESS  ESC   TO EXIT FROM INSERT MODE
:wq!             TO SAVE AND EXIT
[root@server1~]#
```

3. To Delete the lines using Vi editor

[root@server1~]# vim Øfilename ………<PRESS ENTER>

Hello!!!!!!!!

Good Morning Everyone

move your cursor to specific line which you want to delete and type dd

Welcome to Aegis

Good Morning Everyone

PRESS ESC> TO EXIT FROM INSERT MODE

:wq! TO SAVE AND EXIT

[root@server1~]#

4. Move the Cursor end of the line.

[root@server1~]# vimØ filename

Hello!!!!!!!!

Good Morning Everyone

take your cursor start of the line and type shift+$ to move end on the line

Welcome to Aegis

<PRESS ESC> TO EXIT FROM INSERT MODE

:wq! TO SAVE AND EXIT

5. Move the Cursor start of the line.

[root@server1~]# vimØ filename

Hello!!!!!!!!

Good Morning Everyone

take your cursor start of the line and type shift+^ to move start on the line

Welcome to Aegis

<PRESS ESC> TO EXIT FROM INSERT MODE

:wq! TO SAVE AND EXIT

6. Move the Cursor end of the paragraph.

```
[root@server1~]# vimØ  filename        .........<PRESS ENTER>

Hello!!!!!!!!
Good Morning Everyone


Welcome to Aegis

<PRESS  ESC>      TO EXIT FROM INSERT MODE
:wq!              TO SAVE AND EXIT
```

take your cursor start of the line and type shift+} to move end on the paragraph

7. Move the Cursor start of the paragraph.

```
[root@server1~]# vimØ  filename
Hello!!!!!!!!
Good Morning Everyone


Welcome to Aegis

<PRESS  ESC>      <TO EXIT FROM INSERT MODE>
:wq!               <TO SAVE AND EXIT>
```

take your cursor end of the paragraph and type shift+{ to move start of the paragraph

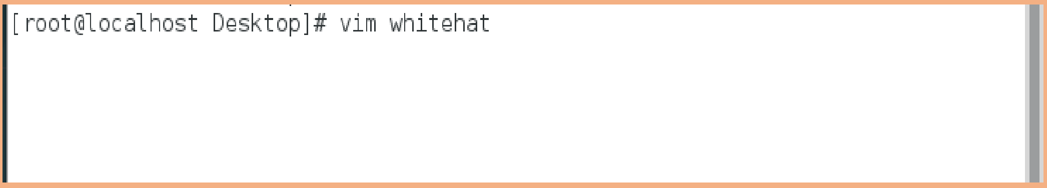
To replace the character.

```
[root@server1~]#vimØ  filename

Welcome to Aeges

Welcome to Aegis
:wq!
```

move your cursor to specific word which you want to replace and press r and do the changes

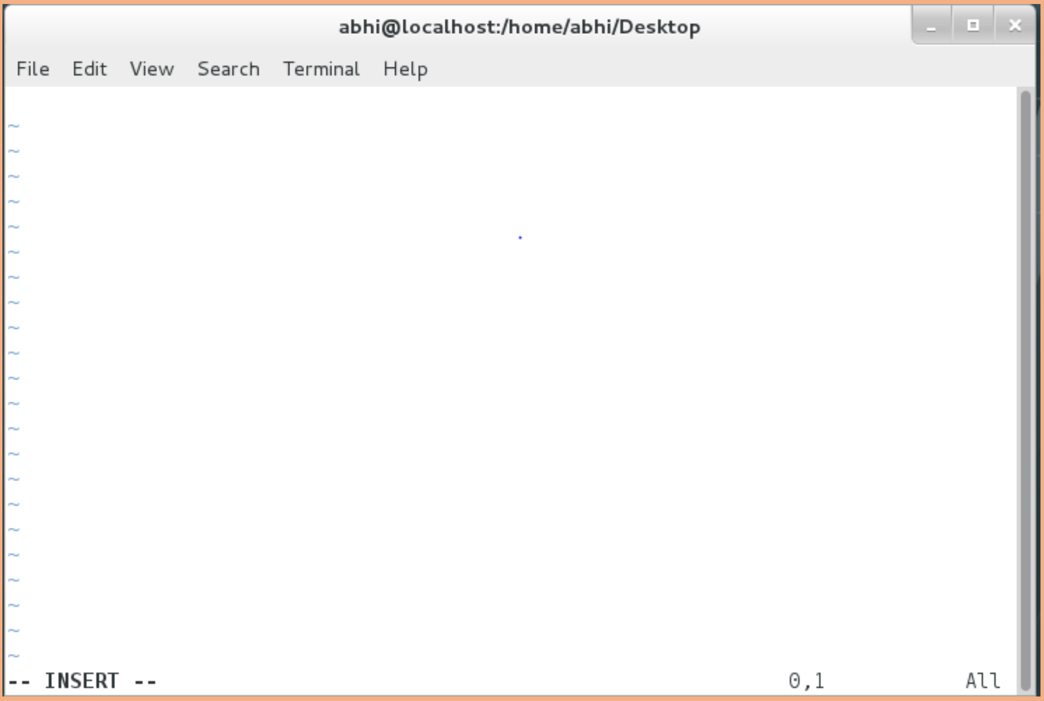
Editing text file with “vim”

# vim “file name”



```
[root@localhost Desktop]# vim whitehat
```

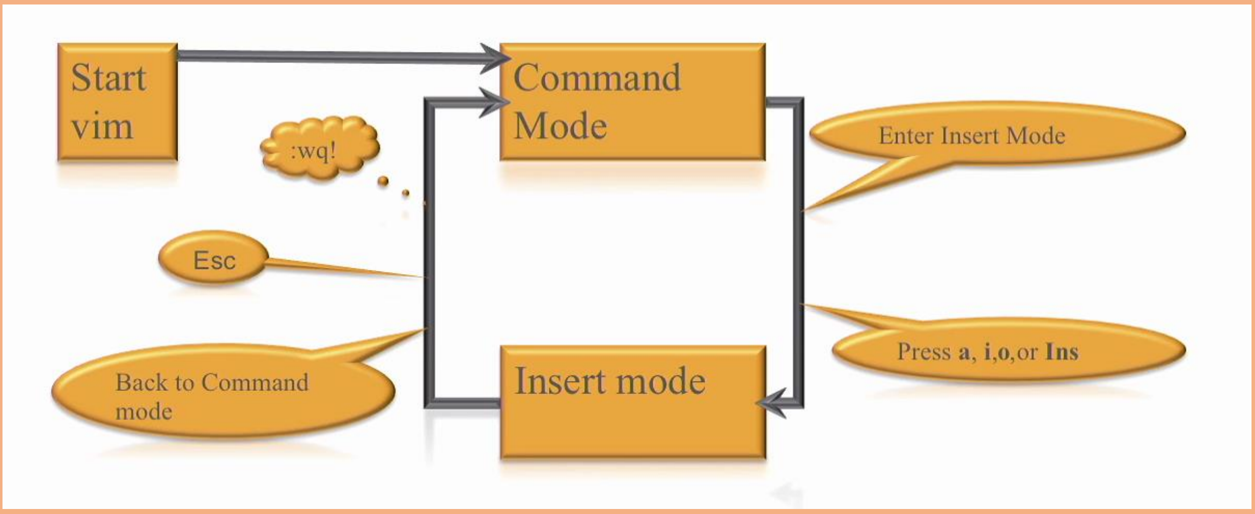
Now press “i” for enable typing in to the file



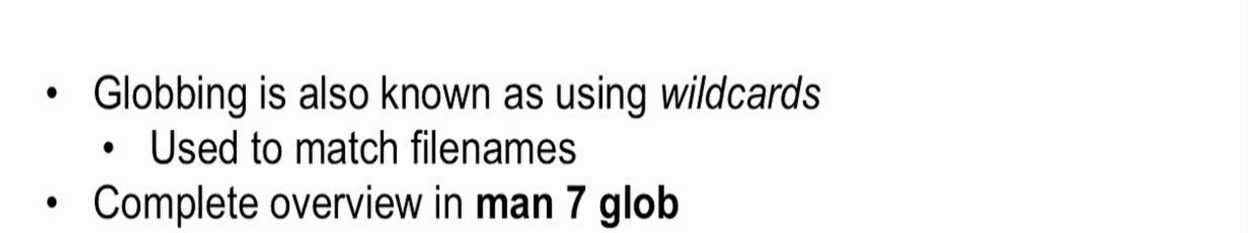
abhi@localhost:/home/abhi/Desktop
File Edit View Search Terminal Help

```
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                                0,1           All
```

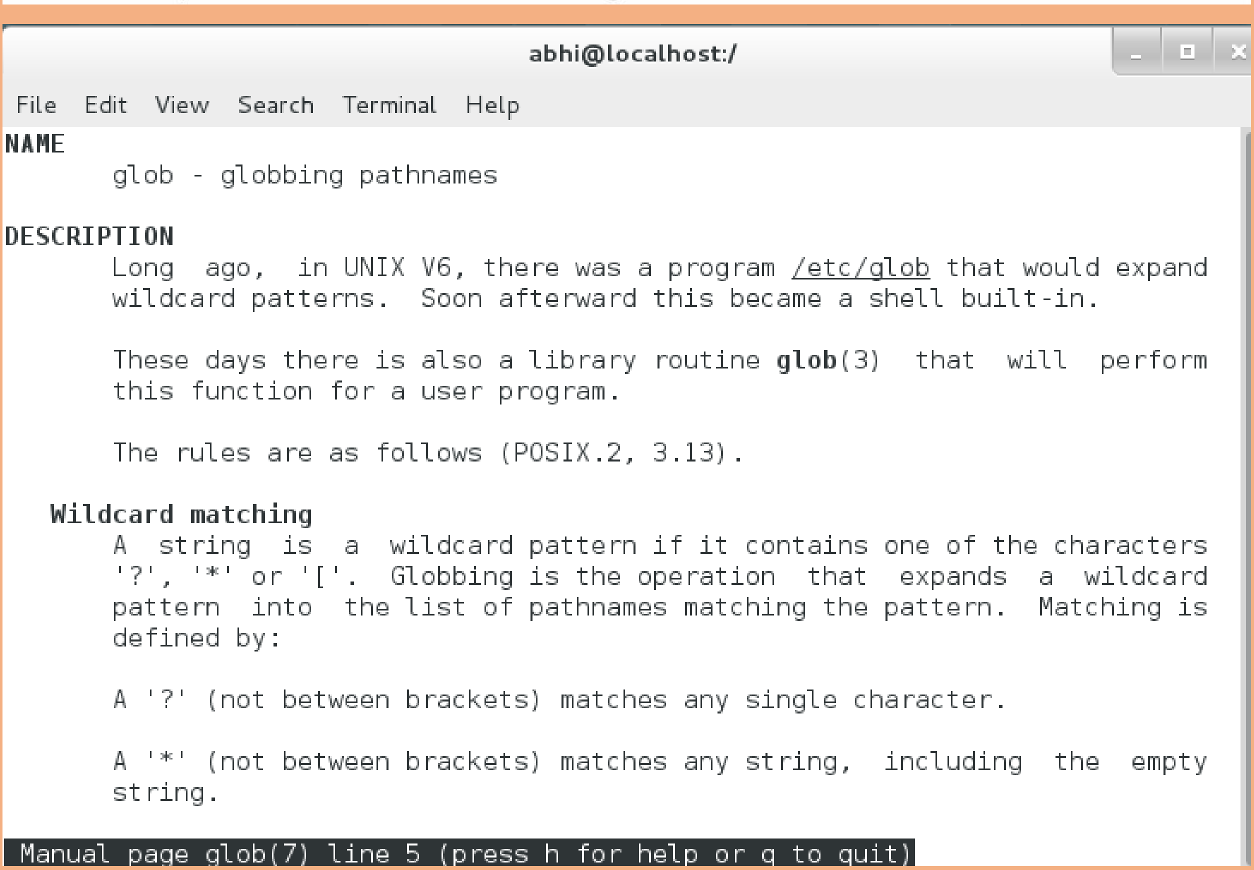
To save file press “Esc” key and “: wq”.



Understanding globing and wildcard



- Globbing is also known as using *wildcards*
  - Used to match filenames
- Complete overview in **man 7 glob**

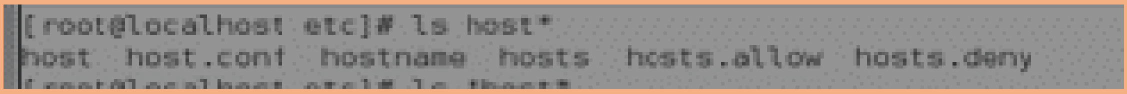


```
NAME
       glob - globbing pathnames

DESCRIPTION
       Long  ago,  in UNIX V6, there was a program /etc/glob that would expand
       wildcard patterns.  Soon afterward this became a shell built-in.

       These days there is also a library routine glob(3)  that  will  perform
       this function for a user program.

       The rules are as follows (POSIX.2, 3.13).

   Wildcard matching
       A  string  is  a  wildcard pattern if it contains one of the characters
       '?', '*' or '['.  Globbing is the operation  that  expands  a  wildcard
       pattern  into  the list of pathnames matching the pattern.  Matching is
       defined by:

       A '?' (not between brackets) matches any single character.

       A '*' (not between brackets) matches any string,  including  the  empty
       string.

 Manual page glob(7) line 5 (press h for help or q to quit)
```

Using Globbing and wildcard

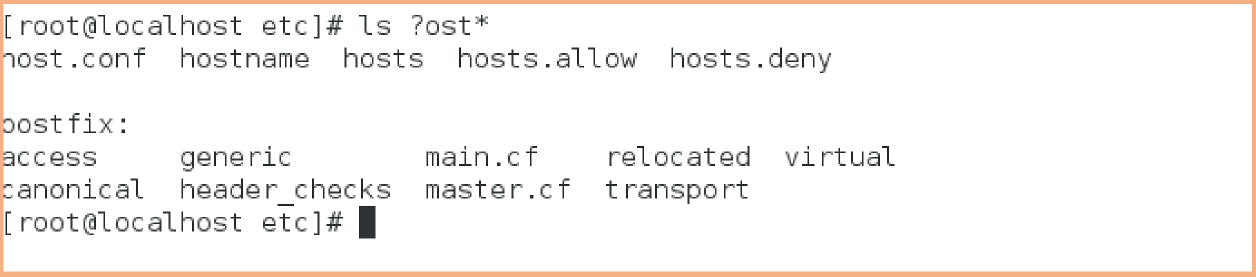
#ls file or folder name

#ls host*



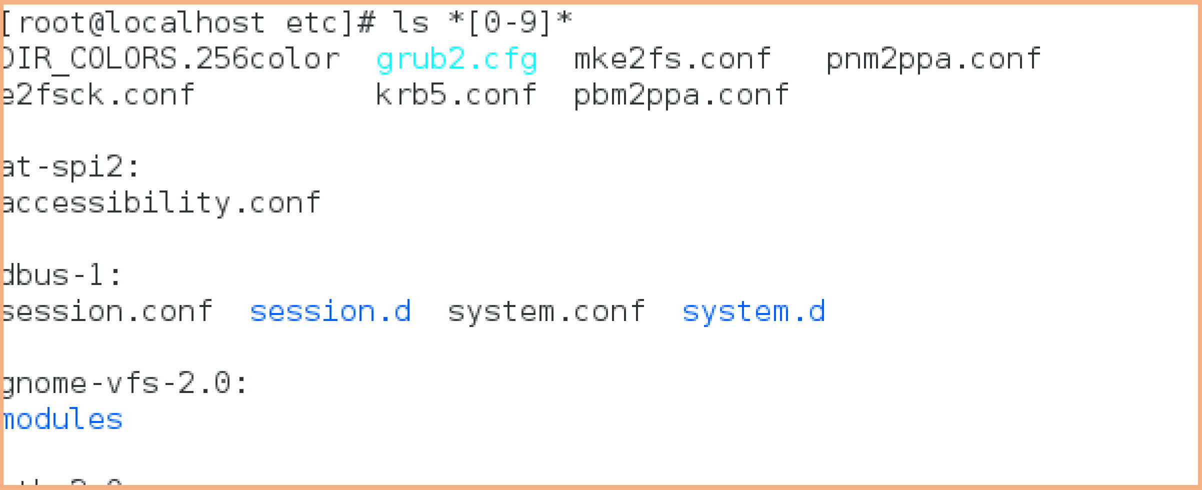
```
[root@localhost etc]# ls host*
host  host.conf  hostname  hosts  hosts.allow  hosts.deny
```

ls ?ost*



```
[root@localhost etc]# ls ?ost*
host.conf  hostname  hosts  hosts.allow  hosts.deny

postfix:
access     generic        main.cf    relocated  virtual
canonical  header_checks  master.cf  transport
[root@localhost etc]# 
```

ls *[0-9]*



```
[root@localhost etc]# ls *[0-9]*
DIR_COLORS.256color  grub2.cfg  mke2fs.conf   pnm2ppa.conf
e2fsck.conf          krb5.conf  pbm2ppa.conf

at-spi2:
accessibility.conf

dbus-1:
session.conf  session.d  system.conf  system.d

gnome-vfs-2.0:
modules
```

Configuring the Date and Time

In latest operating system distinguish between two types of clocks

1. A real-time clock (RTC), commonly referred to as a hardware clock, (typically an integrated circuit on the system board) that is completely independent of the current state of the operating system and runs even when the computer is shut down.

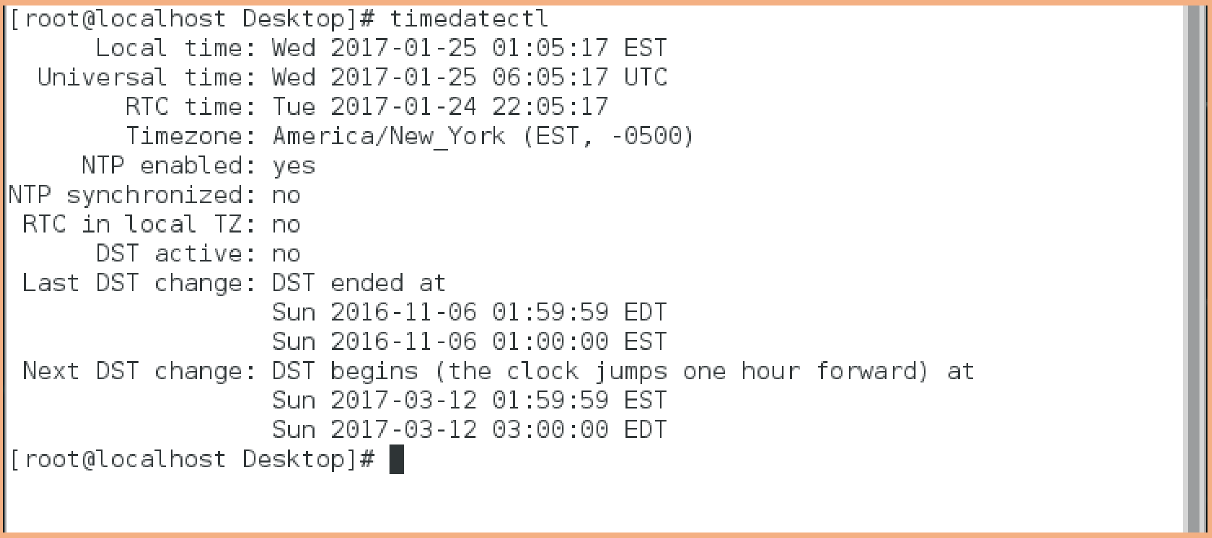
2. A system clock, also known as a software clock, that is maintained by the kernel and its initial value is based on the real-time clock. Once the system is booted and the system clock is initialized, the system clock is completely independent of the real-time clock.

Using the timedatectl Command

This command is use for control date and time setting. Timedatectl may be used to query and change the system clock and its settings..

Displaying the Current Date and Time

~]$ **timedatectl**

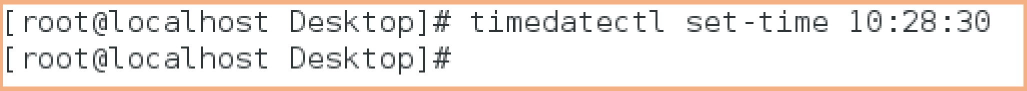


```
[root@localhost Desktop]# timedatectl
      Local time: Wed 2017-01-25 01:05:17 EST
  Universal time: Wed 2017-01-25 06:05:17 UTC
        RTC time: Tue 2017-01-24 22:05:17
        Timezone: America/New_York (EST, -0500)
     NTP enabled: yes
NTP synchronized: no
 RTC in local TZ: no
      DST active: no
 Last DST change: DST ended at
                  Sun 2016-11-06 01:59:59 EDT
                  Sun 2016-11-06 01:00:00 EST
 Next DST change: DST begins (the clock jumps one hour forward) at
                  Sun 2017-03-12 01:59:59 EST
                  Sun 2017-03-12 03:00:00 EDT
[root@localhost Desktop]#
```

To

change the current time, type the following at a shell prompt as root:

~]# timedatectl Øset-time ØHH:MM:SS

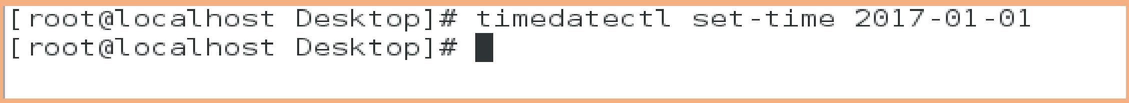


```
[root@localhost Desktop]# timedatectl set-time 10:28:30
[root@localhost Desktop]#
```

Changing the Current Date

To change the current date, type the following at a shell prompt as root:

~]# timedatectl Øset-time ØYYYY-MM-DD

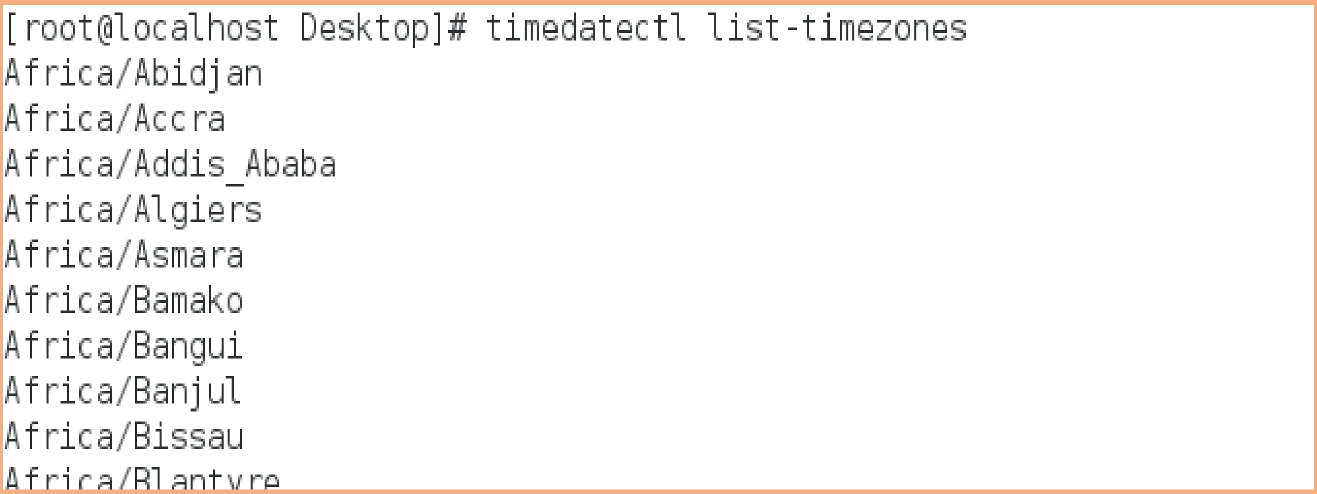


```
[root@localhost Desktop]# timedatectl set-time 2017-01-01
[root@localhost Desktop]# █
```

Changing the Time Zone

To list all available time zones, type the following at a shell prompt:

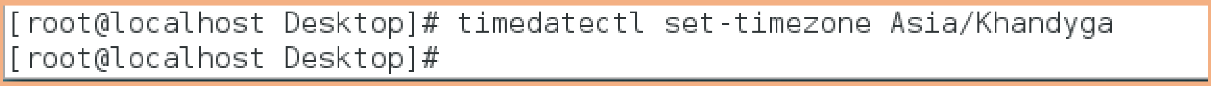
~]#timedatectl Ø list-timezones



```
[root@localhost Desktop]# timedatectl list-timezones
Africa/Abidjan
Africa/Accra
Africa/Addis_Ababa
Africa/Algiers
Africa/Asmara
Africa/Bamako
Africa/Bangui
Africa/Banjul
Africa/Bissau
Africa/Blantyre
```

To change the currently used time zone, type as root:

]#timedatectlØ set-time Øzone time_zone



```
[root@localhost Desktop]# timedatectl set-timezone Asia/Khandyga
[root@localhost Desktop]#
```

Synchronizing the System Clock with a Remote Server

The NTP service can be enabled and disabled using a command as follows:

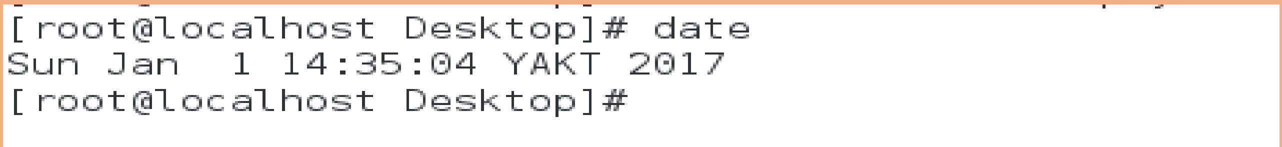
~]#timedatectlØ set-ntp Øyes   (enable=yes, disable=no)

Using the date Command

The date utility is available on all Linux systems and allows you to display and configure the current date and time.

To display the current date and time, run the date command with no additional command line options:

~]$date



```
[root@localhost Desktop]# date
Sun Jan  1 14:35:04 YAKT 2017
[root@localhost Desktop]#
```

To display the current date and time in UTC, type the following at a shell prompt:

```
~]$ date Ø –utc

Mon Sep 16 15:30:34 UTC 2017
```

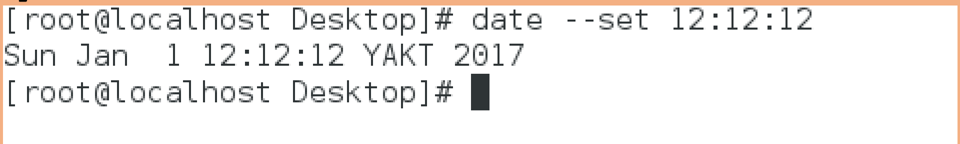
To customize the output of the date command, type:

```
~]$ date Ø +"%Y-%m-%d %H:%M"

2017-01-16 17:30
```

Changing the Current Time

To change the current time, run the date command with the --set or -s option as root:

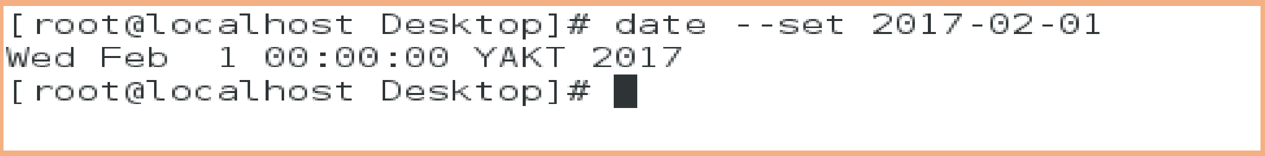
~]#date Ø --set Ø HH:MM:SS



```
[root@localhost Desktop]# date --set 12:12:12
Sun Jan  1 12:12:12 YAKT 2017
[root@localhost Desktop]#
```

To change the current date, run the date command with the --set or -s option as root:

~]#date Ø --set Ø YYYY-MM-DD



```
[root@localhost Desktop]# date --set 2017-02-01
Wed Feb  1 00:00:00 YAKT 2017
[root@localhost Desktop]# 
```

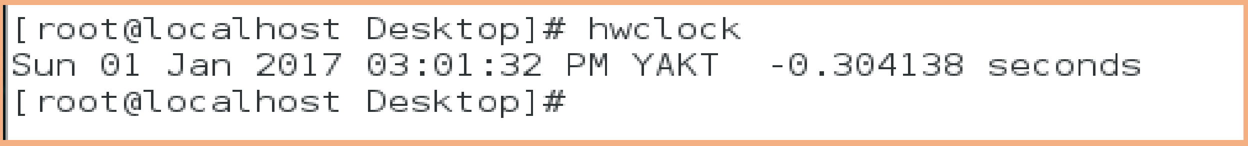
Using the hwclock Command

hwclock is a utility for accessing the hardware clock, also referred to as the Real Time Clock (RTC). This utility is used for displaying the time from the hardware clock. hwclock also contains facilities for compensating for systematic drift in the hardware clock.

Displaying the Current Date and Time

Running hwclock with no command line options as the root user returns the date and time in local time to standard output.

~]#hwclock

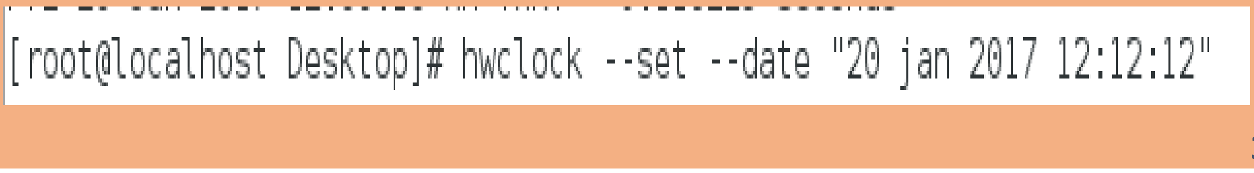


```
[root@localhost Desktop]# hwclock
Sun 01 Jan 2017 03:01:32 PM YAKT  -0.304138 seconds
[root@localhost Desktop]#
```

Setting the Date and Time

When you need to change the hardware clock date and time, you can do so by appending the --set and --date options along with your specification:

~]hwclock Ø --set Ø --date Ø "dd mmm yyyy HH:MM"



```
[root@localhost Desktop]# hwclock --set --date "20 jan 2017 12:12:12"
```

Basic Command For Linux

Note:-Ø=space

1. cat = It use to create file.
    #catØ>filename

2. mkdir= It use to create directory
    #mkdirØdirectoryname

3. mv = It use to rename file name
    #mvØoldfilenameØnewfilename

4. rm=This command is use to remove file
    #rmØfilename
    (–f option is use for forcefully)
5. vi=it is a file editor
    #viØfilename
6. vim= it is also a file editor improved version of vi
    #vimØfilename
7. cd=it use to access any directory
    #cdØdirectoryname or directorypath
8. ls=To list file and directory.
9. lsØ-l=To list file and directory with their permission.
10. ll= work same as a lsØ-l
11. lsØ-lh=To list file and directory with permission with file size.
12. Stat=This show statistics of file or folder
    #statØfilename or folder
13. cdØ..=To exit from any directory
14. cdØ~=To go home directory
15. rmdir= To remove empty directory
    #rmdirØdirectoryname
16. rmdirØ-P=To remove recursive directory
    #rmdirØ-PØdirectory path
17. useradd=is use to create user

#useraddØusername

18. usermodØ--login = this command is use to rename user name.
19. usermodØ-L=This command is use to lock user

    usermodØ-LØusername
20. pwd=This command is use for current location.
21. chmod=This command change access mode for files &directories
22. whoami=This command use see current user
23. chownØ-R=This command use for change ownership of directories
24. history=This command keep the record or execute command
25. top=This show running process
26. cp=This command is use for copy file or folder from one place to another

    #cpØsourceØdestination
27. man= This command is use to get any information about any command.

    #manØcommandname
28. Echo=display a line of text.

    #echoØwhitehat.
29. wcØprint the number of newlines, words, and bytes in files

    #wcØfilename
30. who – show who is logged on

    #who
31. ps – report a snapshot of the current processes

    #ps

    #psØ-aux
32. kill – to kill a process(using signal mechanism)

    #killØprocess id
33. du – estimate file space usage

    #duØfilename
34. df – report filesystem disk space usage

    #dfØ-ah(-a for all,-h for human readable)

35. reboot – reboot the system

```
#reboot
```

36. poweroff – power off the system

```
#poweroff
```

37. whoami-this command show the username

```
#whoami
```

38. whoØamØi-this command show the username with loggen on time

```
#whoØamØi
```

39. echoØ$SHELL- use to see user shell
40. find – search for files in a directory hierarchy

```
#findØfilename
```

41. history – prints recently used commands

```
#history
```

42. locate – find or locate a file

```
#locateØfilename
```

43. file – this command is use see type of in the file

```
#fileØfilename
```

Managing Users and Groups

The control of users and groups is a core element of Red Hat Enterprise Linux system administration. This chapter explains how to add, manage, and delete users and groups in the graphical user interface and on the command line, and covers advanced topics, such as creating group directories

Introduction to Users and Groups

While users can be either people (meaning accounts tied to physical users) or accounts which exist for specific applications to use, groups are logical expressions of organization, tying users together for a common purpose. Users within a group share the same permissions to read, write, or execute files owned by that group.

Each user is associated with a unique numerical identification number called a user ID (UID). Likewise, each group is associated with a group ID (GID). A user who creates a file is also the owner and group owner of that file. The file is assigned separate read, write, and execute permissions for the owner, the group, and everyone else. The file owner can be changed only by root, and access permissions can be changed by both the root user and file owner.

Red Hat Enterprise Linux uses a user private group (UPG) scheme, which makes UNIX groups easier to manage. A user private group is created whenever a new user is added to the system. It has the same name as the user for which it was created and that user is the only member of the user private group.

User private groups make it safe to set default permissions for a newly created file or directory, allowing both the user and the group of that user to make modifications to the file or directory.

The setting which determines what permissions are applied to a newly created file or directory is called a umask and is configured in the /etc/bashrc file. Traditionally on UNIX-based systems, the umask is set to 022, which allows only the user who created the file or directory to make modifications. Under this scheme, all other users, including members of the creator's group, are not allowed to make any modifications. However, under the UPG scheme, this “group protection” is not necessary since every user has their own private group.

A list of all groups is stored in the /etc/group configuration file

Shadow Passwords

In environments with multiple users, it is very important to use shadow passwords provided by the shadow-utils package to enhance the security of system authentication files. For this reason, the installation program enables shadow passwords by default.

The following is a list of the advantages shadow passwords have over the traditional way of storing passwords on UNIX-based systems:

Shadow passwords improve system security by moving encrypted password hashes from the world-readable /etc/passwd file to /etc/shadow, which is readable only by the root user.

Shadow passwords store information about password aging.

Shadow passwords allow the /etc/login.defs file to enforce security policies.

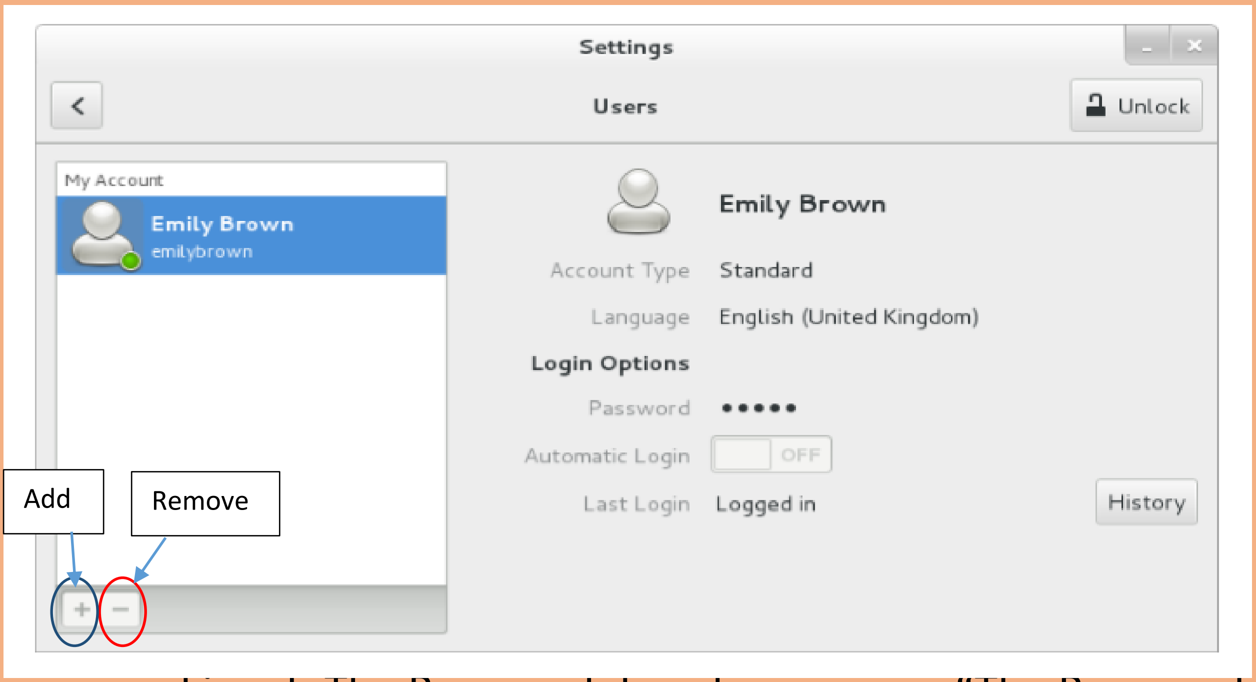
Most utilities provided by the shadow-utils package work properly whether or not shadow passwords are enabled. However, since password aging information is stored exclusively in the /etc/shadow file, some utilities and commands do not work without first enabling shadow passwords

Managing Users in a Graphical Environment

The Users utility allows you to view, modify, add, and delete local users in the graphical user interface.

Using the Users Settings Tool

Application → Setting → Users



Emily Brown
emilybrown

Emily Brown

Account Type Standard

Language English (United Kingdom)

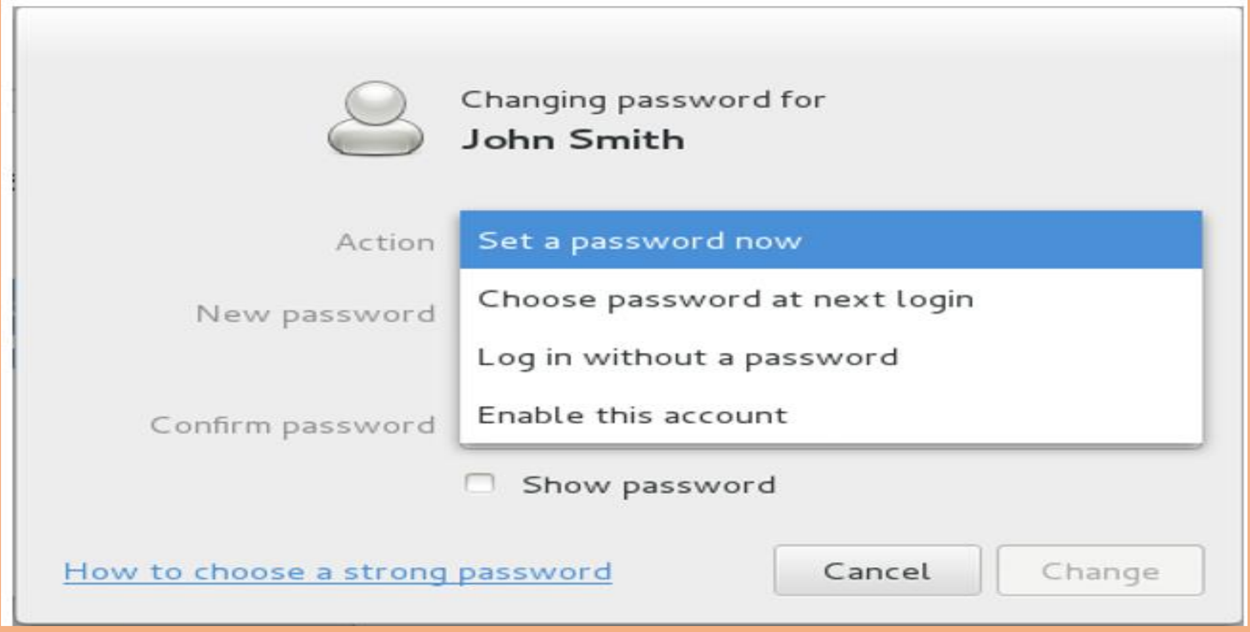
Login Options

Password •••••

Automatic Login OFF

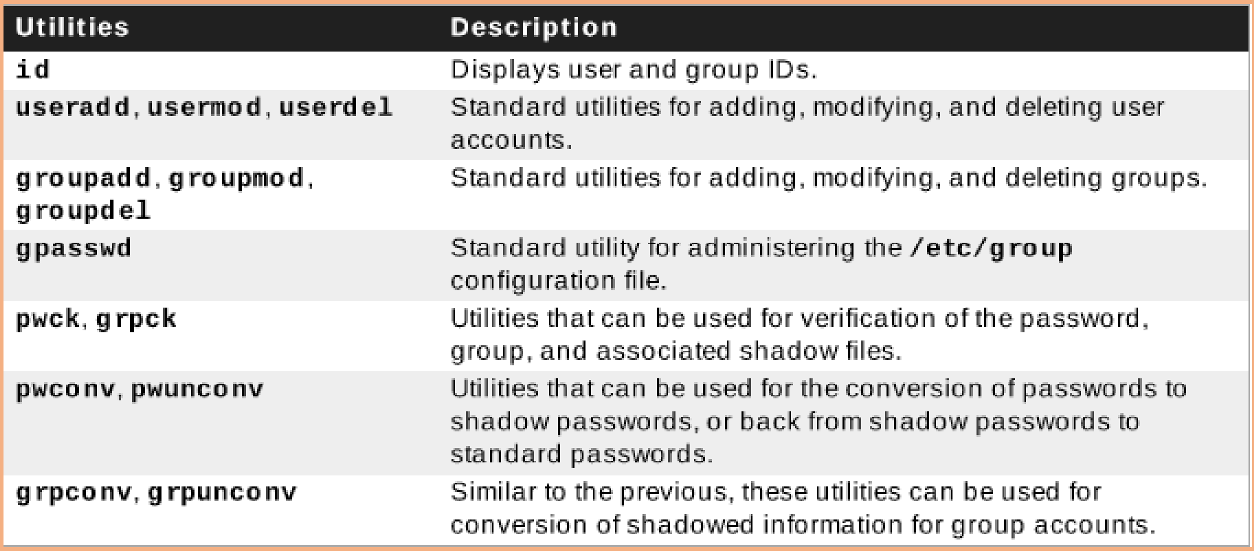
Last Login Logged in

password is set. The Password dropdown menu, “The Password Menu”, contains the options to set a password by the administrator immediately, choose a password by the user at the first login, or create a guest account with no password required to log in. You can also disable or enable an account from this menu.



Changing password for
John Smith

Using Command Line Tools



Utilities	Description
id	Displays user and group IDs.
useradd, usermod, userdel	Standard utilities for adding, modifying, and deleting user accounts.
groupadd, groupmod, groupdel	Standard utilities for adding, modifying, and deleting groups.
gpasswd	Standard utility for administering the **/etc/group** configuration file.
pwck, grpck	Utilities that can be used for verification of the password, group, and associated shadow files.
pwconv, pwunconv	Utilities that can be used for the conversion of passwords to shadow passwords, or back from shadow passwords to standard passwords.
grpconv, grpunconv	Similar to the previous, these utilities can be used for conversion of shadowed information for group accounts.

.

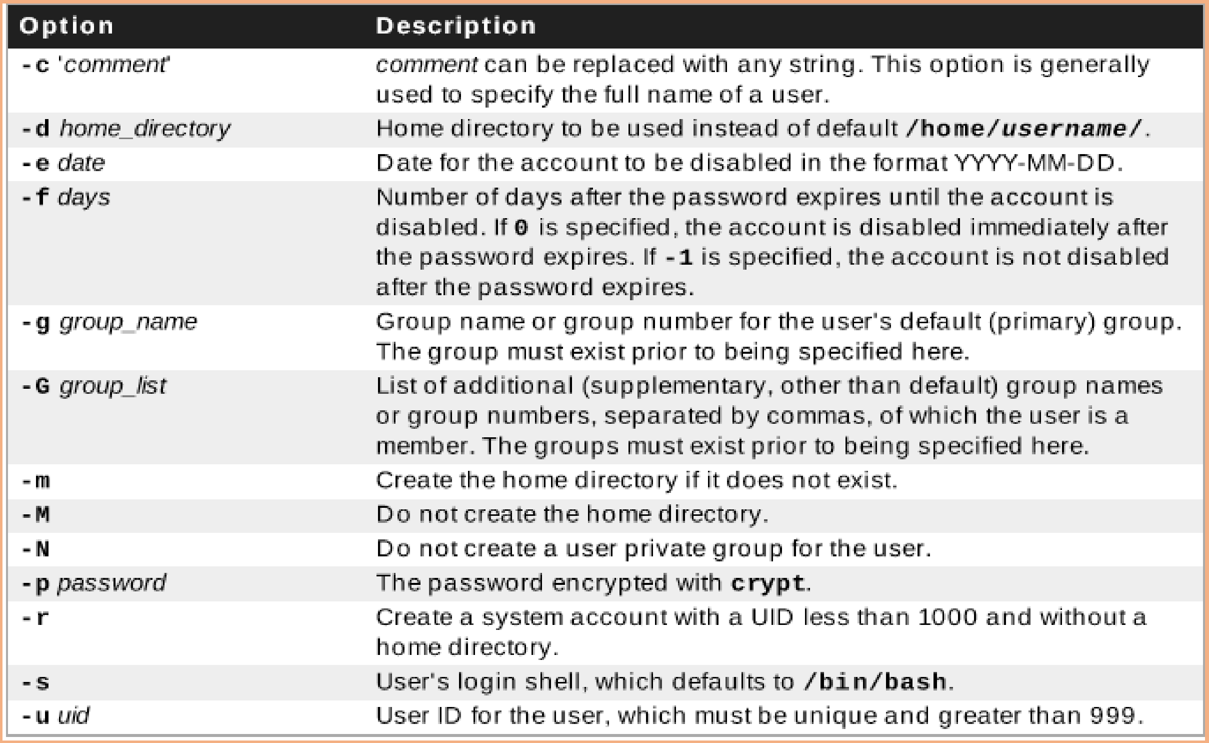
Adding a New User

To add a new user to the system, type the following at a shell prompt as root:

```
#useraddØ [options]Ø username
```

Set password

```
#passwdØusername
```



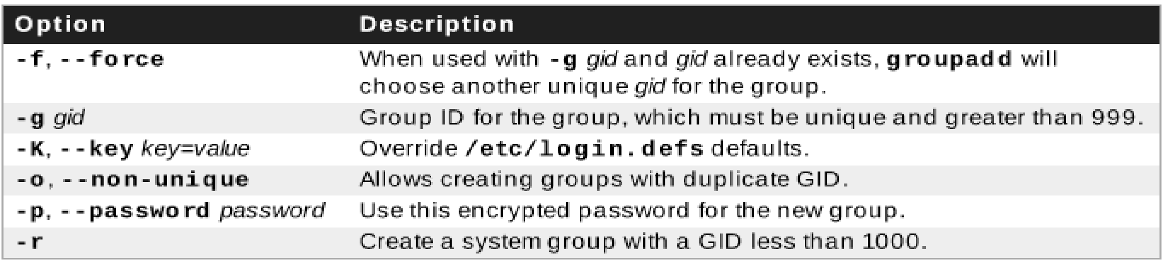
Option	Description
`-c` *'comment'*	*comment* can be replaced with any string. This option is generally used to specify the full name of a user.
`-d` *home_directory*	Home directory to be used instead of default **`/home/username/`**.
`-e` *date*	Date for the account to be disabled in the format YYYY-MM-DD.
`-f` *days*	Number of days after the password expires until the account is disabled. If **`0`** is specified, the account is disabled immediately after the password expires. If **`-1`** is specified, the account is not disabled after the password expires.
`-g` *group_name*	Group name or group number for the user's default (primary) group. The group must exist prior to being specified here.
`-G` *group_list*	List of additional (supplementary, other than default) group names or group numbers, separated by commas, of which the user is a member. The groups must exist prior to being specified here.
`-m`	Create the home directory if it does not exist.
`-M`	Do not create the home directory.
`-N`	Do not create a user private group for the user.
`-p` *password*	The password encrypted with **`crypt`**.
`-r`	Create a system account with a UID less than 1000 and without a home directory.
`-s`	User's login shell, which defaults to **`/bin/bash`**.
`-u` *uid*	User ID for the user, which must be unique and greater than 999.

add a user to another supplementary group, you need to use the -a, --append option with the -G option. Otherwise the list of supplementary groups for the user will be overwritten by those specified with the usermod -G command.

Adding a New Group

To add a new group to the system, type the following at a shell prompt as root:

```
#groupaddØ [options] Øgroup_name
```



Option	Description
`-f`, **`--force`**	When used with **`-g`** *gid* and *gid* already exists, **`groupadd`** will choose another unique *gid* for the group.
`-g` *gid*	Group ID for the group, which must be unique and greater than 999.
`-K`, **`--key`** *key=value*	Override **`/etc/login.defs`** defaults.
`-o`, **`--non-unique`**	Allows creating groups with duplicate GID.
`-p`, **`--password`** *password*	Use this encrypted password for the new group.
`-r`	Create a system group with a GID less than 1000.

Creating Group Directories

System administrators usually like to create a group for each major project and assign people to the group when they need to access that project's files. With this traditional scheme, file management is difficult; when someone creates a file, it is associated with the primary group to which they belong. When a single person works on multiple projects, it becomes difficult to associate the right files with the right group. However, with the UPG scheme, groups are automatically assigned to files created within a directory with the setgid bit set. The setgid bit makes managing group projects that share a common directory very simple because any files a user creates within the directory are owned by the group that owns the directory.

cont.........

For example, a group of people need to work on files in the /opt/myproject/ directory. Some people are trusted to modify the contents of this directory, but not everyone.

1. As root, create the /opt/myproject/ directory by typing the following at a shell prompt:

```
#mkdir Ø/opt/myproject
```

2. Add the myproject group to the system:

```
#groupaddØmyproject
```

3. Associate the contents of the /opt/myproject/ directory with the myproject group:

```
#chown Øroot:myproject Ø/opt/myproject
```

4. Allow users in the group to create files within the directory and set the setgid bit:

```
#chmod Ø2775 Ø/opt/myproject
```

At this point, all members of the myproject group can create and edit files in the /opt/myproject/ directory without the administrator having to change file permissions every time users write new files. To verify that the permissions have been set correctly, run the following command:

```
# ls -ld Ø/opt/myproject
  drwxrwsr-x. 3 root myproject 4096 Mar  3 18:31 /opt/myproject
```

5. Add users to the myproject group:

```
usermod Ø-aGØmyprojectØusername
```

Additional Resources

For more information on how to manage users and groups on Red Hat Enterprise Linux, see the resources listed below.

Installed Documentation

For information about various utilities for managing users and groups, see the following manual pages:

useradd(8) — The manual page for the useradd command documents how to use it to create new users.

userdel(8) — The manual page for the userdel command documents how to use it to delete users.

usermod(8) — The manual page for the usermod command documents how to use it to modify users.

groupadd(8) — The manual page for the groupadd command documents how to use it to create new groups.

groupdel(8) — The manual page for the groupdel command documents how to use it to delete groups.

groupmod(8) — The manual page for the groupmod command documents how to use it to modify group membership.

gpasswd(1) — The manual page for the gpasswd command documents how to manage the /etc/group file.

grpck(8) — The manual page for the grpck command documents how to use it to verify the integrity of the /etc/group file.

pwck(8) — The manual page for the pwck command documents how to use it to verify the integrity of the /etc/passwd and /etc/shadow files.

pwconv(8) — The manual page for the pwconv, pwunconv, grpconv, and grpunconv commands documents how to convert shadowed information for passwords and groups.

id(1) — The manual page for the id command documents how to display user and group IDs.

For information about related configuration files, see:

group(5) — The manual page for the /etc/group file documents how to use this file to define system groups.

passwd(5) — The manual page for the /etc/passwd file documents how to use this file to define user information.

shadow(5) — The manual page for the /etc/shadow file documents how to use this file to set passwords and account expiration information for the system.

FILE AND FOLDER PERMISSION

Basic File Permissions.

In Red Hat Enterprise Linux, all files have file permissions that determine whether a user is allowed to read, write, or execute them.

There are two method of set permission on file and folder

1. Symbolic=r(read)w(write)x(execute)
2. Numerical=4(read)2(write)1(execute)

Note:(lsØ-lØfile and foldername is use see file and folder permission).

There are 3 column of permission

1=for owner permission
2=for group permission
3=for other permission

chmod- This command use to change the file permission

ex=chmodØo+rwxØfilename

(o=other,u=owner,g=group)

ex=chmodØ777Øfile and folder

↓ ↓ ↓

(1 2 3)

1=owner permission
2=group permission
3=other permission

Basic File Access Permissions

Each file and directory has three user based permission groups:

1) owner (Users)- The Owner permissions apply only the owner of the file or directory, they will not impact the actions of other users.

2) group - The Group permissions apply only to the group that has been assigned to the file or directory, they will not effect the actions of other users.

3) all users (Others)- The All Users permissions apply to all other users on the system, this is the permission group that you want to watch the most.

Permission Types

Each file or directory has three basic permission types:

1) read - The Read permission refers to a user's capability to read the contents of the file.

2) write - The Write permissions refer to a user's capability to write or modify a file or directory.

3) execute - The Execute permission affects a user's capability to execute a file or view the contents of a directory.

When you issue the command ls -l, the first column of information contains these file permissions. Within this first column are places for 9 letters or hyphens.

Cont………

Example:-

```
[root@server1 ~]#ll
drwxr-xr-x.     3     root root 4096       may 25    2011
        Documents
↓               ↓        ↓       ↓       ↓      ↓       ↓        ↓          ↓

 1              2        3       4       5      6       7        8          9
```

1 A. The first space is either a hyphen, the letter d, or the letter l.

a) A **hyphen (_)** means it is a file.

b) A letter **d** means it is a directory.

c) A letter **l** means it is a symbolic link to a directory somewhere else on the file system.

1 B.The next nine spaces are divided into three sets of permissions are as follows :-

a) rwx – Read, Write and Execute permission for the owner of the file or directory.

b) r-x – Read and Execute permissions for the group owing file or directory.

c) r-w – Read and Execute permissions for all other users for file or directory.

2. 3 – its link

3. root = Owner name of the file or Directory.

4. root = Group name of the file or Directory.

5. 4096 = File or Directory size.

6. may = Month

7. 25 = Date

8. 2011 = Year

9. Documets = File or Directory name

Methods of Implementing Permission

1. Symbolic Mode :- in Symbolic Mode file or directory permissions are denotes as follows :-

Read Permission = r

Write Permission = w

Execute Permission = x

Example :

Digits	Permission
x	execute
w	write
r	read
wx	write + execute
rx	read + execute
rw	read + write
rwx	read + write + execute

2. Absolute Mode or Octal Value :- in Absolute Mode file or directory permissions are denotes as follows :-

Read Permission = 4

Write Permission = 2

Execute Permission = 1

Example :

Digits	Permission
0	none
1	execute
2	write
4	read
3 (2+1)	write + execute
5 (4+1)	read + execute
6 (4+2)	read + write
7 (4+2+1)	read + write + execute

Default File Permission :- When the file is get created with the help of cat, vi, or touch command it will get the permission for the as **–rw-r—r--** or **644**

Example :-

```
[root@server1 ~]#touch Øfile1 Øfile2
[root@server1 ~]#ll
-rw-r--r--. 3     root root 0     may 25    2011       file1
-rw-r--r--. 3     root root 0     may 25    2011       file2
 ↓  ↓ ↓
 1  2  3
```

1. rw- = read-write permission for the owner of the file.

2. r-- = read permission for the owner’s gorup of the file.

3. r-- = read permissionfor the others.

Default Directory Permission :- When the directory is get created with the mkdir command it will get the permission for the as **drwxr-xr-x** or **755**

Example :-

```
[root@server1 ~]#mkdir dir1 dir2
[root@server1 ~]#ll
drwxr-xr-x.     3     root root 0     may 25    2011       dir1
drwxr-xr-x.     3     root root 0     may 25    2011       dir2
 ↓  ↓ ↓
 1  2 3
```

1. rwx = read-write-execute permission for the owner of the directory.

2. r-x = read-execute permission for the owner’s gorup of the directory.

3. r-x = read-execute permission for the others.

Some examples of this permissions.

Permissions :-

Read (r = 4) Write (w = 2) Others (x = 1)

Owner	**Group**	**Other**
1. rwx	rwx	rwx
7(4+2+1)	7(4+2+1)	7(4+2+1)
2. rwx	rwx	rw
7(4+2+1)	7(4+2+1)	6(4+2)
3. rwx	rwx	rx
7(4+2+1)	7(4+2+1)	5(4+1)
4. rwx	rwx	r
7(4+2+1)	7(4+2+1)	4
5. rwx	rwx	wx
7(4+2+1)	7(4+2+1)	3
6. rwx	rwx	w
7(4+2+1)	7(4+2+1)	2
7. rwx	rwx	x
7(4+2+1)	7(4+2+1)	1
8. rwx	rw	rwx
7(4+2+1)	6(4+2)	7(4+2+1)
9. rwx	rw	rw
7(4+2+1)	6(4+2)	6(4+2)
10. rwx	rw	rx
7(4+2+1)	6(4+2)	5(4+1)

Umask :-The user file-creation mode mask (umask) is use to determine the file permission for newly created files. It can be used to control the default file permission for new files. Only the root user can set UMASK. It is a four-digit octal number. A umask can be set or expressed using:

1. Symbolic values = u=rwx,g=rx,o=rx

2. Octal values = 0022

```
[root@server1 ~]#umask
0022
[root@server1 ~]#umask Ø-S
u=rwx,g=rx,o=rx

[root@server1 ~]#umask Ø –S u=rwx,g=r,o=r
u=rwx,g=r,o=r
[root@server1 ~]#umask
0033
[root@server1 ~]#touch Ø 1
[root@server1 ~]#ll
-rw-r—r--.  1      root   root  0      jun   12    21:28   1

[root@server1 ~]#umask Ø –S Ø u=rwx,g=w,o=w
u=rwx,g=w,o=w
[root@server1 ~]#umask
0055
[root@server1 ~]#touch Ø 2
[root@server1 ~]#ll
-rw--w--w-.  1     root   root  0      jun   12    21:28   2
[root@server1 ~]#umask Ø –S Ø u=rwx,g=x,o=x
u=rwx,g=x,o=x
[root@server1 ~]#umask
0066
```

```
[root@server1 ~]#touch Ø 3
[root@server1 ~]#ll
-rw-------.  1       root   root  0      jun   12    21:28   3
[root@server1 ~]#umask Ø –S Ø u=rwx,g=rw,o=rw
u=rwx,g=rw,o=rw
[root@server1 ~]#umask
0011
[root@server1 ~]#touch Ø 4
[root@server1 ~]#ll
-rw-rw-rw-.  1     root   root  0      jun   12    21:28   4
```

File and Directory Permissions Tutorial.

1. To view file or directory permissions.

```
 [root@server1 ~]#ll
-rw-r--r--.   3      root  root  0      may  25     2011 file1
drwxr-xr-x.3       root  root  0      may  25     2011 dir1
```

2. To change file permission of Users.

```
[root@server1 ~]#useradd  Ø u1
[root@server1~]# passwd Ø u1
Changing password for user username.
New password: * * * * *
BAD PASSWORD: it is based on a directory word
BAD PASSWORD: is too simple
Retype new password:
passwd: all authentication token updated successfully.
[root@server1 ~]#useradd  Ø u2
[root@server1~]# passwd  Ø u2
Changing password for user username.
New password: * * * * *
BAD PASSWORD: it is based on a directory word
BAD PASSWORD: is too simple
Retype new password:
passwd: all authentication token updated successfully.
```

```
[root@server1 ~]#useradd  Ø u3
[root@server1~]# passwd  Ø u3
Changing password for user username.
New password: * * * * *
BAD PASSWORD: it is based on a directory word
BAD PASSWORD: is too simple
Retype new password:
passwd: all authentication token updated successfully.

[root@server1 ~]#cat  Ø > Ø f1
Welcome to Aegis
CTRL+D
[root@server1 ~]#ll
-rw-r--r--.  3     root  root  22    may  25    2011 f1
[root@server1 ~]#chmod   Ø u+rwx   Ø f1
[root@server1 ~]#ll
-rwxr--r--.  3     root  root  22    may  25    2011 f1
```

3. To change file Ownership of Users.

```
[root@server1 ~]#ll
-rwxr--r--.  3     root  root  22    may  25    2011 f1
[root@server1 ~]#chown Ø  f1 Ø /home/u1

Now login with that user from another console.
<PRESS> CTRL+ALT+F2
Server1 Login : u1
Password: ******
[u1@server1 ~]$ll
-rwxr--r--.  3     root  root  22    may  25    2011 f1
[u1@server1 ~]$cat  Ø f1
Welcome to Aegis
[u1@server1 ~]$cat Ø  >> Ø  f1
-bash:  f1:  Permission  denied
<PRESS>  CTLR+D
```

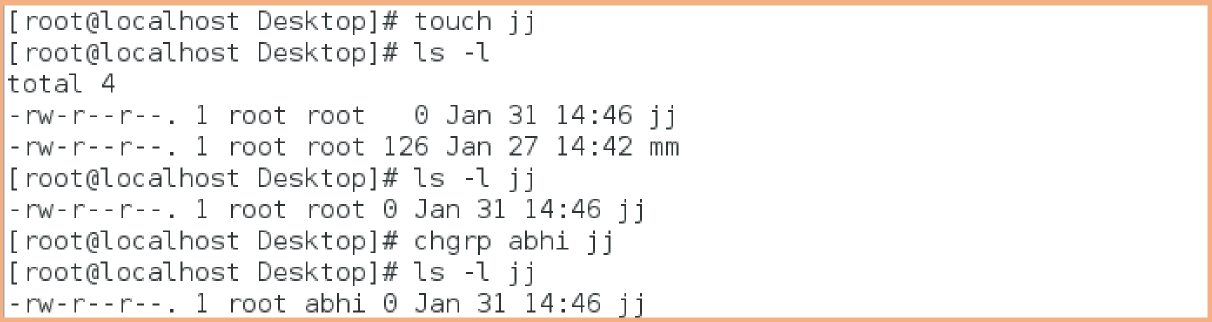
Now login with root user so that he can change users ownership.

```
Server1 Login : root
Password: ******
[root@server1 ~]#ll
-rwxr--r--.  3      root  root  22    may  25    2011 f1
[root@server1 ~]#chown   u1  f1
                         |   |
          Username  <----    ---> Filename
[root@server1 ~]#ll
-rwxr--r--.  3      u1    root  22    may  25    2011 f1
```

4. To change file Permission of Group.

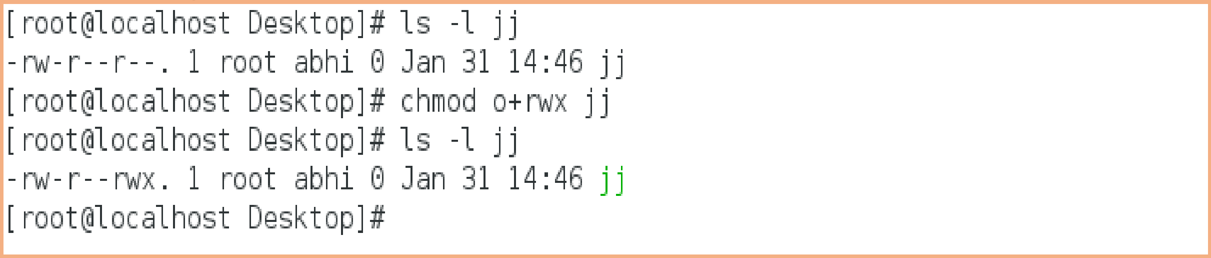
```
[root@server1 ~]#cat Ø > Ø f2
Welcome to Aegis
CTRL+D
[root@server1 ~]#ll
-rw-r--r--.   3      root  root  22    may  25    2011 f2
[root@server1 ~]#chmod   Ø g+rw   Ø f1
[root@server1 ~]#ll
-rw-rw-r--. 3      root  root  22    may  25    2011 f2
```

5. To change file Ownership of Groups.



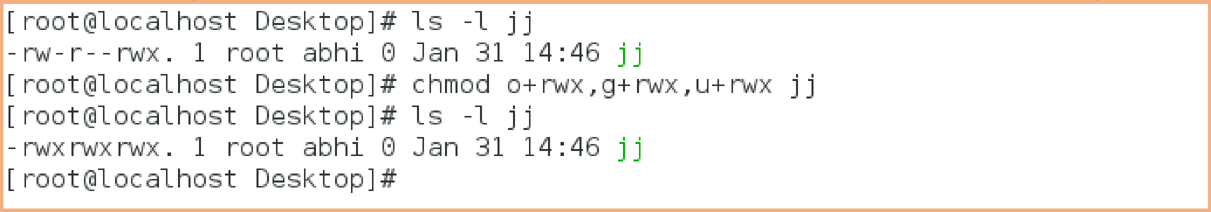
```
[root@localhost Desktop]# touch jj
[root@localhost Desktop]# ls -l
total 4
-rw-r--r--. 1 root root   0 Jan 31 14:46 jj
-rw-r--r--. 1 root root 126 Jan 27 14:42 mm
[root@localhost Desktop]# ls -l jj
-rw-r--r--. 1 root root 0 Jan 31 14:46 jj
[root@localhost Desktop]# chgrp abhi jj
[root@localhost Desktop]# ls -l jj
-rw-r--r--. 1 root abhi 0 Jan 31 14:46 jj
```

6. To change file permission of Others.



```
[root@localhost Desktop]# ls -l jj
-rw-r--r--. 1 root abhi 0 Jan 31 14:46 jj
[root@localhost Desktop]# chmod o+rwx jj
[root@localhost Desktop]# ls -l jj
-rw-r--rwx. 1 root abhi 0 Jan 31 14:46 jj
[root@localhost Desktop]#
```

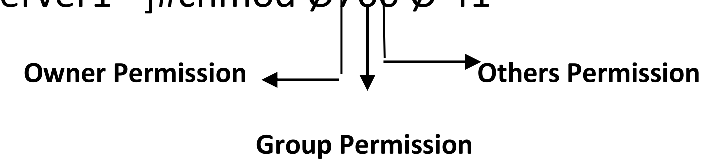
7. To change the file permission of User, Group and Others together.



```
[root@localhost Desktop]# ls -l jj
-rw-r--rwx. 1 root abhi 0 Jan 31 14:46 jj
[root@localhost Desktop]# chmod o+rwx,g+rwx,u+rwx jj
[root@localhost Desktop]# ls -l jj
-rwxrwxrwx. 1 root abhi 0 Jan 31 14:46 jj
[root@localhost Desktop]#
```

8. To change file permission using absolute mode.

```
[root@server1 ~]#cat Ø  > Ø f6
Welcome to Aegis
CTRL+D
[root@server1 ~]#ll
-rw-r--r--.   3      root  root  22     may  25     2011 f6
[root@server1 ~]#chmod Ø760 Ø  f1
```



Owner Permission

Others Permission

Group Permission

```
[root@server1 ~]#ll
-rwxrw----. 3      root  root  22     may  25     2011 f6
```

9. To change directory permission of Others.

```
[root@server1 ~]#mkdir Ø  dir1
[root@server1 ~]#ll
drwxr-xr-x.3      root  root  22     may  25     2011 dir1
[root@server1 ~]#chmod Ø   g+w Ø   f1
[root@server1 ~]#ll
drwxrwxr-x.       3      root  root  22     may  25     2011 dir1
```

10. To change the directory permission of User, Group and Others together.

```
[root@server1 ~]#mkdir Ø  dir2
drwxr-xr-x.3      root  root  22     may  25     2011 dir2
[root@server1 ~]#chmod   u+rwx,g+rw,o+x   f1
[root@server1 ~]#ll
drwxrw---x.       3      root  root  22     may  25     2011 dir2
```

11. To change directory permission using absolute mode.

```
[root@server1 ~]#mkdir Ø dir3
[root@server1 ~]#ll
drwxr-xr-x.3      root  root  22    may  25    2011 dir3
[root@server1 ~]#chmod Ø760 Ø f1

   Users Permission ←——  ↓  ——→Others Permission
                  Group Permission

[root@server1 ~]#ll
drwxrw----.       3      root  root  22    may  25    2011 dir3
```

Chown=This Command use to change the owner of file and folder.

Example chownØownernameØfilename and folder name

```
~]#chownØITØking.txt
```

chgrp=This command use to change the group of file and folder.

Example:chgrpØgroupnameØfile and foldername.

```
~]#chgrpØAegisØking.txt
```

Installing and Managing Software

There are two manager to manage software

1. RPM (Red hat Package Manager)

2. YUM (Yellow Dog Update Modifier)

RPM

rpm is a powerful Package Manager, which can be used to build, install, query, verify, update, and erase individual software packages. A package consists of an archive of files and meta-data used to install and erase the archive files. The meta-data includes helper scripts, file attributes, and descriptive information about the package. Packages come in two varieties: binary packages, used to encapsulate software to be installed, and source packages, containing the source code and recipe necessary to produce binary packages.

One of the following basic modes must be selected: Query, Verify,Install/Upgrade/Freshen, Uninstall, Set Owners/Groups, Show Querytags, and Show Configuration.

YUM

All software on a Red Hat Enterprise Linux system is divided into RPM packages, which can be installed, upgraded, or removed. This part describes how to manage packages on Red Hat Enterprise Linux using Yum.

YUM

(Yellow Dog update modifier)

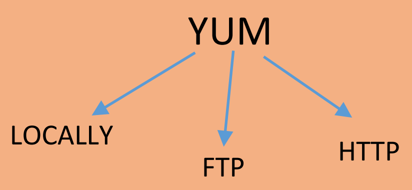
Yum is the Red Hat package manager that is able to query for information about available packages, fetch packages from repositories, install and uninstall them, and update an entire system to the latest available version. Yum performs automatic dependency resolution when updating, installing, or removing packages, and thus is able to automatically determine, fetch, and install all available dependent packages.

Configuring Yum and Yum Repositories:

The configuration information for yum and related utilities is located at /etc/yum.conf. This filecontains one mandatory [main] section, which enables you to set yum options that have global effect, and can also contain one or more [*repository*] sections, which allow you to set repositoryspecific options. However, it is recommended to define individual repositories in new or existing.repo files in the /etc/yum.repos.d/ directory. The values you define in individual [*repository*] sections of the /etc/yum.conf file override values set in the [main] section.

This section shows you how to:

- Set global yum options by editing the [main] section of the /etc/yum.conf configuration file;
- Set options for individual repositories by editing the [*repository*] sections in /etc/yum.conf and .repo files in the /etc/yum.repos.d/ directory;
- Use yum variables in /etc/yum.conf and files in the /etc/yum.repos.d/ directory so that dynamic version and architecture values are handled correctly;
- Add, enable, and disable yum repositories on the command line; and
- Set up your own custom yum repository.



How to configure YUM

1. First create directory.

   #mkdirØ/directoryname

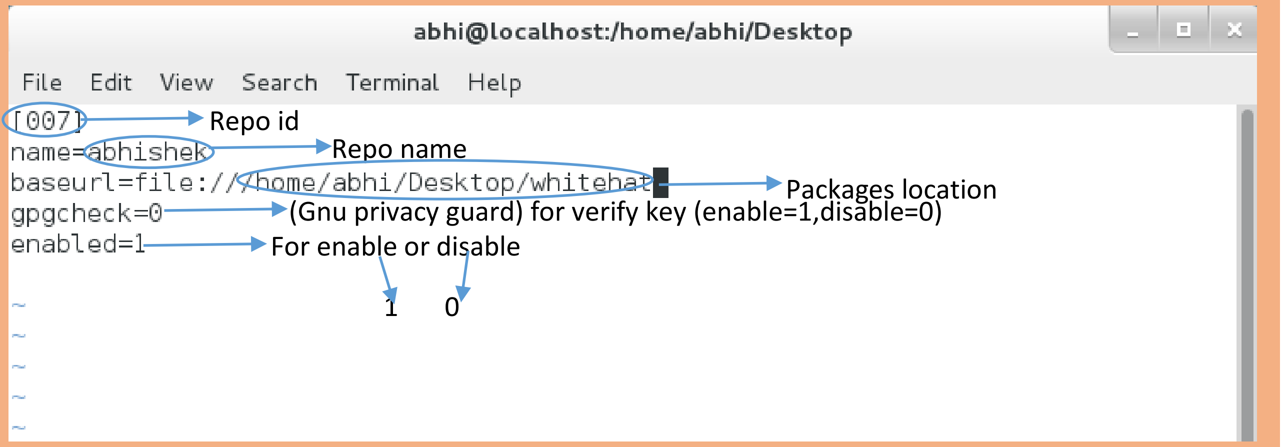
2. Copy all the packages from your media to that directory what you created.

   #cpØ-ivrØ/run/media/username/Rhel-7.server/*Ø/directoryname

3. Now create repository file.

   #viØ/etc/yum.repos.d/filename.repo

   And entre below input on this file


abhi@localhost:/home/abhi/Desktop
File Edit View Search Terminal Help

```
[007]
name=abhishek
baseurl=file:///home/abhi/Desktop/whitehat
gpgcheck=0
enabled=1
~
~
~
~
~
```

Repo id

Repo name

Packages location

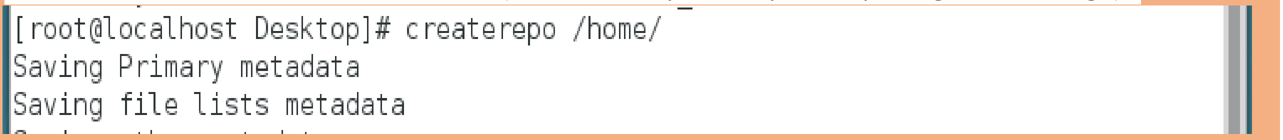
(Gnu privacy guard) for verify key (enable=1,disable=0)

For enable or disable

1 0

4. Now create index

   #createrepoØ/directoryname (That directory where your all packages are storage)

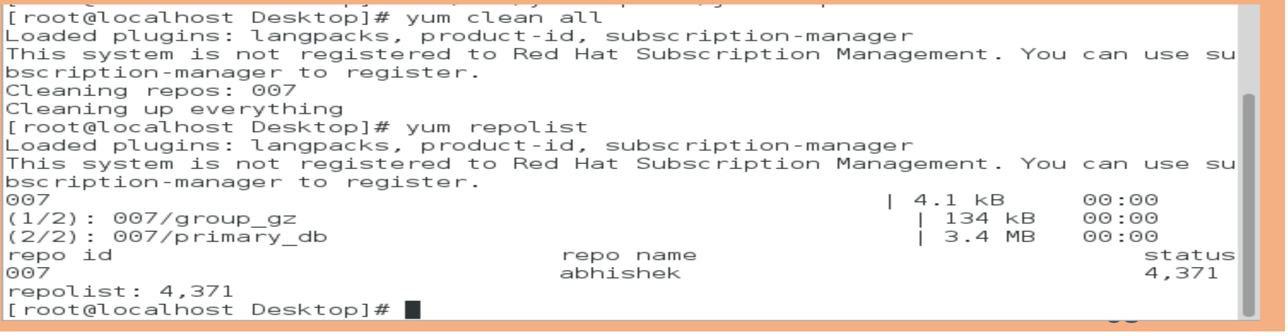


```
[root@localhost Desktop]# createrepo /home/
Saving Primary metadata
Saving file lists metadata
```

5. Now check your repository

   #yumØcleanØall

   #yumØrepolist



```
[root@localhost Desktop]# yum clean all
Loaded plugins: langpacks, product-id, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use su
bscription-manager to register.
Cleaning repos: 007
Cleaning up everything
[root@localhost Desktop]# yum repolist
Loaded plugins: langpacks, product-id, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use su
bscription-manager to register.
007                                                          | 4.1 kB     00:00
(1/2): 007/group_gz                                            | 134 kB     00:00
(2/2): 007/primary_db                                          | 3.4 MB     00:00
repo id                              repo name                              status
007                                  abhishek                               4,371
repolist: 4,371
[root@localhost Desktop]# 
```

Note:- 1. If you want to search any package so use

```
#yumØsearchØpackagename
```

2. If you want to remove package so use

```
#yumØremoveØpackagename
```

3. yumØhelp-> will display installed and available packages

```
#yumØhelp
```

4. obtains and installs a software package, including any dependencies

```
#yumØinstallØpackagename
```

5. for update the newer version of the software package, including any dependencies.

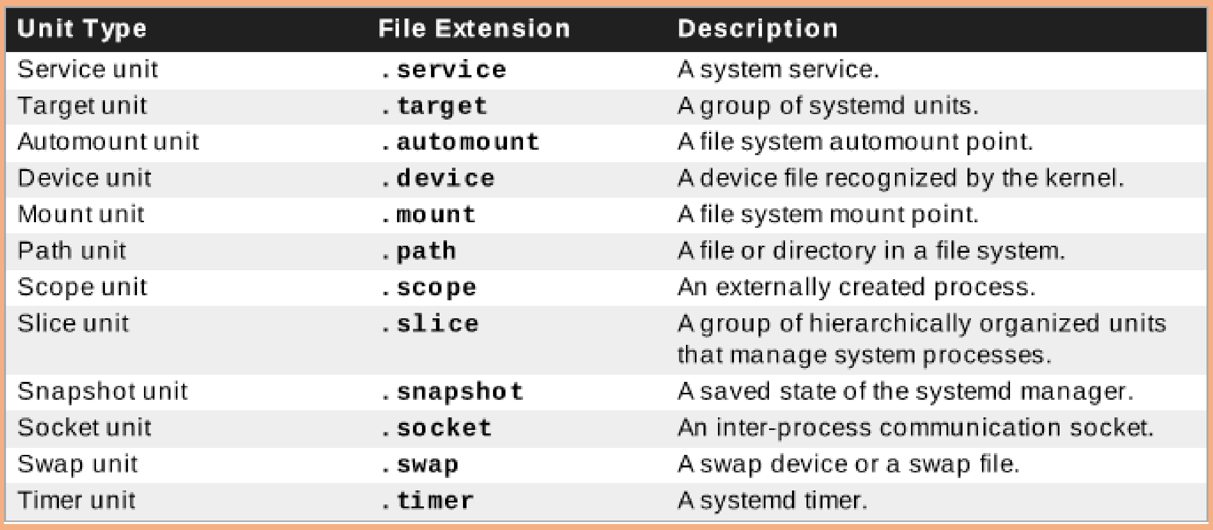
```
#yumØupdateØpackagename
```

Managing Services with system

Introduction to systemd

Systemd is a system and service manager for Linux operating systems. In Red Hat Enterprise Linux 7, systemd replaces Upstart as the default init system. It provides a number of features such as parallel startup of system services at boot time, on-demand activation of daemons, support for system state snapshots, or dependency-based service control logic.

Available systemd Unit Types:



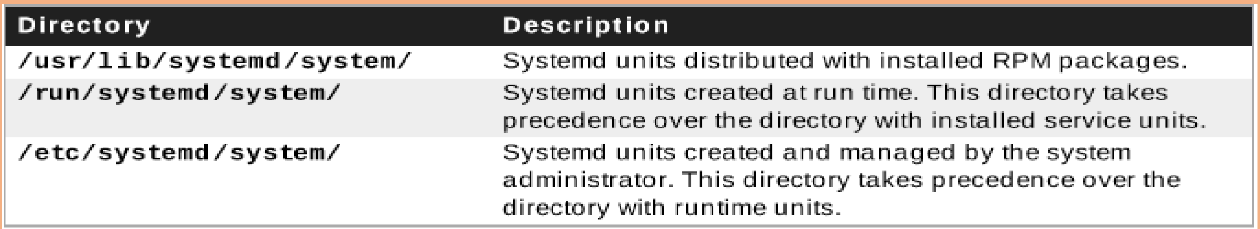
Unit Type	File Extension	Description
Service unit	`.service`	A system service.
Target unit	`.target`	A group of systemd units.
Automount unit	`.automount`	A file system automount point.
Device unit	`.device`	A device file recognized by the kernel.
Mount unit	`.mount`	A file system mount point.
Path unit	`.path`	A file or directory in a file system.
Scope unit	`.scope`	An externally created process.
Slice unit	`.slice`	A group of hierarchically organized units that manage system processes.
Snapshot unit	`.snapshot`	A saved state of the systemd manager.
Socket unit	`.socket`	An inter-process communication socket.
Swap unit	`.swap`	A swap device or a swap file.
Timer unit	`.timer`	A systemd timer.

Systemd Unit Locations

Main Features

In Red Hat Enterprise Linux 7, the systemd system and service manager provides the following main features:

Socket-based activation — At boot time, systemd creates listening sockets for all system services that support this type of activation, and passes the sockets to these services as soon as they are started. This not only allows systemd to start services in parallel, but also makes it possible to restart a service without losing any message sent to it while it is unavailable: the corresponding socket remains accessible and all messages are queued. Systemd uses socket units for socket-based activation.



Directory	Description
/usr/lib/systemd/system/	Systemd units distributed with installed RPM packages.
/run/systemd/system/	Systemd units created at run time. This directory takes precedence over the directory with installed service units.
/etc/systemd/system/	Systemd units created and managed by the system administrator. This directory takes precedence over the directory with runtime units.

Bus-based activation — System services that use D-Bus for inter-process communication can be started on-demand the first time a client application attempts to communicate with them. Systemd uses D-Bus service files for bus-based activation.

Device-based activation — System services that support device-based activation can be started ondemand when a particular type of hardware is plugged in or becomes available. Systemd uses device units for device-based activation.

Path-based activation — System services that support path-based activation can be started ondemand when a particular file or directory changes its state. Systemd uses path units for pathbased activation.

System state snapshots — Systemd can temporarily save the current state of all units or restore a previous state of the system from a dynamically created snapshot. To store the current state of the system, systemd uses dynamically created snapshot units.

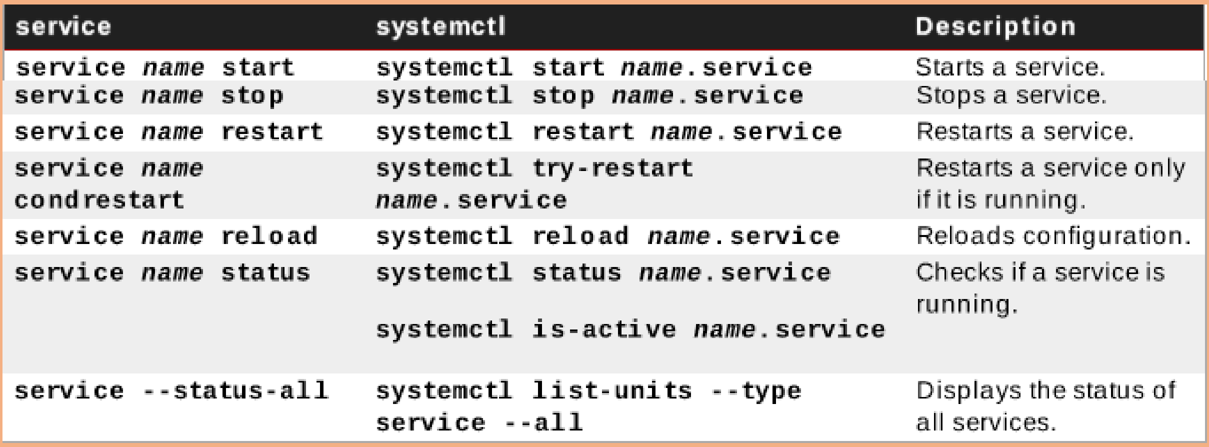
Mount and automount point management — Systemd monitors and manages mount and automount points. Systemd uses mount units for mount points and automount units for automount points.

Aggressive parallelization — Because of the use of socket-based activation, systemd can start system services in parallel as soon as all listening sockets are in place. In combination with system services that support on-demand activation, parallel activation significantly reduces the time required to boot the system.

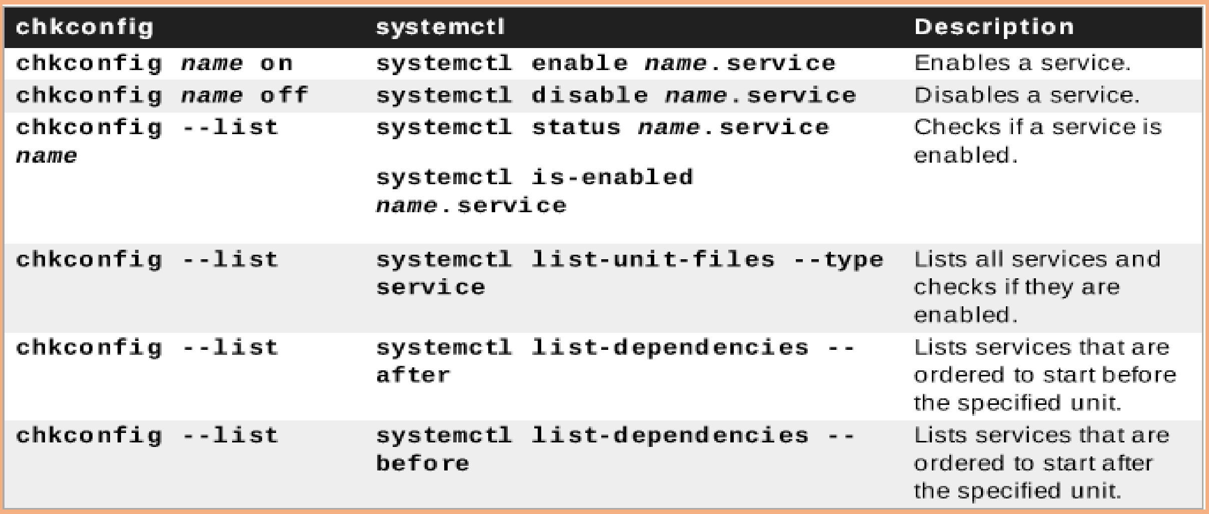
Transactional unit activation logic — Before activating or deactivating a unit, systemd calculates its dependencies, creates a temporary transaction, and verifies that this transaction is consistent. If a transaction is inconsistent, systemd automatically attempts to correct it and remove non-essential jobs from it before reporting an error.

Backwards compatibility with SysV init — Systemd fully supports SysV init scripts as described in the Linux Standard Base Core Specification, which eases the upgrade path to systemd service units.

Comparison of the service Utility with systemctl



service	systemctl	Description
service *name* start	systemctl start *name*.service	Starts a service.
service *name* stop	systemctl stop *name*.service	Stops a service.
service *name* restart	systemctl restart *name*.service	Restarts a service.
service *name* condrestart	systemctl try-restart *name*.service	Restarts a service only if it is running.
service *name* reload	systemctl reload *name*.service	Reloads configuration.
service *name* status	systemctl status *name*.service systemctl is-active *name*.service	Checks if a service is running.
service --status-all	systemctl list-units --type service --all	Displays the status of all services.



chkconfig	systemctl	Description
`chkconfig` *name* `on`	`systemctl enable` *name*`.service`	Enables a service.
`chkconfig` *name* `off`	`systemctl disable` *name*`.service`	Disables a service.

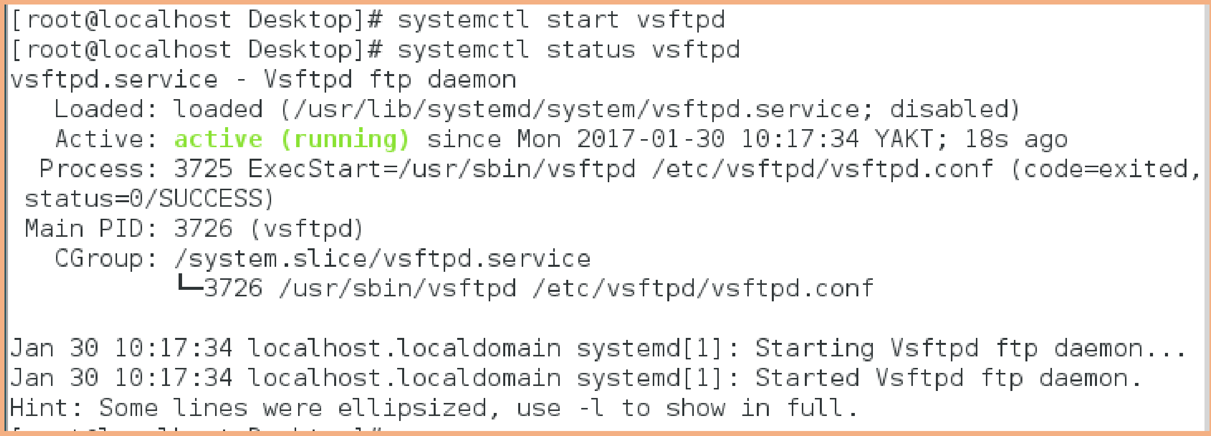
`chkconfig --list` *name*	`systemctl status` *name*`.service`
	`systemctl is-enabled` *name*`.service`
		Checks if a service is enabled.

`chkconfig --list`	`systemctl list-unit-files --type service`	Lists all services and checks if they are enabled.
`chkconfig --list`	`systemctl list-dependencies --after`	Lists services that are ordered to start before the specified unit.
`chkconfig --list`	`systemctl list-dependencies --before`	Lists services that are ordered to start after the specified unit.

How to management service using systemctl

1. To Start any services use below command

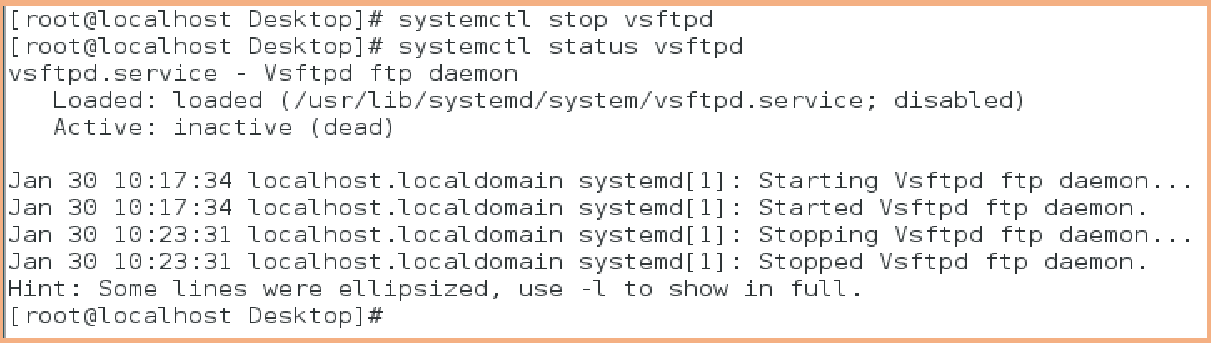
]#systemctlØstartØservice name.service



```
[root@localhost Desktop]# systemctl start vsftpd
[root@localhost Desktop]# systemctl status vsftpd
vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; disabled)
   Active: active (running) since Mon 2017-01-30 10:17:34 YAKT; 18s ago
  Process: 3725 ExecStart=/usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf (code=exited,
 status=0/SUCCESS)
 Main PID: 3726 (vsftpd)
   CGroup: /system.slice/vsftpd.service
           └─3726 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

Jan 30 10:17:34 localhost.localdomain systemd[1]: Starting Vsftpd ftp daemon...
Jan 30 10:17:34 localhost.localdomain systemd[1]: Started Vsftpd ftp daemon.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost Desktop]#
```

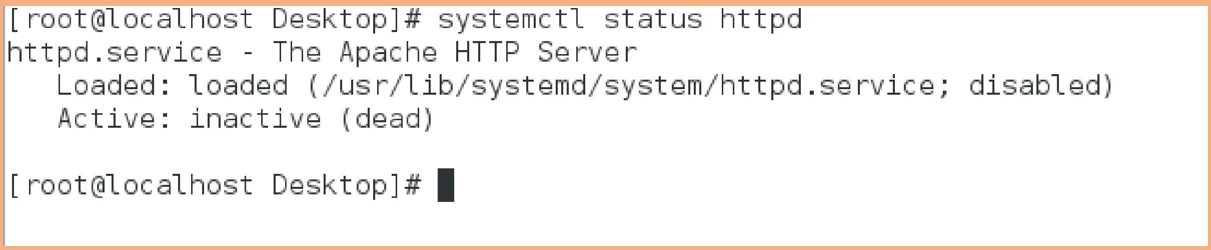
2. To stop any services run below command

#systemctlØstopØservice name.service



```
[root@localhost Desktop]# systemctl stop vsftpd
[root@localhost Desktop]# systemctl status vsftpd
vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; disabled)
   Active: inactive (dead)

Jan 30 10:17:34 localhost.localdomain systemd[1]: Starting Vsftpd ftp daemon...
Jan 30 10:17:34 localhost.localdomain systemd[1]: Started Vsftpd ftp daemon.
Jan 30 10:23:31 localhost.localdomain systemd[1]: Stopping Vsftpd ftp daemon...
Jan 30 10:23:31 localhost.localdomain systemd[1]: Stopped Vsftpd ftp daemon.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost Desktop]#
```

3. To see the status of service run below



```
[root@localhost Desktop]# systemctl status httpd
httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled)
   Active: inactive (dead)

[root@localhost Desktop]# █
```

systemctlØstatusØservice name.service

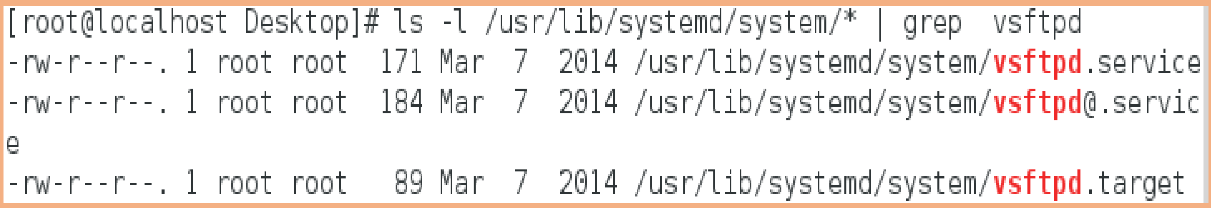
Note:-

Without the extension you can also manage any service

~]#systemctlØstartØvsftpd.service

To find all aliases that can be used for a particular unit, use:

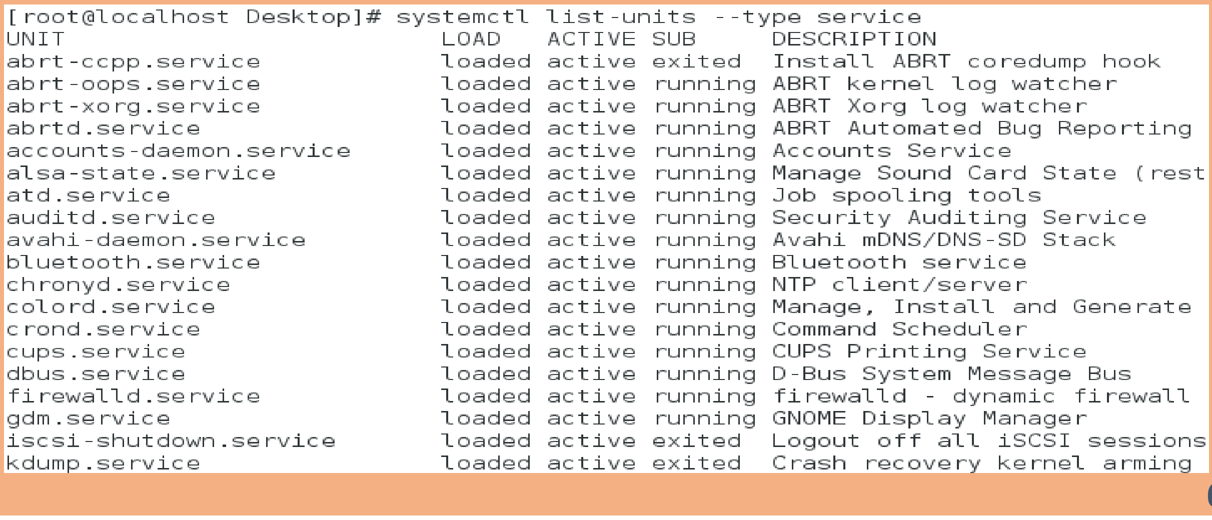
~]# lsØ -l Ø/usr/lib/systemd/system/* Ø| grepØ vsftpd



```
[root@localhost Desktop]# ls -l /usr/lib/systemd/system/* | grep  vsftpd
-rw-r--r--. 1 root root  171 Mar  7  2014 /usr/lib/systemd/system/vsftpd.service
-rw-r--r--. 1 root root  184 Mar  7  2014 /usr/lib/systemd/system/vsftpd@.servic
e
-rw-r--r--. 1 root root   89 Mar  7  2014 /usr/lib/systemd/system/vsftpd.target
```

To list all currently loaded service units, type the following at a shell prompt:

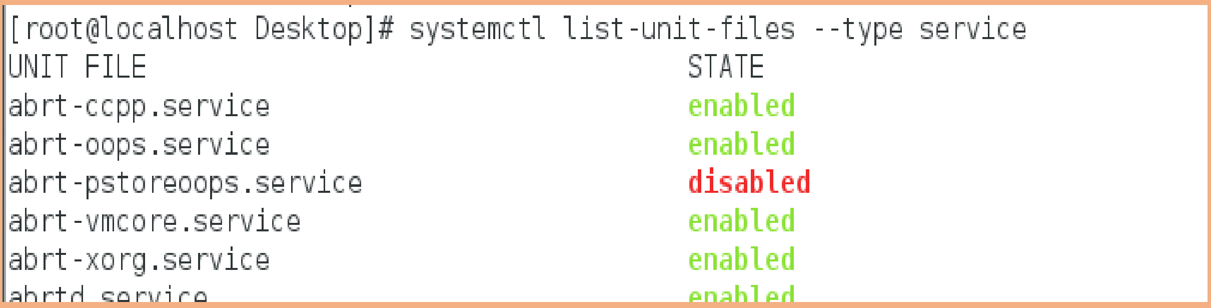
]# systemctlØlist-unitsØ--type service



```
[root@localhost Desktop]# systemctl list-units --type service
UNIT                          LOAD   ACTIVE SUB     DESCRIPTION
abrt-ccpp.service             loaded active exited  Install ABRT coredump hook
abrt-oops.service             loaded active running ABRT kernel log watcher
abrt-xorg.service             loaded active running ABRT Xorg log watcher
abrtd.service                 loaded active running ABRT Automated Bug Reporting
accounts-daemon.service       loaded active running Accounts Service
alsa-state.service            loaded active running Manage Sound Card State (rest
atd.service                   loaded active running Job spooling tools
auditd.service                loaded active running Security Auditing Service
avahi-daemon.service          loaded active running Avahi mDNS/DNS-SD Stack
bluetooth.service             loaded active running Bluetooth service
chronyd.service               loaded active running NTP client/server
colord.service                loaded active running Manage, Install and Generate
crond.service                 loaded active running Command Scheduler
cups.service                  loaded active running CUPS Printing Service
dbus.service                  loaded active running D-Bus System Message Bus
firewalld.service             loaded active running firewalld - dynamic firewall
gdm.service                   loaded active running GNOME Display Manager
iscsi-shutdown.service        loaded active exited  Logout off all iSCSI sessions
kdump.service                 loaded active exited  Crash recovery kernel arming
```

You can also list all available service units to see if they are enabled. To do so, type:

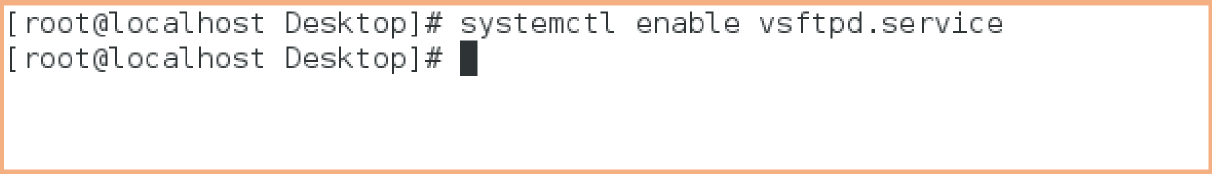
~]#systemctl Ølist-unit-files Ø--type service



```
[root@localhost Desktop]# systemctl list-unit-files --type service
UNIT FILE                                   STATE
abrt-ccpp.service                           enabled
abrt-oops.service                           enabled
abrt-pstoreoops.service                     disabled
abrt-vmcore.service                         enabled
abrt-xorg.service                           enabled
abrtd.service                               enabled
```

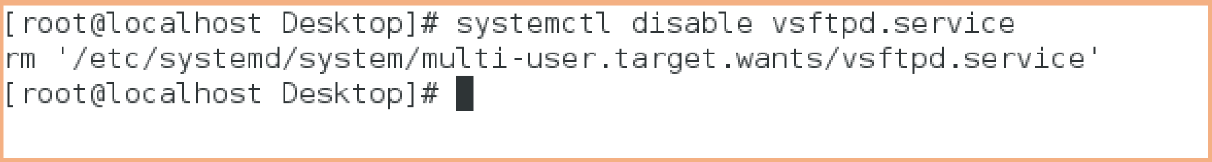
Enabling and Disabling a Service

To configure a service unit that corresponds to a system service to be automatically started at boot time,



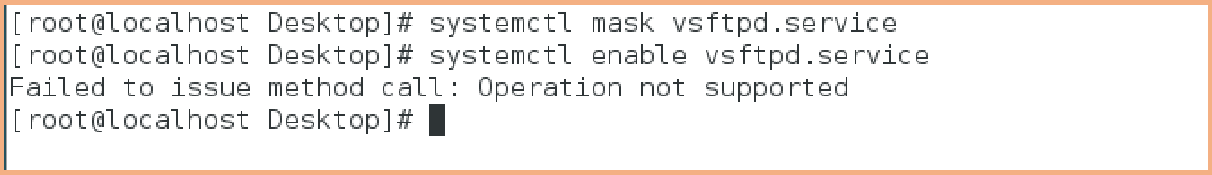
```
[root@localhost Desktop]# systemctl enable vsftpd.service
[root@localhost Desktop]# █
```

To prevent a service unit that corresponds to a system service from being automatically started at boot time



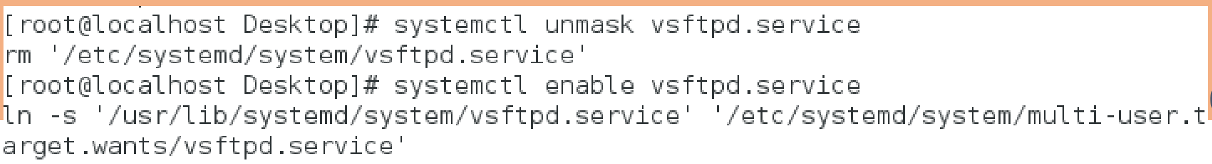
```
[root@localhost Desktop]# systemctl disable vsftpd.service
rm '/etc/systemd/system/multi-user.target.wants/vsftpd.service'
[root@localhost Desktop]# █
```

you can mask any service unit to prevent it from being started manually or by another service.



```
[root@localhost Desktop]# systemctl mask vsftpd.service
[root@localhost Desktop]# systemctl enable vsftpd.service
Failed to issue method call: Operation not supported
[root@localhost Desktop]# █
```

To revert this action and unmask a service unit

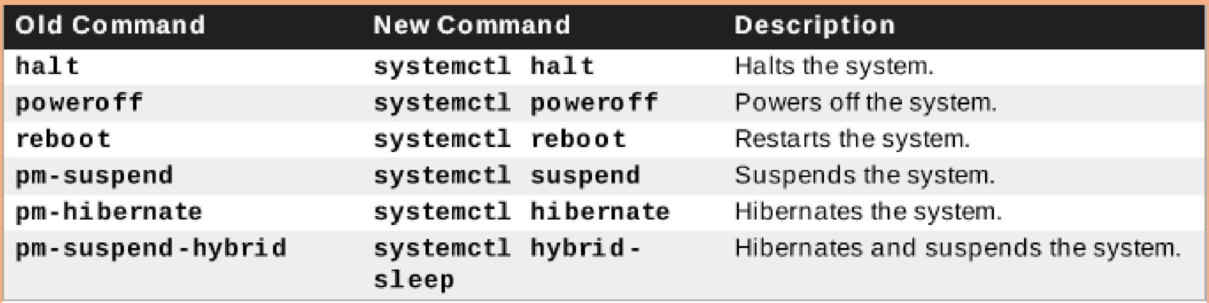


```
[root@localhost Desktop]# systemctl unmask vsftpd.service
rm '/etc/systemd/system/vsftpd.service'
[root@localhost Desktop]# systemctl enable vsftpd.service
ln -s '/usr/lib/systemd/system/vsftpd.service' '/etc/systemd/system/multi-user.t
arget.wants/vsftpd.service'
```

Shutting Down, Suspending, and Hibernating the System

In Red Hat Enterprise Linux 7, the systemctl utility replaces a number of power management commands used in previous versions of the Red Hat Enterprise Linux system.

Comparison of Power Management Commands with “systemctl”



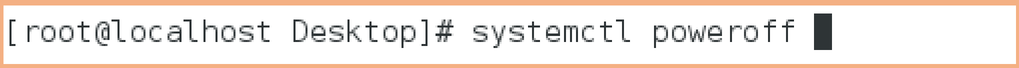
Old Command	New Command	Description
`halt`	`systemctl halt`	Halts the system.
`poweroff`	`systemctl poweroff`	Powers off the system.
`reboot`	`systemctl reboot`	Restarts the system.
`pm-suspend`	`systemctl suspend`	Suspends the system.
`pm-hibernate`	`systemctl hibernate`	Hibernates the system.
`pm-suspend-hybrid`	`systemctl hybrid-sleep`	Hibernates and suspends the system.

Shutting Down the System

The systemctl utility provides commands for shutting down the system, however the traditional shutdown command is also supported. Although the shutdown command will call the systemctl utility to perform the shutdown, it has an advantage in that it also supports a time argument.

To shut down the system and power off the machine

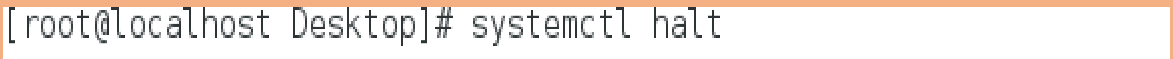
~]#systemctlØ poweroff



```
[root@localhost Desktop]# systemctl poweroff
```

To shut down and halt the system without powering off the machine

~]#systemctlØ halt

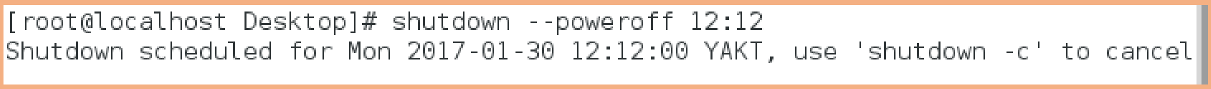


```
[root@localhost Desktop]# systemctl halt
```

Using the shutdown Command

To shut down the system and power off the machine at a certain time

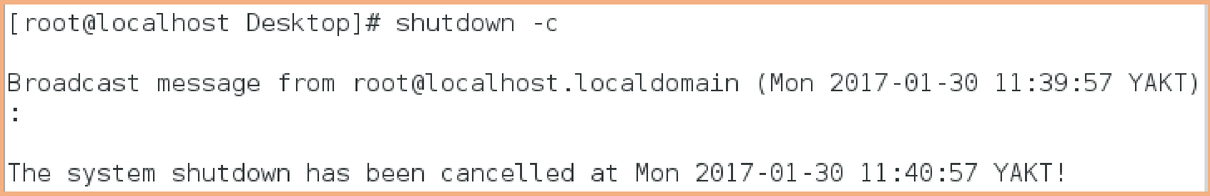
```
~]#shutdownØ --poweroffØ hh:mm
```



```
[root@localhost Desktop]# shutdown --poweroff 12:12
Shutdown scheduled for Mon 2017-01-30 12:12:00 YAKT, use 'shutdown -c' to cancel
```

A pending shutdown can be canceled by the root user as follows:

```
~]#shutdownØ –c
```

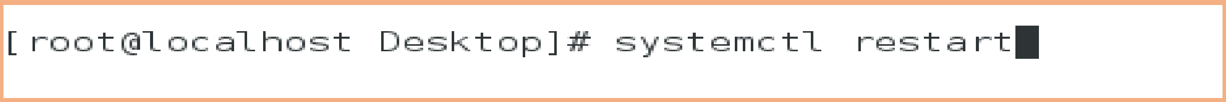


```
[root@localhost Desktop]# shutdown -c

Broadcast message from root@localhost.localdomain (Mon 2017-01-30 11:39:57 YAKT)
:

The system shutdown has been cancelled at Mon 2017-01-30 11:40:57 YAKT!
```

Restarting the System

To restart the system

```
~]#systemctlØ reboot
```

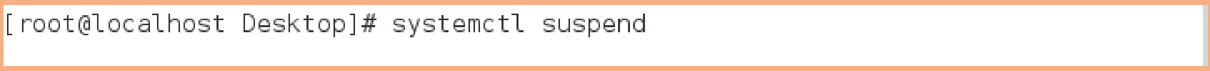


```
[root@localhost Desktop]# systemctl restart
```

Suspending the System

To suspend the system

```
~]#systemctlØ suspend
```

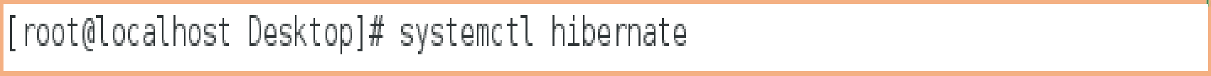


```
[root@localhost Desktop]# systemctl suspend
```

Hibernating the System

To hibernate the system

```
~]#systemctlØ hibernate
```

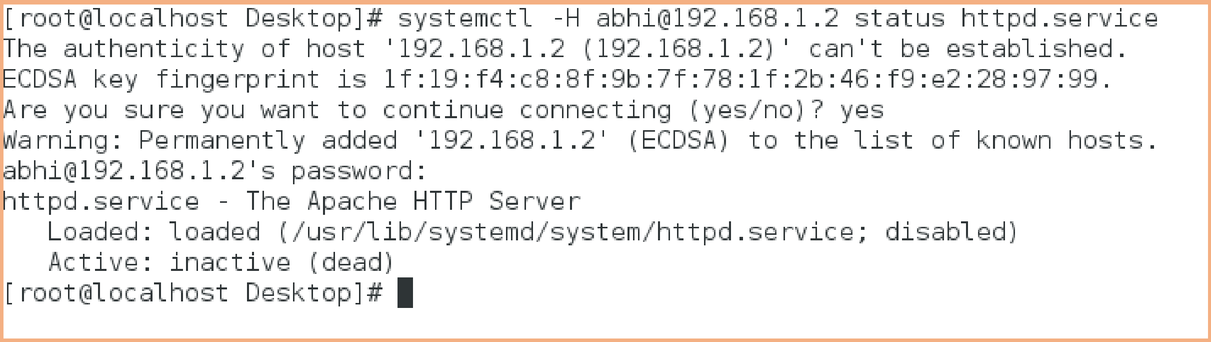


```
[root@localhost Desktop]# systemctl hibernate
```

Controlling systemd on a Remote Machine

In addition to controlling the systemd system and service manager locally, the systemctl utility also allows you to interact with systemd running on a remote machine over the SSH protocol. Provided that the sshd service on the remote machine is running, you can connect to this machine by running the systemctl command with the --host or -H command line option

~]#systemctlØ --hostØ user_name@host_name command



```
[root@localhost Desktop]# systemctl -H abhi@192.168.1.2 status httpd.service
The authenticity of host '192.168.1.2 (192.168.1.2)' can't be established.
ECDSA key fingerprint is 1f:19:f4:c8:8f:9b:7f:78:1f:2b:46:f9:e2:28:97:99.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.2' (ECDSA) to the list of known hosts.
abhi@192.168.1.2's password:
httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled)
   Active: inactive (dead)
[root@localhost Desktop]#
```

Additional Resources

For more information on systemd and its usage on Red Hat Enterprise Linux 7, see the resources listed below.

Installed Documentation:

systemctl(1) — The manual page for the systemctl command line utility provides a complete list of supported options and commands.

systemd(1) — The manual page for the systemd system and service manager provides more information about its concepts and documents available command line options and environment variables, supported configuration files and directories, recognized signals, and available kernel options.

systemd-delta(1) — The manual page for the systemd-delta utility that allows to find extended and overridden configuration files.

systemd.unit(5) — The manual page named systemd.unit provides in-depth information about systemd unit files and documents all available configuration options.

systemd.service(5) — The manual page named systemd.service documents the format of service unit files.

systemd.target(5) — The manual page named systemd.target documents the format of target unit files.

systemd.kill(5) — The manual page named systemd.kill documents the configuration of the process killing procedure.

WEB SERVER

1. First install Http package

```
#yumØinstallØhttpd
```

2. Now give webserver detail to httpd.conf file.

```
#viØ/etc/httpd/conf/httpd.conf (go to last of the document and put detail)
{<VirtualHostØserverip:80>
ServerNameØwww.xyz.com
DocumentRootØ /var/www/html
</VirtualHost>}
```

3. Now check httpd syntax

```
#httpdØ-t
```

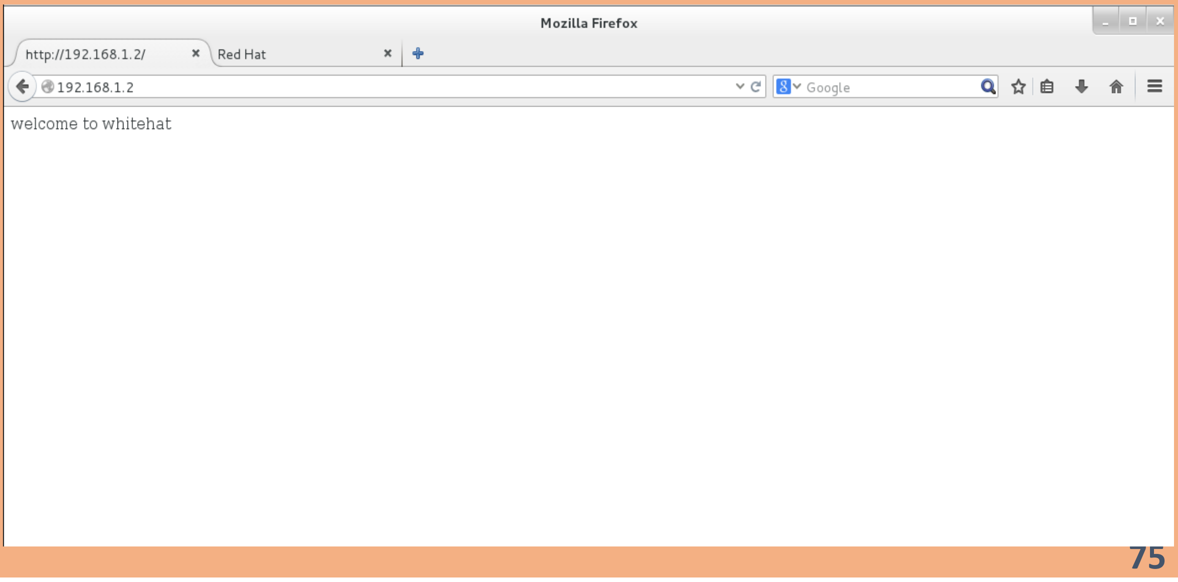
4. Now host website in /var/www/html/ directory
   (Note: httpd default directory is /var/www/html/)
5. Now start httpd service and enable service

```
#systemctlØstartØhttpd.service
#systemctlØenableØhttpd.service
```

6. Now on firewall port of http

```
# firewall-cmdØ--permanentØ--add-service=http
# firewall-cmdØ--reload
```

7. Now open your browser and check your web server



welcome to whitehat

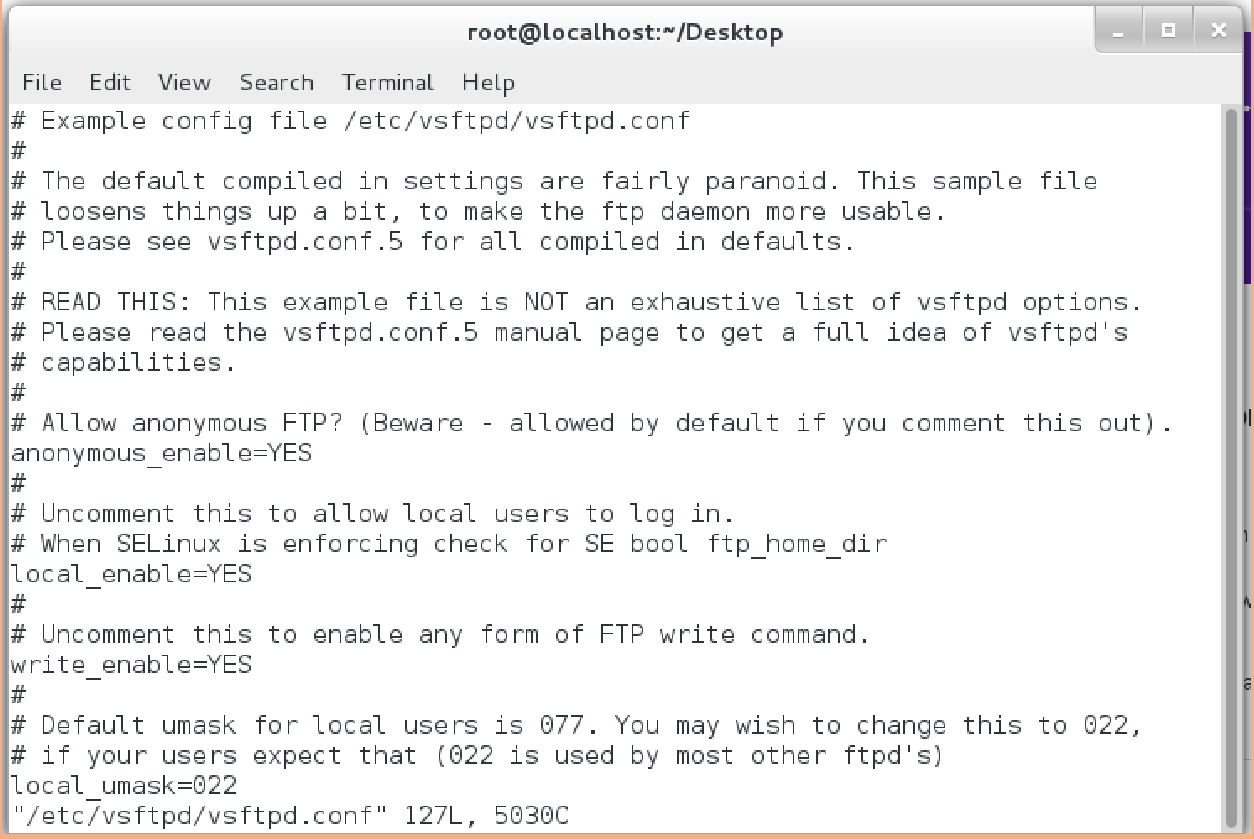
FTP Server

1.Install vsftpd package

```
#yumØinstallØvsftpd
```

2.Now check ftp configuration file

```
#viØ/etc/vsftpd/vsftpd.conf
```



```
# Example config file /etc/vsftpd/vsftpd.conf
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=YES
#
# Uncomment this to allow local users to log in.
# When SELinux is enforcing check for SE bool ftp_home_dir
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
local_umask=022
"/etc/vsftpd/vsftpd.conf" 127L, 5030C
```

3.Now start your ftp service

```
#systemctlØstartØvsftpd.service

#systemctlØenableØvsftpd.service
```

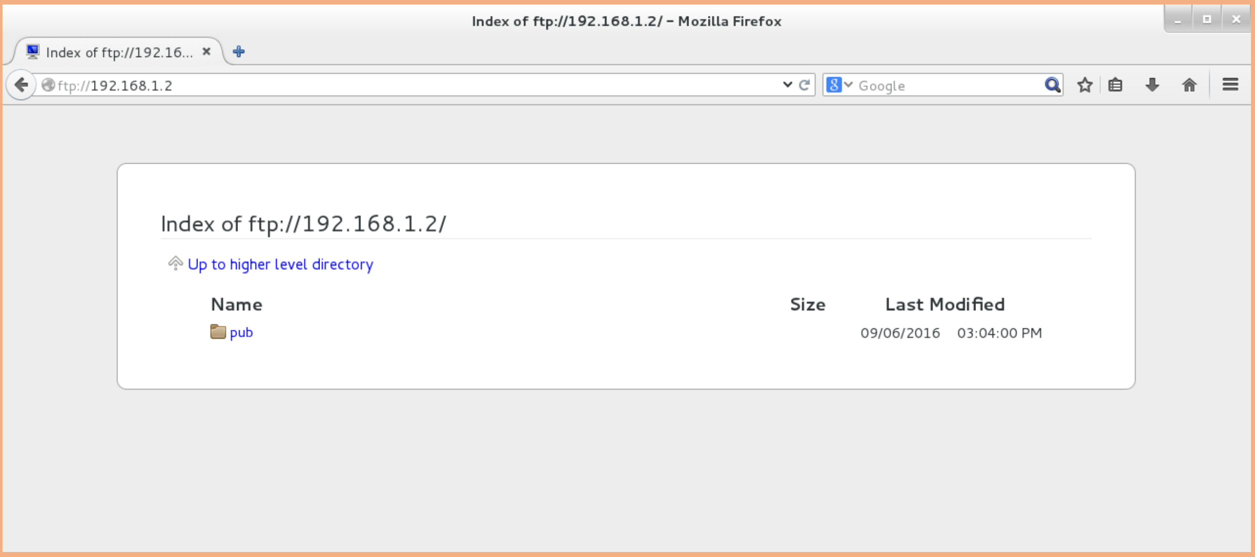
4.Now on firewall

```
#firewall-cmdØ--permanentØ--add-service=ftp

#firewall-cmdØ--reload
```

Note: FTP default directory is /var/ftp/pub

5.Store Data in /var/ftp/pub/ directory

6.Open browser and check ftp server



Index of ftp://192.168.1.2/

Up to higher level directory

Name	Size	Last Modified
pub		09/06/2016 03:04:00 PM

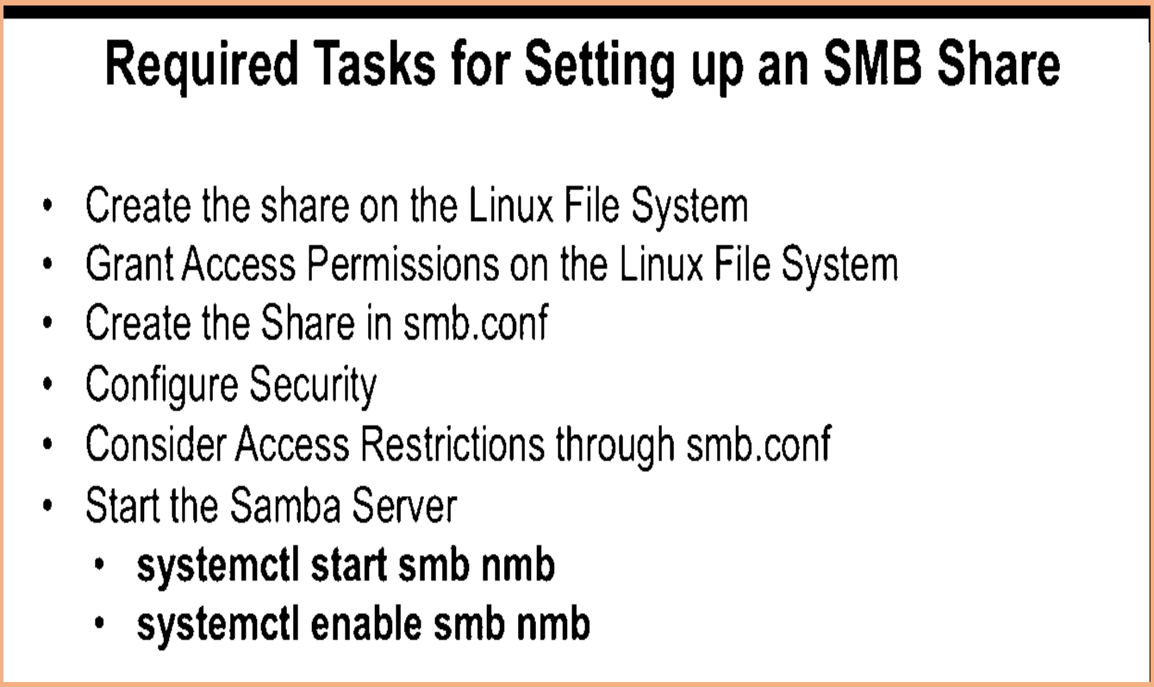
Accessing Network File Sharing Services

A network file system is a file system that, instead of being provided by a block device like a hard drive, is provided by a network attached storage server to multiple hosts over a network. Clients access the remote storage through a special file system protocol and format.

There are two primary protocols which are used in Linux to access network file systems: NFS and CIFS. NFS, the Network file system, acts much like a standard file system for Linux, UNIX and similar operating systems. CIFS the Common Internet File system, is the standard network file system for Microsoft windows systems.

Samba Server

Samba use CIFS file system to mount a share between window and Linux OS For that you have samba server and other side clients.



Required Tasks for Setting up an SMB Share

- Create the share on the Linux File System
- Grant Access Permissions on the Linux File System
- Create the Share in smb.conf
- Configure Security
- Consider Access Restrictions through smb.conf
- Start the Samba Server
  - **systemctl start smb nmb**
  - **systemctl enable smb nmb**

What Samba can do:

- Serve directory trees and printers to Linux, UNIX, and Windows clients.
- Assist in network browsing (with NetBIOS).
- Authenticate Windows domain logins.
- Provide Windows Internet Name Service (WINS) name server resolution.
- Act as a Windows NT®-style Primary Domain Controller (PDC).
- Act as a Backup Domain Controller (BDC) for a Samba-based PDC.
- Act as an Active Directory domain member server.
- Join a Windows NT/2000/2003/2008 PDC/Windows Server 2012.

What Samba cannot do:

- Act as a BDC for a Windows PDC (and vice versa)
- File and Print Servers
- Act as an Active Directory domain controller

Samba Daemons and Related Services

Samba is comprised of three daemons (smbd, nmbd, and winbindd). Three services (smb, nmb, and winbind) control how the daemons are started, stopped, and other service-related features. These services act as different init scripts. Each daemon is listed in detail below, as well as which specific service has control over it.

smbd

The smbd server daemon provides file sharing and printing services to Windows clients. In addition, it is responsible for user authentication, resource locking, and data sharing through the SMB protocol. The default ports on which the server listens for SMB traffic are TCP ports 139 and 445.The smbd daemon is controlled by the smb service.

nmbd

The nmbd server daemon understands and replies to NetBIOS name service requests such as those produced by SMB/CIFS in Windows-based systems. These systems include Windows 95/98/ME, Windows NT, Windows 2000, Windows XP, and LanManager clients. It also participates in the browsing protocols that make up the Windows Network Neighborhood view. The default port that the server listens to for NMB traffic is UDP port 137.

The nmbd daemon is controlled by the nmb service.

winbindd

The winbind service resolves user and group information received from a server running Windows NT, 2000, 2003, Windows Server 2008, or Windows Server 2012. This makes Windows user and group information understandable by UNIX platforms. This is achieved by using Microsoft RPC calls, Pluggable Authentication Modules (PAM), and the Name Service Switch (NSS). This allows Windows NT domain and Active Directory users to appear and operate as UNIX users on a UNIX machine. Though bundled with the Samba distribution, the winbind service is controlled separately from the smb service.

The winbind daemon is controlled by the winbind service and does not require the smb service to be started in order to operate. winbind is also used when Samba is an Active Directory member, and may also be used on a Samba domain controller (to implement nested groups and interdomain trust). Because winbind is a client-side service used to connect to Windows NT-based servers, further discussion of winbind is beyond the scope of this chapter.

How to create samba server

1. Install samba

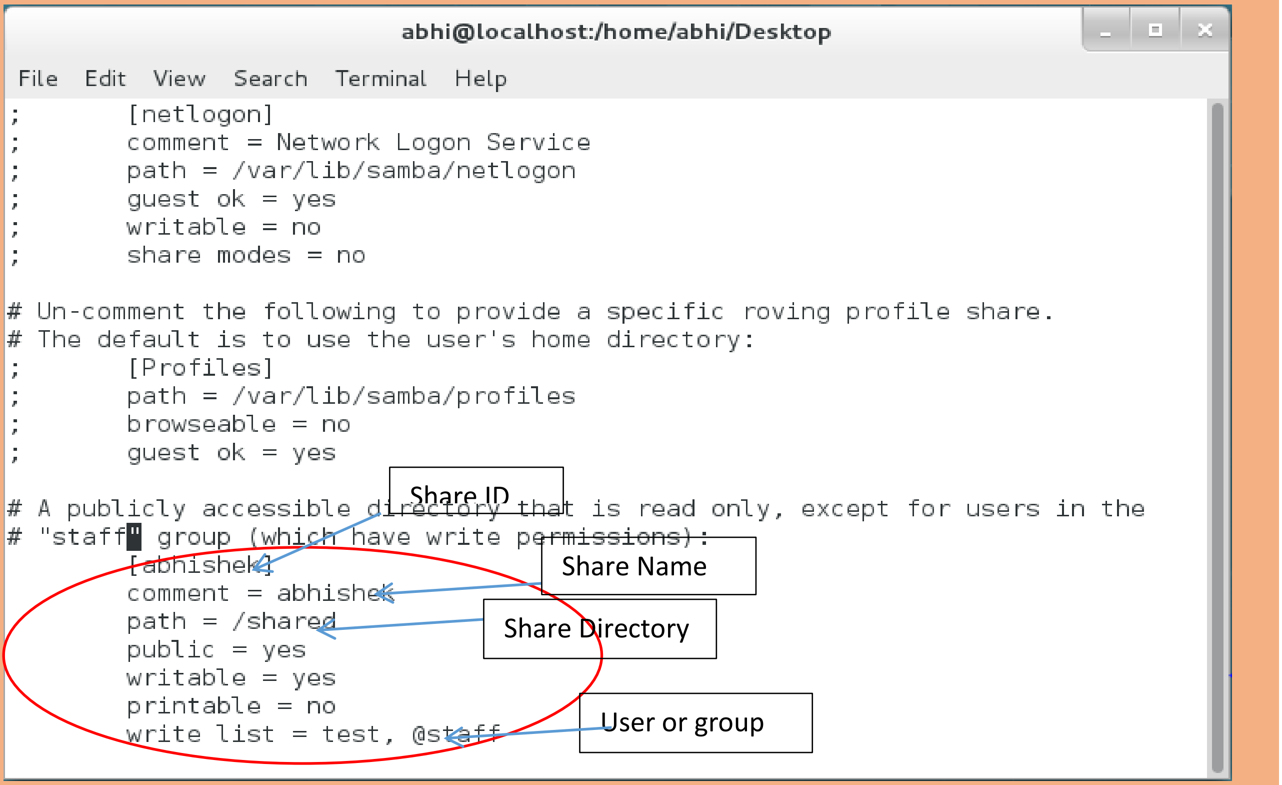
```
#yumØinstallØsamba
```

2. Now create directory for sharing

```
#mkdirØ/directory name
```

3. Now enter sharing information in smb.conf file.

```
#viØ/etc/samba/smb.conf
```



abhi@localhost:/home/abhi/Desktop
File Edit View Search Terminal Help
Share ID
Share Name
Share Directory
User or group

```
;       [netlogon]
;       comment = Network Logon Service
;       path = /var/lib/samba/netlogon
;       guest ok = yes
;       writable = no
;       share modes = no

# Un-comment the following to provide a specific roving profile share.
# The default is to use the user's home directory:
;       [Profiles]
;       path = /var/lib/samba/profiles
;       browseable = no
;       guest ok = yes

# A publicly accessible directory that is read only, except for users in the
# "staff" group (which have write permissions):
        [abhishek]
        comment = abhishek
        path = /shared
        public = yes
        writable = yes
        printable = no
        write list = test, @staff
```

4. Create user and set password for samba server

```
# useradd Øu1
# passwd Øu1
Changing password for user u1.
New password: * * * * * (u1)
BAD PASSWORD: it is WAY too short
BAD PASSWORD: is too simple
Retype new password: * * * * *(u1)
passwd: all authentication tokens updated successfully.
```

5. Now set smb password also, so client can access share by using smb password

```
# smbpasswd Ø-a Øu1
New SMB password:* * * * * (u1@123)
Retype new SMB password:* * * * *(u1@123)
Added user u1.
# smbpasswd-e Øu1
Enabled user u1.
```

6. Start the service of smb and nmb

```
#systemctlØstartØsmbØnmb
```

7. Enable the service of smb and nmb

```
#systemctlØenableØsmbØnmb
```

8. Now on the firewall port of samba and reload

```
#firewall-cmdØ--permanent--add-service=samba
#firewall-cmdØ--reload
```

9. Now set SElinux on share directory

```
#semanageØfcontextØ-tØ-oØsamba_share_tØ
“sharedirectory(/.*)?”
#restoreconØ-RØ-vØsharedirectory
```

Client side (linux)

1. Check the share

```
#smbclientØ-LØserveripØ-UØusername
```

2. Install cifs-utils

```
#yumØcifs-utils
```

3. Now create directory to mount

```
#mkdirØdirectoryname
```

4. Mount the share on client

```
#mountØ-oØusername=usernameØ//serverip/sharename

 Ødirectoryname
```

For window client machine:-

{Go to Start-Run-\\serverip\
type username and password
username - u1
password - *****(u1@123)

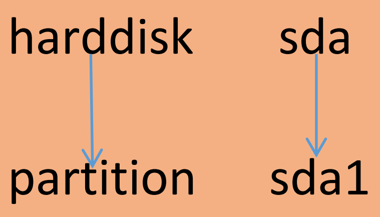
Now your are able to access kavi directory and files from it & user u1 home directory as well.)}

Managing Partition

Simple Partition and file systems

Storage is a basic need of every computer system. Red Hat Enterprise Linux includes powerful tools for managing many types of storage devices in a wide range of scenarios.

fdisk is a utility to manage disk partitions. You can view disks and their partitioning by running the utility with the -l option and the name of the disk (fdiskØ-l).(Note: partition default directory is /dev)



Disk Management.

Disk Partitioning :-

1. Primary Partition
2. Extended Partition
3. Logical Partition

How to create a new partition

1. Create partition

# fdiskØ/dev/sda

```
Command (m for help): m                    Command action
a   toggle a bootable flag
b   edit bsd disklabel
c   toggle the dos compatibility flag
d   delete a partition
l  list known partition types
m  print this menu
n   add a new partition
o   create a new empty DOS partition table
p   print the partition table
q   quit without saving changes
s   create a new empty Sun disklabel
t   change a partition's system id
u  change display/entry units
v  verify the partition table
w  write table to disk and exit
x   extra functionality (experts only)

Command (m for help): p

Disk /dev/sda: 42.9 GB, 42949672960 bytes
255 heads, 63 sectors/track, 5221 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000d5093

Device Boot             Start          End          Blocks        Id      System
/dev/sda1         *      1              131     1048576       83      Linux
Partition 1 does not end on cylinder boundary.
/dev/sda2               131             1437    10485760      83      Linux
/dev/sda3       1437          2742    10485760       83      Linux
/dev/sda4       2742          5222    19921920       5       Extended
/dev/sda5       2742          2873    1048576        83      Linux
/dev/sda6       2873          3003    1048576        82      Linux swap / Solaris
/dev/sda7       3004          3134    1048576        83      Linux
/dev/sda8       3134          3160    204800         83      Linux
/dev/sda9       3160          3200    327200+        83      Linux



Command (m for help): n
First cylinder (3003-5222, default 3003): 3201

Last cylinder, +cylinders or +size{K,M,G} (3201-5222, default 5222):
3250
```

```
Command (m for help): p
Disk /dev/sda: 42.9 GB, 42949672960 bytes
255 heads, 63 sectors/track, 5221 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000d5093

Device Boot          Start        End          Blocks        Id       System
/dev/sda1        *    1           131     1048576        83       Linux
Partition 1 does not end on cylinder boundary.
/dev/sda2             131         1437   10485760        83       Linux
/dev/sda3        1437        2742   10485760        83       Linux
/dev/sda4        2742        5222   19921920        5        Extended
/dev/sda5        2742        2873   1048576         83       Linux
/dev/sda6        2873        3003   1048576         82       Linux swap / Solaris
/dev/sda7        3004        3134   1048576         83       Linux
/dev/sda8        3134        3160   204800          83       Linux
/dev/sda9        3160        3200   327200          83       Linux
/dev/sda10            3201        3250   401593+       83       Linux

Command (m for help): w                                   …….<PRESS ENTER>

The partition table has been altered!

Calling ioctl() to re-read partition table.
WARNING: Re-reading the partition table failed with error 16: Device or resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks
```

2. Now reboot your system

```
#reboot
```

3. Now format your partition

```
#mkfsØ-tØfilesystemØ/dev/partition
```

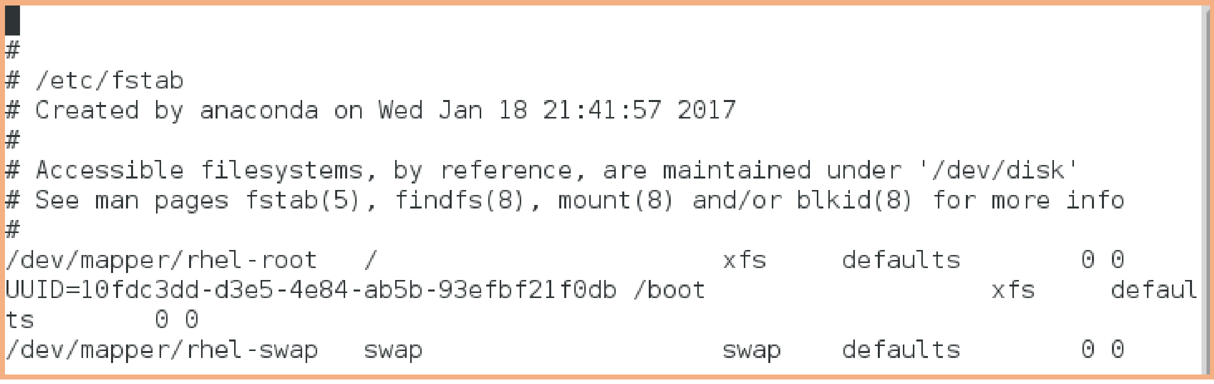
4. Now create directory for mount the partition

```
#mkdirØ/directory
```

5. Now mount your partition on that directory

```
#mountØ-tØfilesystemØ/dev/partitionØ/directory
```

6. Now add an entry to /etc/fstab #viØ/etc/fstab



```

#
# /etc/fstab
# Created by anaconda on Wed Jan 18 21:41:57 2017
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/mapper/rhel-root   /                       xfs     defaults        0 0
UUID=10fdc3dd-d3e5-4e84-ab5b-93efbf21f0db /boot                   xfs     defaul
ts        0 0
/dev/mapper/rhel-swap   swap                    swap    defaults        0 0
```

Disk Quota Management

Disk Quotas are used to limit a user's or a group of users' ability to consume disk space. This prevents a small group of users from monopolizing disk capacity and potentially interfering with other users or the entire system.Disk quotas are commonly used by ISPs, by Web hosting companies, on FTP sites, and on corporate file servers to ensure continued availability of their systems. Using disk quota administrator can restrict user in two ways :-

1. Restricting user or group by creating files in a specific location.
2. Restricting user or group by the disk space in a specific location.

Disk Quota Terms

1. **Soft Link** = Disk space a user can use
2. **Hard Link** = Absolute limit a user can use
3. **Grace Periods** = Time duration till user can use hard limit space
4. **1 inode** = 1KB
5. **dd** = used to create a blank file of specific size
6. **required RPM** = quata-*
7. **/etc/fstab option** = userquota, grpquota
8. **Quota files** = aquota.user. aquota.group
9. **Necessary Commands** = mount, quotarun, quotacheck, edquota, quotaoff, quotaon

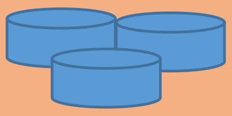
LVM

(LOGICAL VOLUME MANAGEMENT)

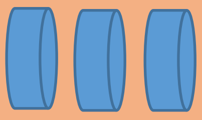
This utility to manage partition means if you create partition you can't extend or shrink that partition normally. But through LVM you can do this things LVM was first developed by HP for it HPUX operating system.

(Note: It is default volume management system in RHEL-7) you can manage LVM through command line tools, It use a collection of disks, the disks can be of different size, this disk referred as physical volume, physical volume are collected in to volume groups, logical volumes core component of LVM it contain file system and it created from physical volumes. It done in online mean you need to reboot your system after resizing LVM. LVS and PVS are broken-up in to chunks of data known as extents LVMs can be grown or shrink by increasing or decreasing the extents of disk space used in MBs or GBs, LVM provide the backup facility through snapshots during the backup, no down time is needed, the /boot cannot be placed on LVM.

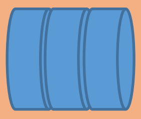
Process for create LVM.



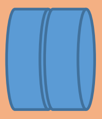
1. Create physical storage partition



2. Create physical volume



3. Create Volume Group



4. Create Logical Volume

LVM Definitions

Physical volume : A partition that marked as usable space for LVM on an MBR disk, marked partition type 0X86.

Volume group: A collection of one or more physical volumes can be thought of as a virtual disk drive.

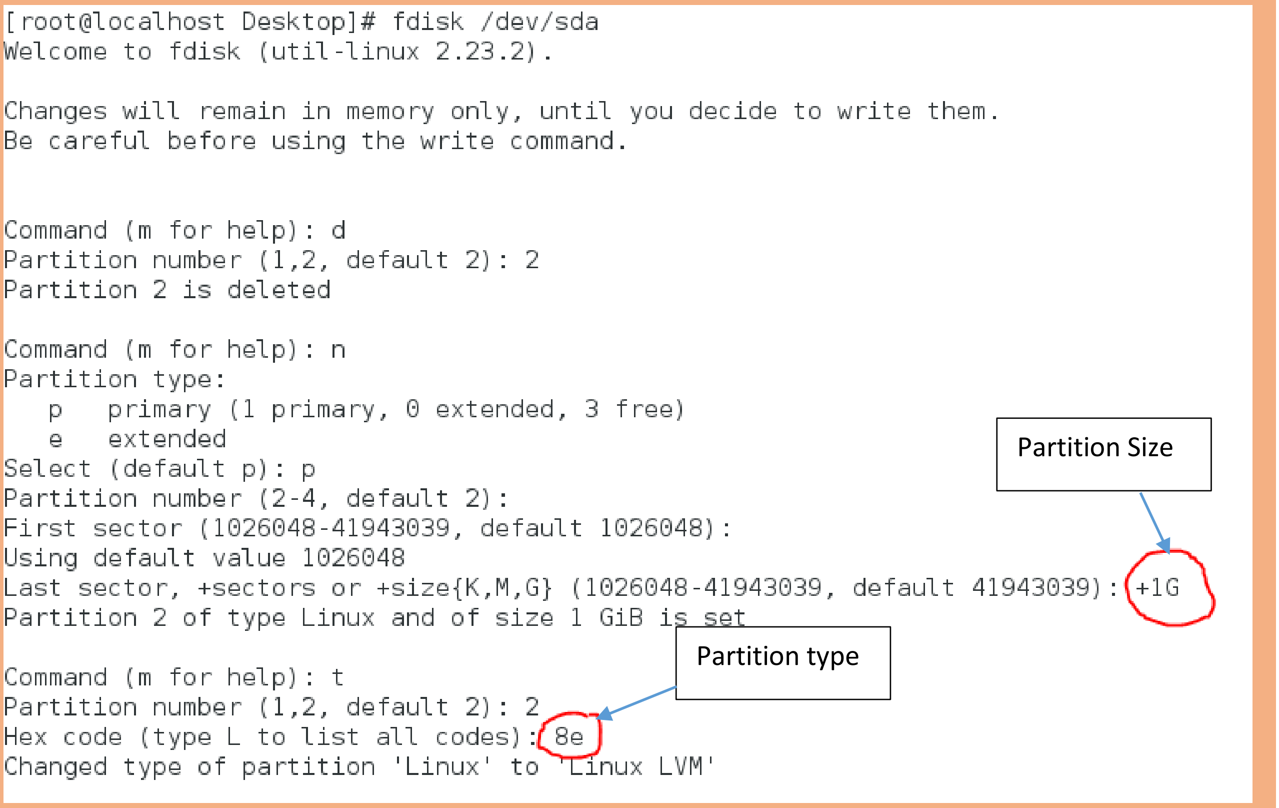
Logical volume: It can be thought of as a virtual partition of the volume group. This is formatted with file system and used like a partition.

Physical Extent: A disks space is allocated from physical volume by the volume group to logical volumes in large chunks called physical extents.

How to create LVM

1. Create standard partition and give type 8e

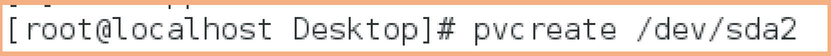
   #fdiskØ/dev/sda



```
[root@localhost Desktop]# fdisk /dev/sda
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.


Command (m for help): d
Partition number (1,2, default 2): 2
Partition 2 is deleted

Command (m for help): n
Partition type:
   p   primary (1 primary, 0 extended, 3 free)
   e   extended
Select (default p): p
Partition number (2-4, default 2):
First sector (1026048-41943039, default 1026048):
Using default value 1026048
Last sector, +sectors or +size{K,M,G} (1026048-41943039, default 41943039): +1G
Partition 2 of type Linux and of size 1 GiB is set

Command (m for help): t
Partition number (1,2, default 2): 2
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'
```

2. Now convert physical partition to physical volume

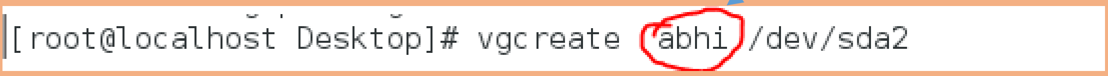
   #pvcreateØ/dev/sda2



```
[root@localhost Desktop]# pvcreate /dev/sda2
```

3. Now create a group of physical volume

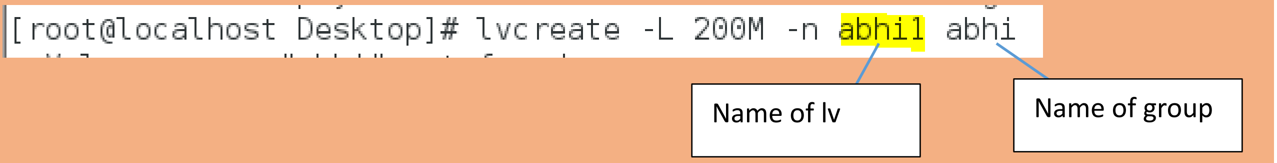
   #vgcreateØnameØ/dev/sda2Ø/dev/sda3



```
[root@localhost Desktop]# vgcreate abhi /dev/sda2
```

4. Now create Logical volume

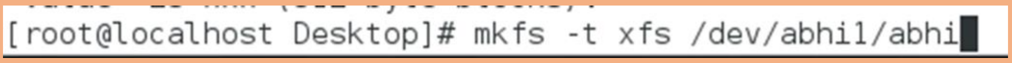
   #lvcreateØ-LØsizeØ-nØnameØvolumegroupname



```
[root@localhost Desktop]# lvcreate -L 200M -n abhi1 abhi
```

5. You create file system on it

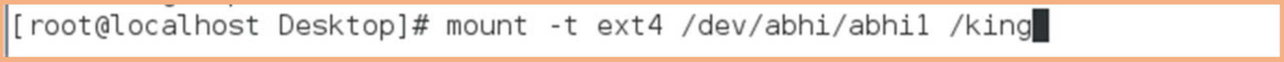
#mkfsØ-tØxfsØ/dev/volumegroup/logicalvolume



```
[root@localhost Desktop]# mkfs -t xfs /dev/abhi1/abhi
```

6. Now mount your partition

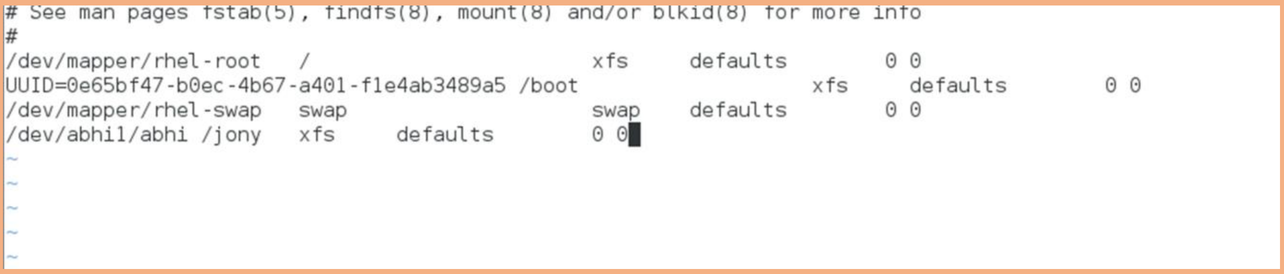
#mountØ-tØext4Ø/dev/vg/lvØ/directory



```
[root@localhost Desktop]# mount -t ext4 /dev/abhi/abhi1 /king
```

7. Now entry this in /etc/fstab

#viØ/etc/fstab



```
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/mapper/rhel-root   /                       xfs     defaults        0 0
UUID=0e65bf47-b0ec-4b67-a401-f1e4ab3489a5 /boot                   xfs     defaults        0 0
/dev/mapper/rhel-swap   swap                    swap    defaults        0 0
/dev/abhi1/abhi /jony   xfs     defaults        0 0
~
~
~
~
~
```

LVM Basic Command

- pvdisplay, pvs: This command use to see physical volume details
- vgdisplay, vgs: This command use to see volume group details
- lvdisplay, lvs : This command use to see logical volume details

SWAP

Swap Space Concepts

A swap space is an area of a disk which can be used with the Linux kernel memory management subsystem. Swap spaces are used to supplement the system RAM by holding inactive pages of memory. The combined system RAM plus swap spaces is called virtual memory.

Create Swap Memory

Now create SWAP partition from newly added disk using below commands

```
[root@localhost ~]# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.23.2).



Changes will remain in memory only, until you decide to write them.

Be careful before using the write command.



Device does not contain a recognized partition table

Building a new DOS disklabel with disk identifier 0x989078c0.



Command (m for help): n

Partition type:

   p   primary (0 primary, 0 extended, 4 free)

   e   extended

Select (default p): p
```

```
Partition number (1-4, default 1): 1

First sector (2048-2097151, default 2048): ENTER

Using default value 2048

Last sector, +sectors or +size{K,M,G} (2048-2097151, default 2097151): ENTER

Using default value 2097151

Partition 1 of type Linux and of size 1023 MiB is set



Command (m for help): t

Selected partition 1

Hex code (type L to list all codes): 82

Changed type of partition 'Linux' to 'Linux swap / Solaris'



Command (m for help): w

The partition table has been altered!



Calling ioctl() to re-read partition table.

Syncing disks.

[root@localhost ~]# partx -a /dev/sdb

partx: /dev/sdb: error adding partition 1

[root@localhost ~]# partx -a /dev/sdb

partx: /dev/sdb: error adding partition 1
```

2.You can verify your newly created SWAP partition by using fdisk command as below.

```
[root@localhost ~]# fdisk -l /dev/sdb



Disk /dev/sdb: 1073 MB, 1073741824 bytes, 2097152 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk label type: dos

Disk identifier: 0x989078c0



   Device Boot      Start         End      Blocks   Id  System

/dev/sdb1         2048    2097151    1047552   82  Linux swap / Solaris
```

3. Now format this newly created partition as SWAP using mkswap command.

```
[root@localhost ~]# mkswap /dev/sdb1

Setting up swapspace version 1, size = 1047548 KiB

no label, UUID=4a28616d-065f-4123-b801-97d2b25019b7
```

4. After formatting the disk with SWAP, you will have to enable the SWAP with swapon command and then you can verify it.

```
[root@localhost ~]# swapon /dev/sdb1

[root@localhost ~]#
```

5. Now to make newly added SWAP memory to be available at next boot, put its entry in fstab file like below.

```
UUID=4a28616d-065f-4123-b801-97d2b25019b7      swap    swap    defaults 0  0
```

This is how you can increase or add SWAP memory in Linux 7 server easily.

SElinux

(Security Enhanced Linux)

SElinux is an additional layer of system security. SElinux is secure our data to system service which can compromised. Whenever we talk about security we always point out on user based security which is called (DAC) Discretionary Access Control. SElinux provide object based security which control by policy & rules and is Mandatory Access Control (MAC). SElinux is a set of security rules that determine which process can access which files, directories, and ports.

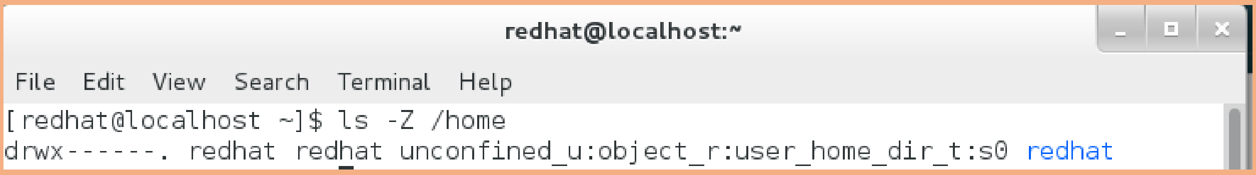
Where SElinux can set

- User
- Process
- Folder & Files.

How SElinux work

SElinux confirm the Access according to matching of labels which contain by files and folder with the appropriate service. Labels or context is like sticker which you can find on the properties of files. If you want to see labels of files & folder you can run below command

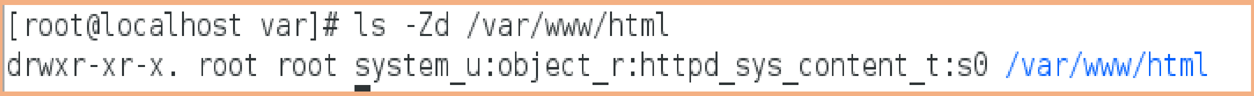
Example: # lsØ-ZØ/home.



```
[redhat@localhost ~]$ ls -Z /home
drwx------. redhat redhat unconfined_u:object_r:user_home_dir_t:s0 redhat
```

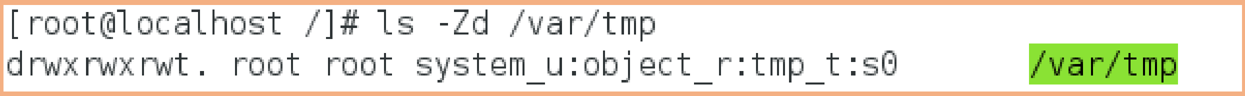
SElinux labels have several context: user, role, type and sensitivity. The targeted policy, which is the default policy enabled in RHEL, bases its rules on the third context: the type context. Type context names usually end with _t. The type context for the webserver is httpd_t . The type context for files and directories normally found in /var/www/html is httpd_sys_content_t . There is a policy rule that permits Apache (web server process running as httpd_t) to access files and directories with a

Context normally found in /var/www/html and other web server directories (httpd_sys_content_t).



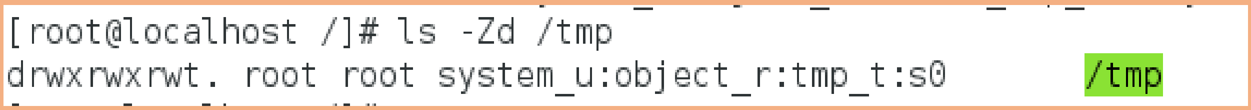
```
[root@localhost var]# ls -Zd /var/www/html
drwxr-xr-x. root root system_u:object_r:httpd_sys_content_t:s0 /var/www/html
```

The type contexts for files and directories normally found in /tmp and /var/tmp is tmp_t.



```
[root@localhost /]# ls -Zd /var/tmp
drwxrwxrwt. root root system_u:object_r:tmp_t:s0       /var/tmp
```

&

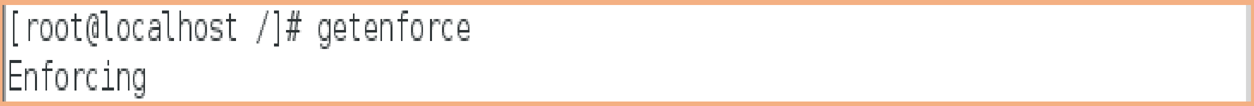


```
[root@localhost /]# ls -Zd /tmp
drwxrwxrwt. root root system_u:object_r:tmp_t:s0       /tmp
```

SElinux modes

- Enforcing mode:
  SElinux actively denies access to the web server attempting to read files with tmp_t type context.
- Permissive mode:
  This mode is often used to troubleshoot issues. In permissive mode, SElinux allows all interactions, even if there is no explicit rule, and it logs those interactions it would have denied in enforcing mode.
- Disabled mode:
  In this mode SElinux is disable no prevention and logs are created

To display the current SElinux mode in effect, use the **getenforce** command.

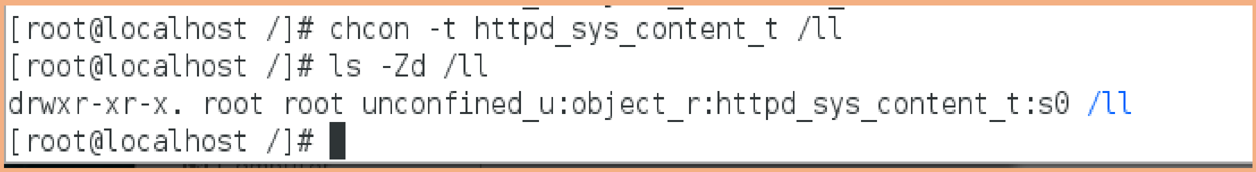


```
[root@localhost /]# getenforce
Enforcing
```

How to change SElinux context of file and folder.

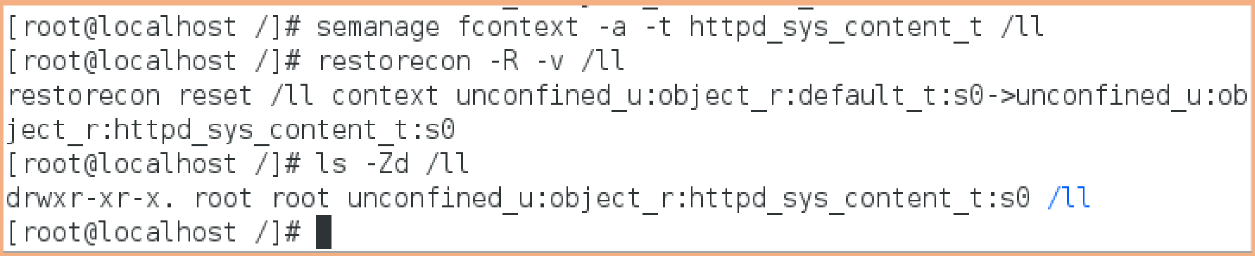
There are two command use to change

1. chcon (It use to change temporary SElinux)



```
[root@localhost /]# chcon -t httpd_sys_content_t /ll
[root@localhost /]# ls -Zd /ll
drwxr-xr-x. root root unconfined_u:object_r:httpd_sys_content_t:s0 /ll
[root@localhost /]# 
```

2. semanage (It use to change permanently)



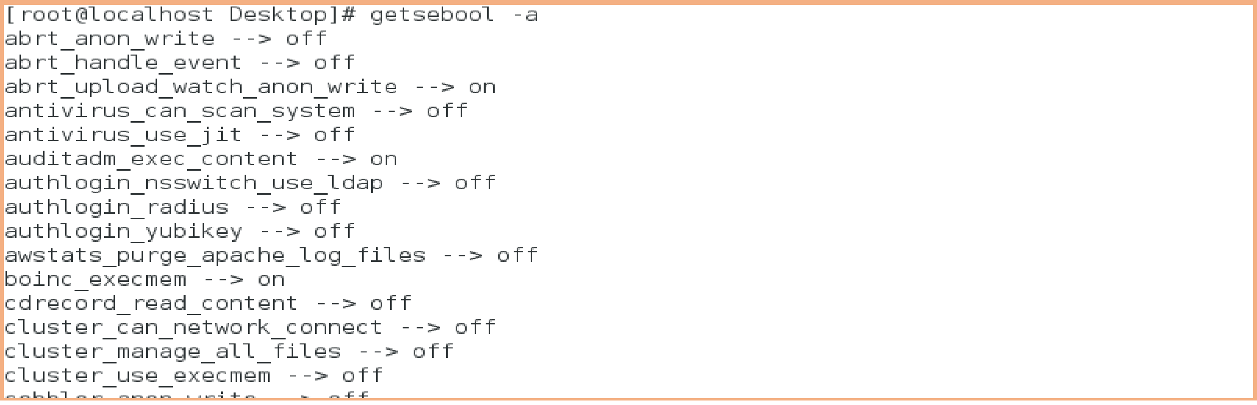
```
[root@localhost /]# semanage fcontext -a -t httpd_sys_content_t /ll
[root@localhost /]# restorecon -R -v /ll
restorecon reset /ll context unconfined_u:object_r:default_t:s0->unconfined_u:ob
ject_r:httpd_sys_content_t:s0
[root@localhost /]# ls -Zd /ll
drwxr-xr-x. root root unconfined_u:object_r:httpd_sys_content_t:s0 /ll
[root@localhost /]# 
```

restorecon this command use for relabeling of SElinux context on file and folder. If you want to set context you have to run this command so system can relabel that context what you want to set on particular file and folder. And it run after semanage command, so system can permanently change SElinux context.

SElinux Booleans

SElinux Booleans are switches that change the behavior of the SElinux policy. SElinux Booleans are rules that can be enabled or disabled. They can be used by security administrators to tune the policy to make selective adjustments.

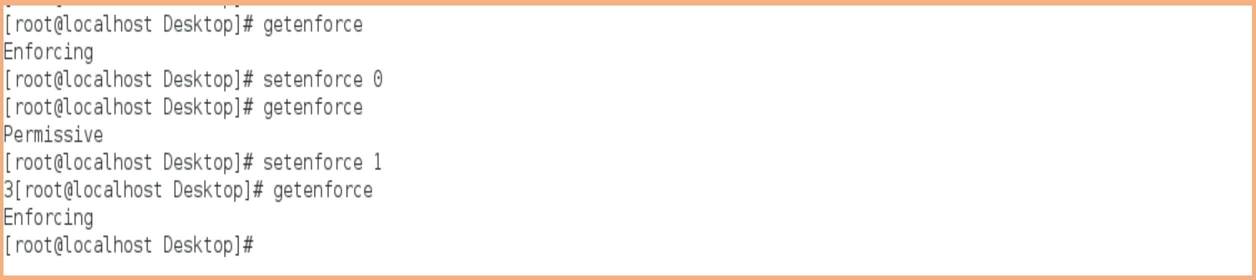
The getsebool command is used to display SElinux Booleans and their current value. The -a option causes this command to list all of the Booleans.



```
[root@localhost Desktop]# getsebool -a
abrt_anon_write --> off
abrt_handle_event --> off
abrt_upload_watch_anon_write --> on
antivirus_can_scan_system --> off
antivirus_use_jit --> off
auditadm_exec_content --> on
authlogin_nsswitch_use_ldap --> off
authlogin_radius --> off
authlogin_yubikey --> off
awstats_purge_apache_log_files --> off
boinc_execmem --> on
cdrecord_read_content --> off
cluster_can_network_connect --> off
cluster_manage_all_files --> off
cluster_use_execmem --> off
cobbler_anon_write --> off
```

Changing SElinux Modes

The setenforce command modifies the current SElinux mode:

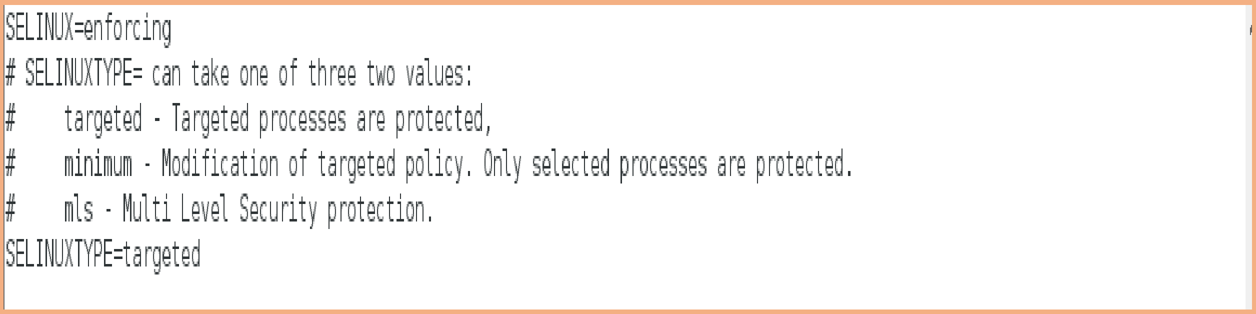


```
[root@localhost Desktop]# getenforce
Enforcing
[root@localhost Desktop]# setenforce 0
[root@localhost Desktop]# getenforce
Permissive
[root@localhost Desktop]# setenforce 1
3[root@localhost Desktop]# getenforce
Enforcing
[root@localhost Desktop]#
```

Setting the default SElinux mode

The configuration file that determines what the SElinux mode is at boot time is

/etc/selinux/config. Notice that it contains some useful comments:



```
SELINUX=enforcing
# SELINUXTYPE= can take one of three two values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Note: SElinux Troubleshooter is best utility for SElinux Troubleshooting.

So, please use SElinux Troubleshooter so you can easily manage SElinux.

Location: Application → sundry → SElinux Troubleshooter

TigerVNC

TigerVNC (Tiger Virtual Network Computing) is a system for graphical desktop sharing which allows you to remotely control other computers.

TigerVNC works on the client-server principle: a server shares its output (vncserver) and a client (vncviewer) connects to the server.

vncserver is a utility which starts a VNC (Virtual Network Computing) desktop. It runs Xvnc with appropriate options and starts a window manager on the VNC desktop. vncserver allows users to run separate sessions in parallel on a machine which can then be accessed by any number of clients from anywhere

TigerVnc configuration

1. On Server side

   - First install tigervnc-server package

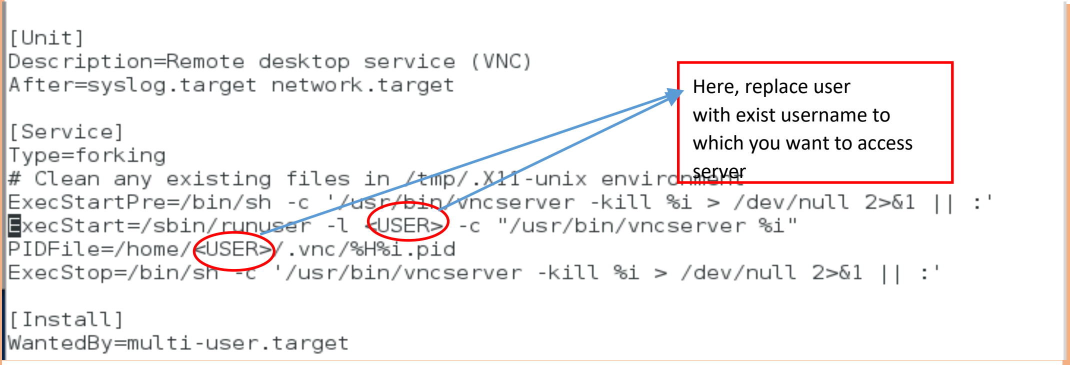
     Ex:# yumØinstallØtigervnc-server

   - Now copy vnc configuration from "/usr/lib/systemd/system/vncserver@.service" to "/etc/systemd/system/vncserver@.service"

     Ex:# cp /usr/lib/systemd/system/vncserver@.service Ø/etc/systemd/system/vncserver@.service

   - Now edit user name in vnc configuration file /etc/systemd/system/vncserver@.service

     Ex:#viØ/etc/systemd/system/vncserver@:1.service



```
[Unit]
Description=Remote desktop service (VNC)
After=syslog.target network.target

[Service]
Type=forking
# Clean any existing files in /tmp/.X11-unix environment
ExecStartPre=/bin/sh -c '/usr/bin/vncserver -kill %i > /dev/null 2>&1 || :'
ExecStart=/sbin/runuser -l <USER> -c "/usr/bin/vncserver %i"
PIDFile=/home/<USER>/.vnc/%H%i.pid
ExecStop=/bin/sh -c '/usr/bin/vncserver -kill %i > /dev/null 2>&1 || :'

[Install]
WantedBy=multi-user.target
```

Here, replace user with exist username to which you want to access server

And save the setting

- To make the changes take effect immediately, issue the following command:

```
        # systemctl daemon-reload
```

- Set the password for the user or users defined in the configuration file. Note that you need to switch from root to USER first.

```
        # suØ - username (exist user)
        $ vncpasswd
        Password:
        Verify:
```

- Now start and enable tigervnc service

```
#systemctlØstartØvncserver@:1.service
# systemctlØenableØvncserver@:1.service
```

- Now enable tigervnc on firewall

```
#firewall-cmdØ--permanentØ--add-service=vncserver
#firewall-cmdØ--reload
```

2. Client side

- Install vnc viewer package

```
#yumØinstallØtigervnc
```

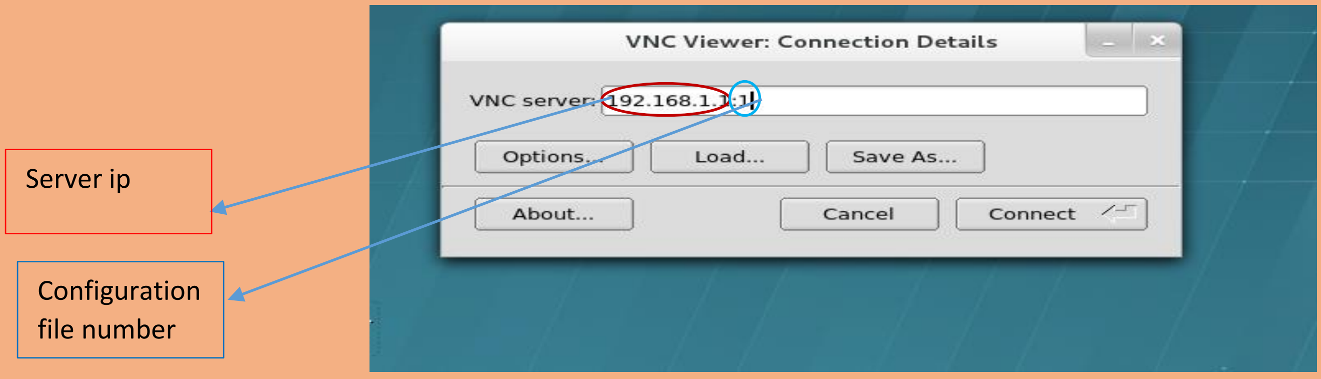
- Connecting to a VNC Server Using the CLI

  Enter the viewer command with the address and display number as arguments

```
#vncviewerØserveraddress:display_number
```

- Connecting to a VNC Server Using the GUI Tool

Open vncviewer tool: Application ⟶ Internet ⟶ Tigervnc viewer



VNC Viewer: Connection Details

VNC server: 192.168.1.1:1

Server ip

Configuration file number

Additional Resources

For more information about TigerVNC, see the resources listed below.

Installed Documentation

vncserver(1) — The manual page for the VNC server utility.

vncviewer(1) — The manual page for the VNC viewer.

vncpasswd(1) — The manual page for the VNC password command.

Xvnc(1) — The manual page for the Xvnc server configuration options.

x0vncserver(1) — The manual page for the TigerVNC server for sharing existing X servers.