T. Y. B.Sc. I.T.
Semester V

LINUX ADMINISTRATION

TEACHER's REFERENCE MANUAL FOR PRACTICALS

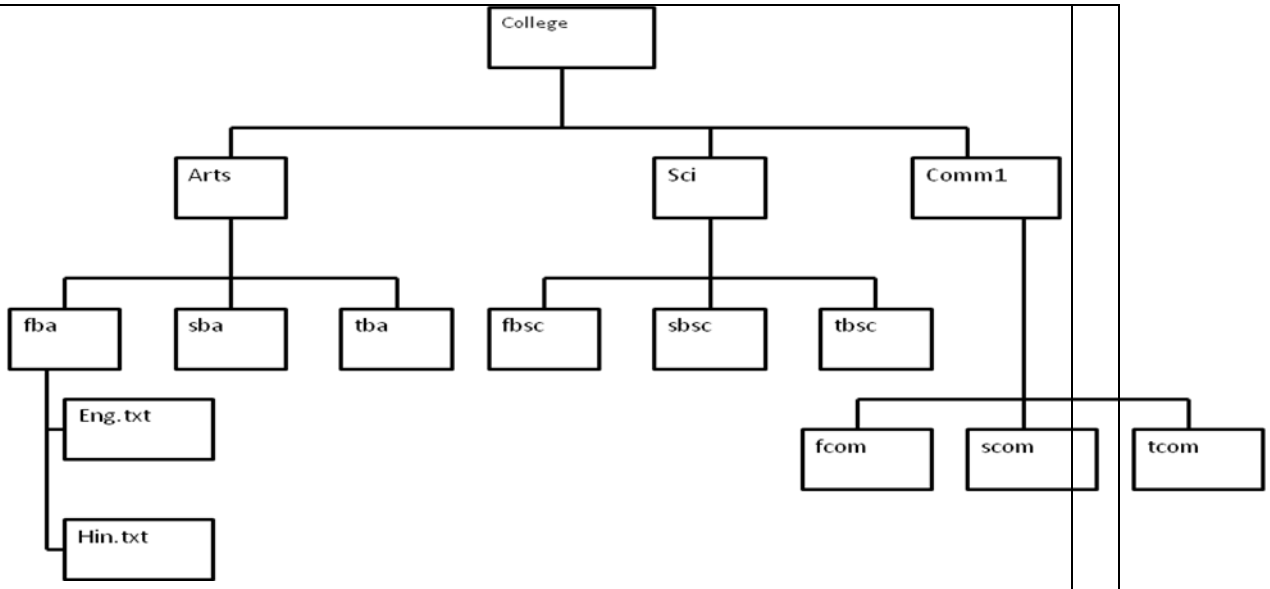
2013 – 2014

Prepared by Prof. Kiran Gurbani

1. INDEX OF Linux Administration Practicals

Table of Contents

2.	**Practical 1:** **Introduction to Basic Linux Commands & Editors**	
	Contents to be taught :- **1) Introduction to basic Linux commands to create a file, directory, copying a file, changing to a directory, moving, removing a file and directory, coming out of a directory, displaying present working directory(cat cmd with Ip/Op redirections, mkdir, cp, mv, rm, cd, rmdir, cd .., pwd)** **2) Introduction to ls (Listing) and ls – l (Long Listing) command and chmod (change mode of permissions) command with binary and symbolic masking.** **3) Introduction to pipe command and grep command.** **4) Use of vi editor with its modes.** **File Structure Example** **1) Creation of Tree structure with respective directories & Subdirectories with files. File creation can be taught with cat>filename command as well as through vi editor.** **2) Combining the files into new file & copying files from one location to another.** **3) Removing files & directories with the help of rm and rmdir Command.** **4) Changing permissions of files through chmod command.** **5) Displaying or filtering the files according to the requirement by using grep command & combining the commands through pipe command.**	



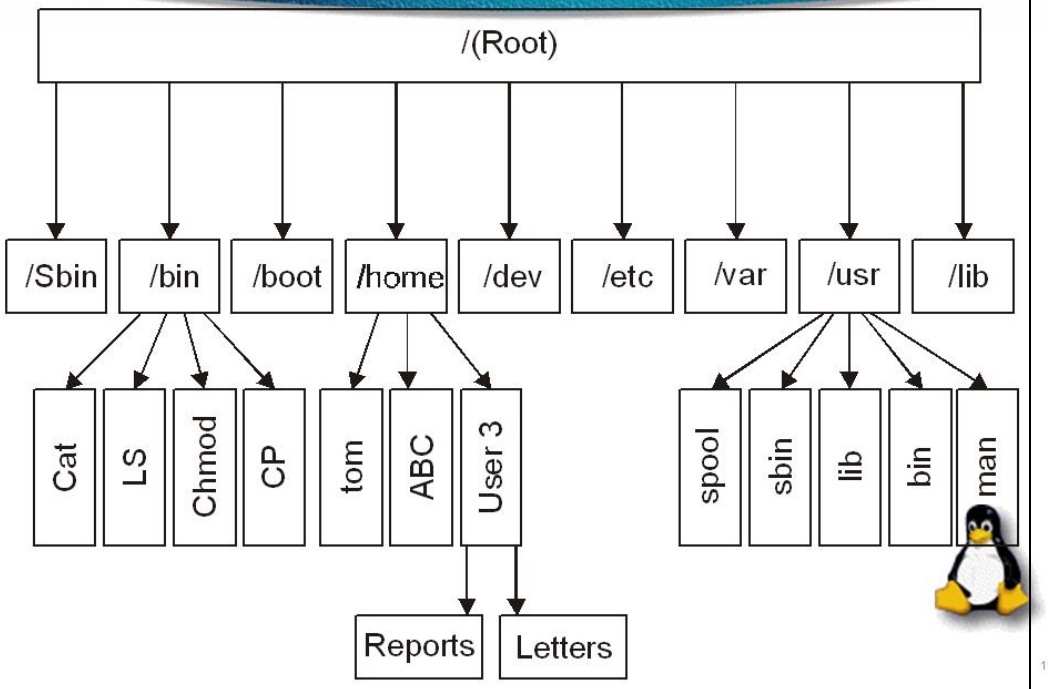
Example :

- /home/student]$ mkdir kiran
- student]$ cd kiran
- usr]$ mkdir college
- usr]$ cd college
- college]$ mkdir arts sci comm
- college]$ cd arts
- arts]$ mkdir fba sba tba
- arts]$ cd fba
- fba]$ cat>eng.txt
- Hello, welcome to eng
- ctrl+d
- fba]$ cat>hin.txt
- Hello welcome to hin
- ctrl+d
- fba]$ ls
- eng.txt hin.txt
- fba]$ cat eng.txt hin.txt>fba.txt
- fba]$ ls
- eng.txt hin.txt fba.txt
- fba]$ pwd
- /home/student/usr/college/arts/fba
- fba]$ cp fba.txt /home/student/usr/college/arts
- fba]$ cd ..
- arts]$ cd sba
- sba]$ cat>s1.txt
- sba]$ cat>s2.txt
- sba]$ cat s1.txt s2.txt>sba.txt
- sba]$ cp sba.txt /home/student/usr/college/arts
    - sba]$ cd ..
- arts]$ cd tba

- tba]$cat
- cat
- tba]$ cat t1.txt t2.txt>tba.txt
- tba]$ cp tba.txt arts
- tba]$ cd ..
- arts]$ ls
- fba.txt sba.txt tba.txt
- arts]$ cat fba.txt sba.txt tba.txt>arts.txt
- arts]$ cp arts.txt /home/student/usr/college
- arts]$ cd ..
- college]$ cd sci
- college]$ tree
- remove a file
-]$ rm abc.txt
- Rm abc.txt xyz.txt
- Wild card characters
- , ?
- multiple unknown characters
- ? single unknown character
- Rm *.txt delete all txt files
- Rm a*.txt
- Rm ab?c.doc
- Rm *ing.txt
- Rmdir (Romove a directory)
- Rules to be followed for removing a directory
- directory should be empty
- out of that directory
- rmdir fba

1) File System Introduction with GUI :

Filesystem



11-8

File Directory Tree Structure

	Directory	Function
(1)	Root '/ '	It begin file system structure call the root, The root of the virtual directory.
(2)	/home	Contains users home directory.
(3)	/bin	The binary directory holds all standard commands and utility programs (like Vi editor), also executable files for the commands.
(4)	/usr	Holds those files & commands used by the system. This directory breaks down into several sub directories. The user-installed software directory.
(5)	/usr/sbin	Holds system administration commands with its executable files.
(6)	/usr/lib	It holds libraries for programming language
(7)	/usr/doc	Holds linux documentations.
(8)	/usr/man	Holds online manual man files i.e. Help files.
(9)	/usr/spool	Holds spool files such as those generated for printing jobs, waiting printing jobs will be stored in spool directory.
(10)	/sbin	Holds system administration command for booting of system. The system binary directory, where many GNU admin-level utilities are stored.

11-9

Filesystem Contd….

(11)	/var	It holds the file that vary. Hence this directory has information about different utilities of linux e.g. /val/log. – directory contains files that stores operating system log files or error reports file. The variable directory, for files that change frequently, such as log files.
(12)	/dev	Holds file interfaces for devices such as terminal and printers. It stores the device files for input and output hardware devices.
(13)	/etc	Holds system configuration files and any other system files. This stores operating system related data which users and operating system needs to refer such as password file.
(14)	/tmp	The temporary directory, where temporary work files can be created and destroyed.
(15)	/mnt	The mount directory, another common place for mount points used for removable media.
(16)	/Lib	The library directory, where system and application library files are stored.
(17)	/boot	The boot directory, where boot files are stored.

2) Change Mode of File Permissions(Chmod):

Decoding File Permissions:

You can decode the cryptic file permissions you've seen when using the ls command. Here we will specify how to decipher the permissions.

Using File Permission Symbols:

The ls command displays the file permissions for files, directories, and devices on the Linux system:

```
$ ls -l
total 68
-rw-rw-r-- 1 rich rich 50 2007-09-13 07:49 file1.gz
-rwxrwxr-x 1 rich rich 4882 2007-09-18 13:58 myprog
drwxrwxr-x 2 rich rich 4096 2007-09-03 15:12 test1
$
```

The first field in the output listing is a code that describes the permissions for the files and directories.

The first character in the field defines the type of the object, These are the different options of file types.

- - for files d for directories l for links
- c for character devices b for block devices n for network devices

After that, there are three sets of three characters. Each set of three characters defines an access permission triplet:

- r for read permission for the object
- w for write permission for the object
- x for execute permission for the object

If a permission is denied, a dash appears in the location. The three sets relate the three levels of security for the object:

- The owner of the object
- The group that owns the object
- Everyone else on the system

-rwxrwxr-x 1 rich rich 4882 2007-09-18 13:58 myprog

The **chmod** command is used for changing the permissions of a file.

Permissions can be changed by two ways:

Binary Masking Method (Absolute Permission Method/):

In this method binary 1 or 0 is assigned to the permissions

1 → Assigning (Granting permission)

0 → Removing (Revoking permission)

Syntax for Binary Masking:

Chmod	Binary Number	filename
Owner	Group	Other
111	101	001
7	5	1

Chmod 751 hello.txt

7 indicate (111) Owner has all three permissions (Read, Write, Execute)

5 indicate (101) Group user has Read & Execute Permission.

1 indicate (001) Other or Guest User has Only Execute Permission.

2) Symbolic Masking Method (Relative Method of changing permissions):

In this method symbols (Abbreviations) are used for assigning & Removing Permissions:

r → Read, w → Write, x → Execute, g → group, o → Others

u → Owner, a → All users, +→ Assinging a permission (plus sign)

- → Removing a permission. (minus sign)

Syntax:

Example:

Chmod g+rw hello.txt

Chmod a-wx xyz.txt

3) Vi Editor:

Vi stands for visual editor. It is the one of the most widely used editors in Linux & used to create & edit text files. It is a screen oriented editor that is extremely fast when scrolling through large documents. It does not support any document formatting like bold/Italic, spell checking or any views of a document as it will look when printed.

Also Vi Emacs are the keyboard editors which will be used for keyboard for two different operations.

(1) To specify editing commands.

(2) Receive character input.

Alphabetic keys are reserve for characters. I/P keys specify editing commands & Esc & Enter keys are editing commands.

To invoke Vi editor use may use one of the following commands.

$ vi

$ vi <filename>

If vi <filename> is typed at command line & given filename is already exists, it opens for editing.

$ vi

It opens a editor creates a new file with noname. You can edit, type the command & then can save it with Esc, **:** , w, filename.

- i.e. esc: w filename & save the unknown file with valid filename.
  - esc: Q will quit from Vi editor
  - esc: WQ ABC.txt ↔ save & quit.

Modes of operation in vi editor:

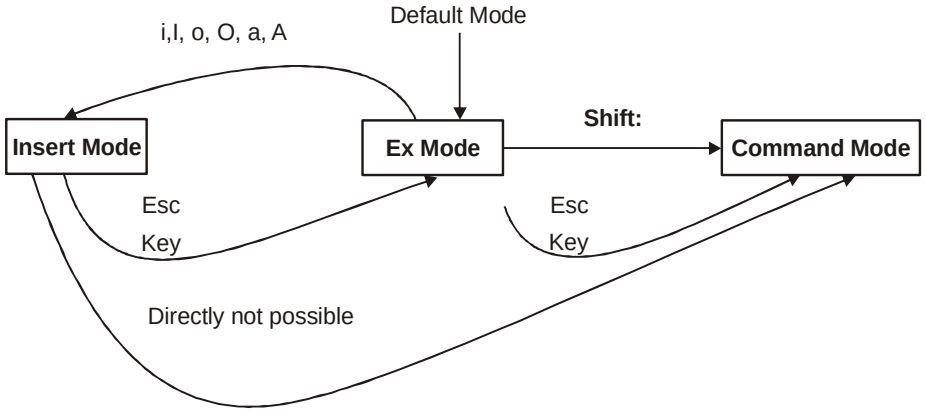
In vi editor three types of modes are available.

(1) Command mode (normal mode): The Default mode of the editor where every key pressed is interpreted as a command to run on text. You can copy & delete the text, move between lines, move between words, move between characters, delete a word, delete a line, copy line, find, search word.

In vi editor the**.** **(dot**) key is used for repeating last command. The principle is use the actual command only once and then repeat at other places with the dot command. To take a simple e.g. if u have deleted 2 lines of text with 2dd, then to repeat this operation elsewhere, all u have to do is to position the cursor at the desired location and press '.' This will repeat the last editing instruction, i.e. it will also delete two lines of text.

(2) Input mode (Insert Mode): This mode is invoked by pressing Key I or a or o. In this mode the text can be edited or inserted or modified.

Ex mode (Last Line Mode): This mode used to handle files (like saving) and perform substitution. Pressing a: in the Command Mode invokes this mode.



pattern can be searched with a / and ?

/ pattern searches forward

? pattern searches backward

3.	**Practical 2:** **Installation of Red Hat Linux Operating System.**	

1. First Screen to install, Press enter key to install GUI of Linux.
2. Make Last drive of windows as Free space for Linux Installation
3. Make Bootable sequence first to CD-Drive & then insert Bootable CD.

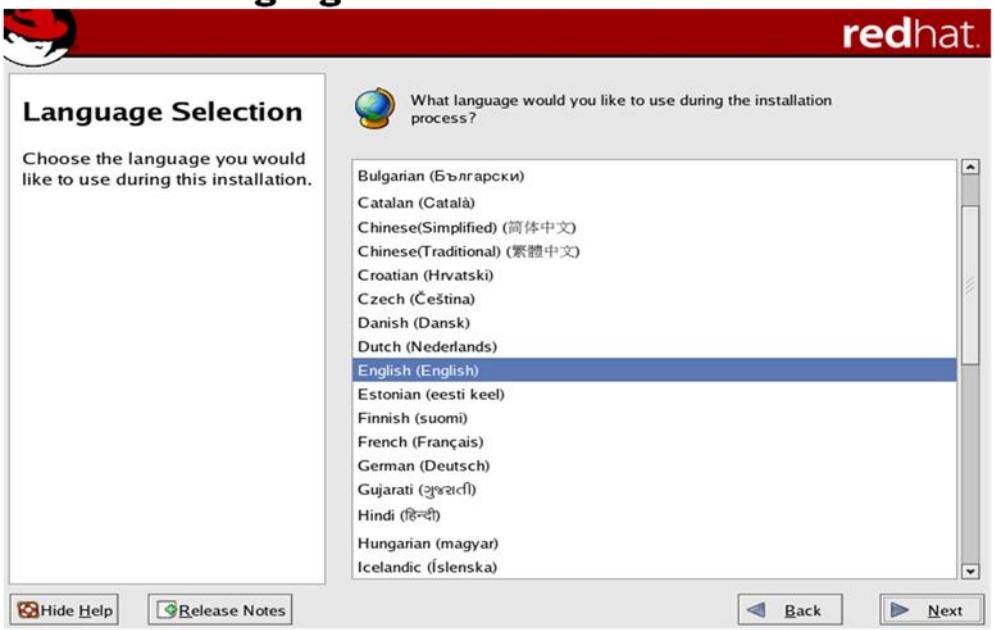
1. Select Install or upgrade existing system options.

➔ **Select Install o Upgrade Option**

1. Display of Welcome Screen

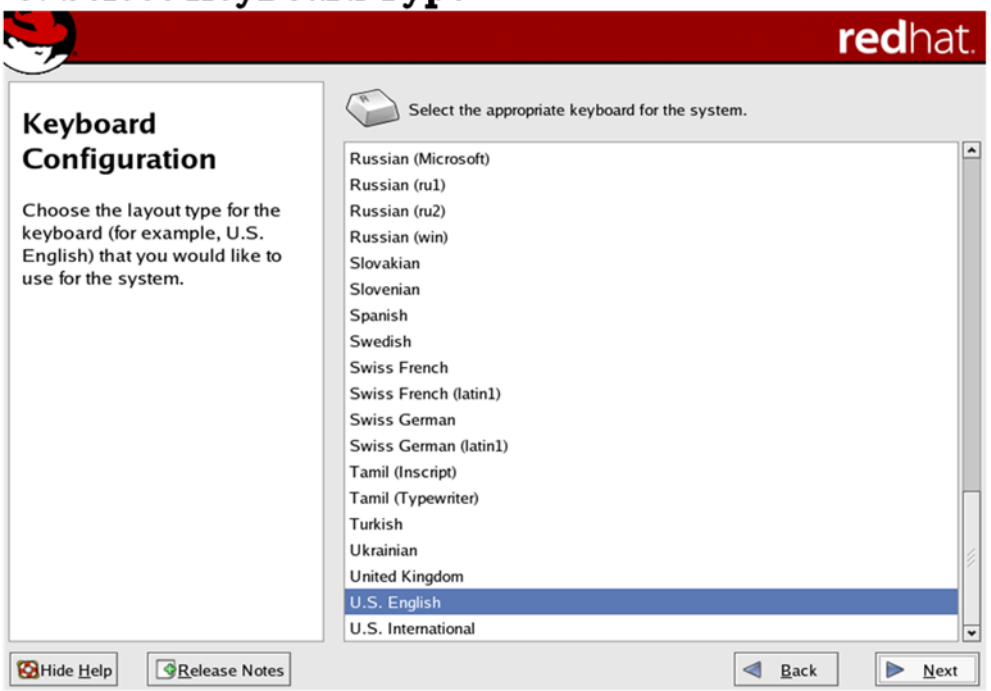
➔ **Click on Next Button**

2. Select Language



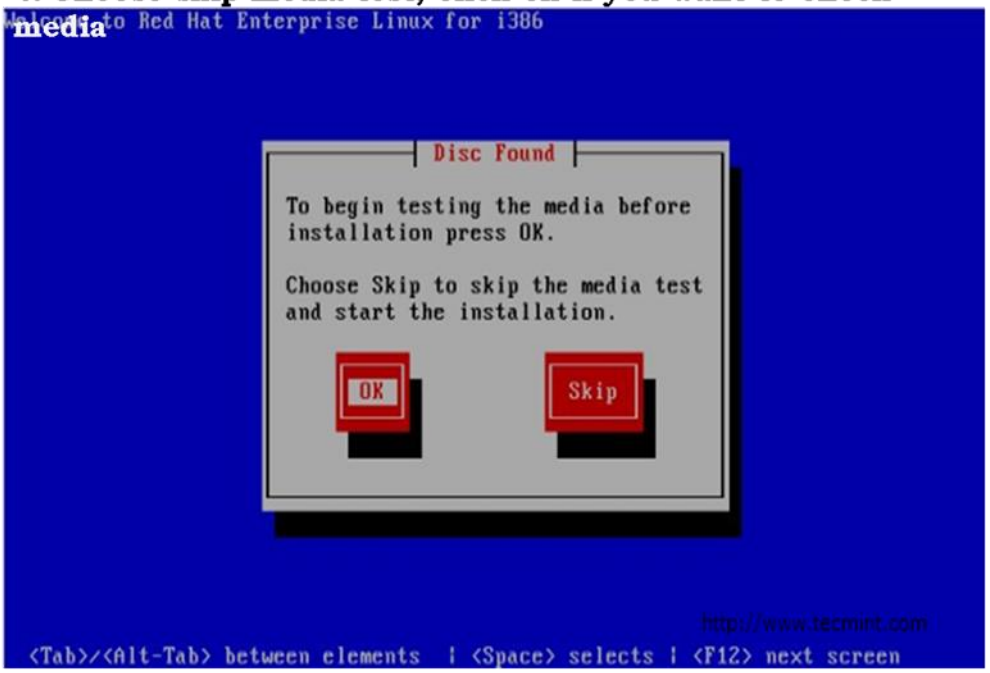
➔ **Select RHEL 6 language (English)**

3. Select KeyBoard Type



➔ **Select RHEL 6 Keyboard Lang. (U.S. English)**

4. Choose skip media test, click ok if you want to check media

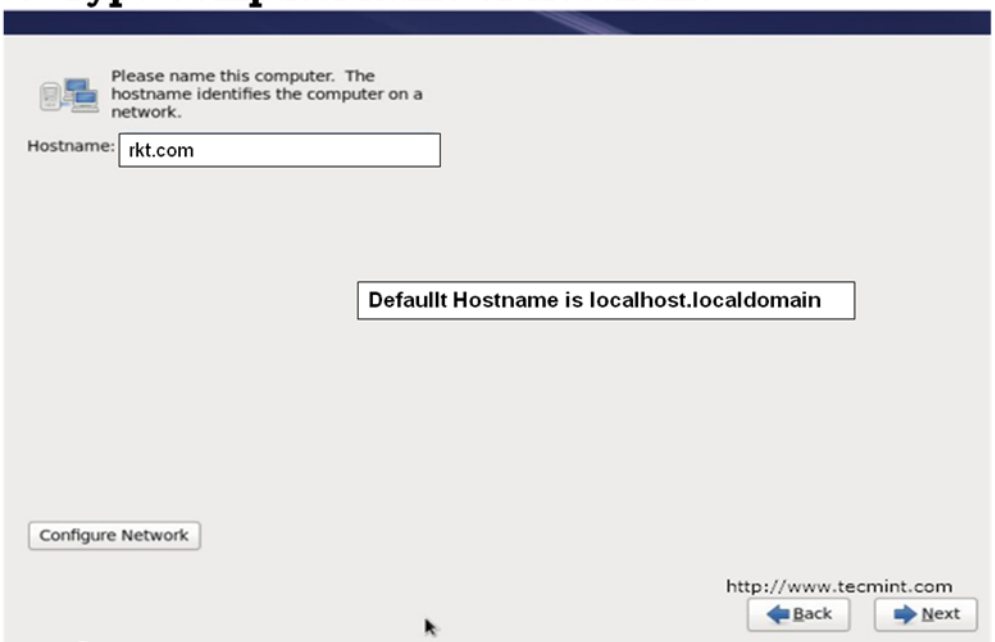


➔ Skip RHEL 6 media test

5. Select storage device

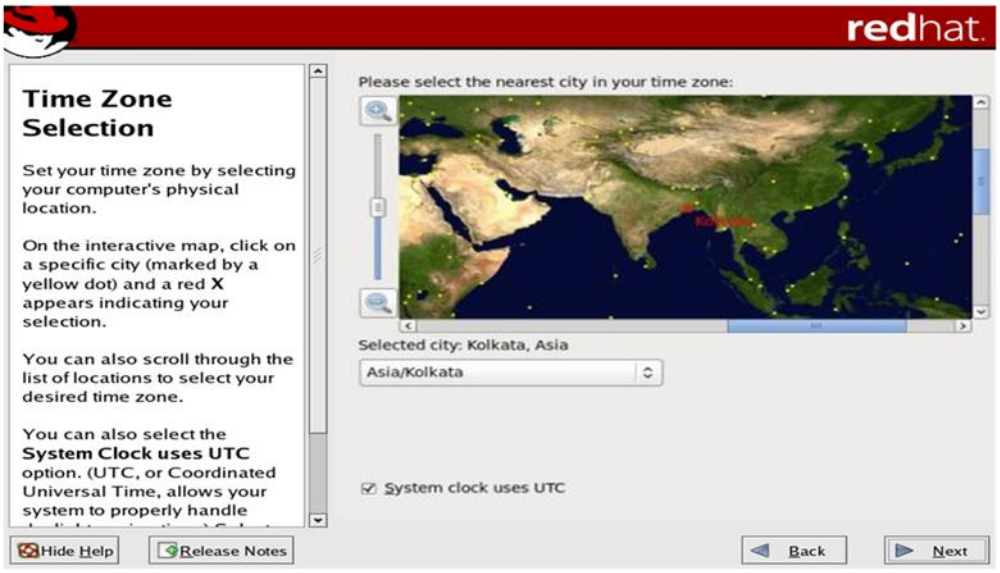
➔ Select RHEL 6 Basic Storage device

6. Type Computer Name or Hostname



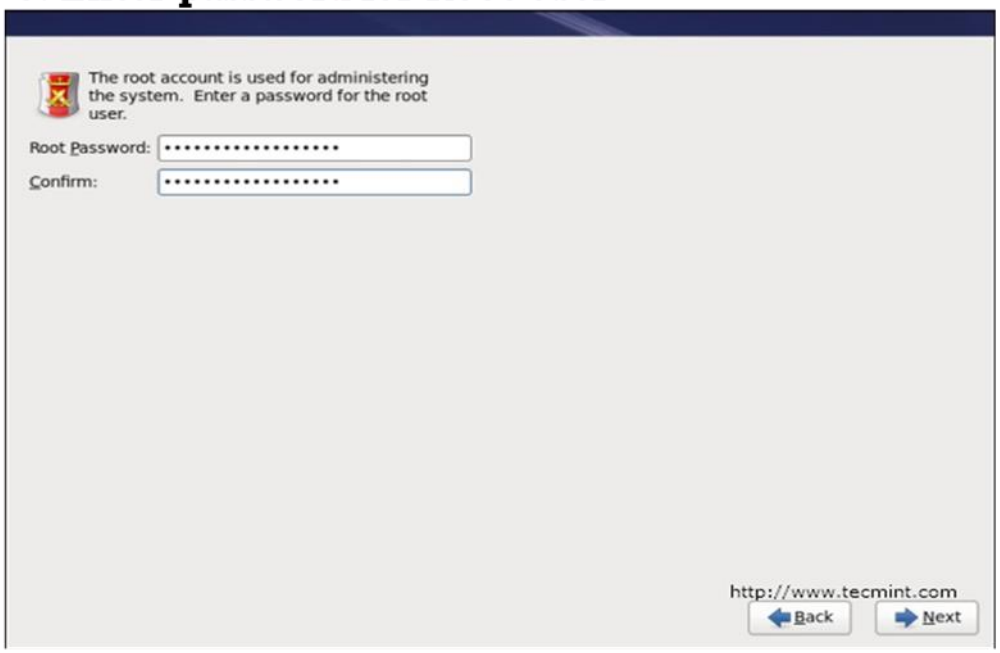
➔ **Set RHEL 6 Hostname to rkt.com**

7. Select Time Zone Location



➔ **Select RHEL 6 TimeZone as Asia/Kolkata**

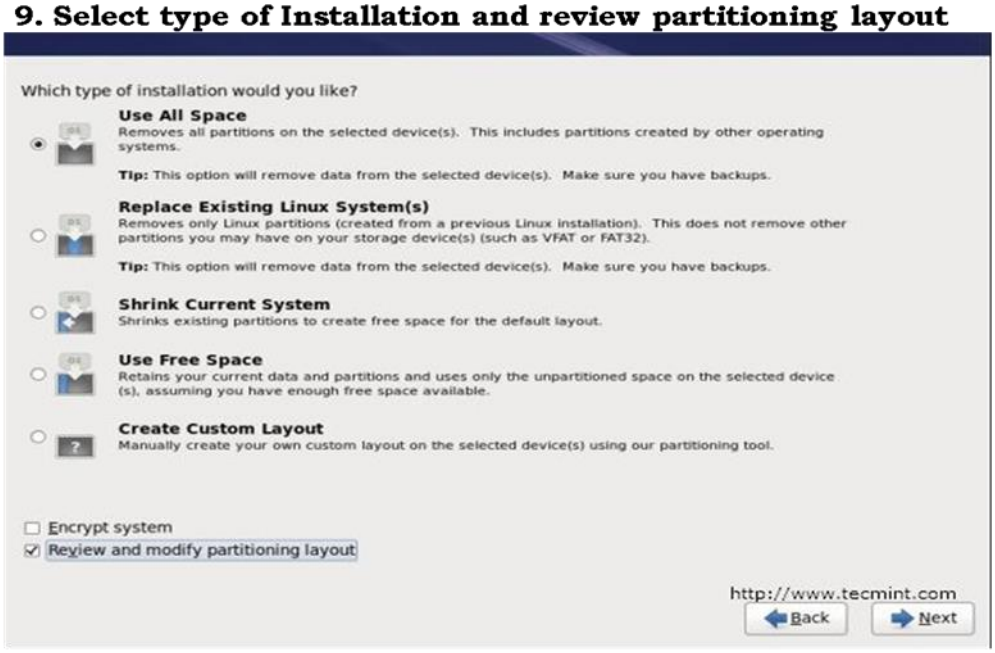
8. Enter password for Root User



➔ **Set RHEL 6 Root Password**

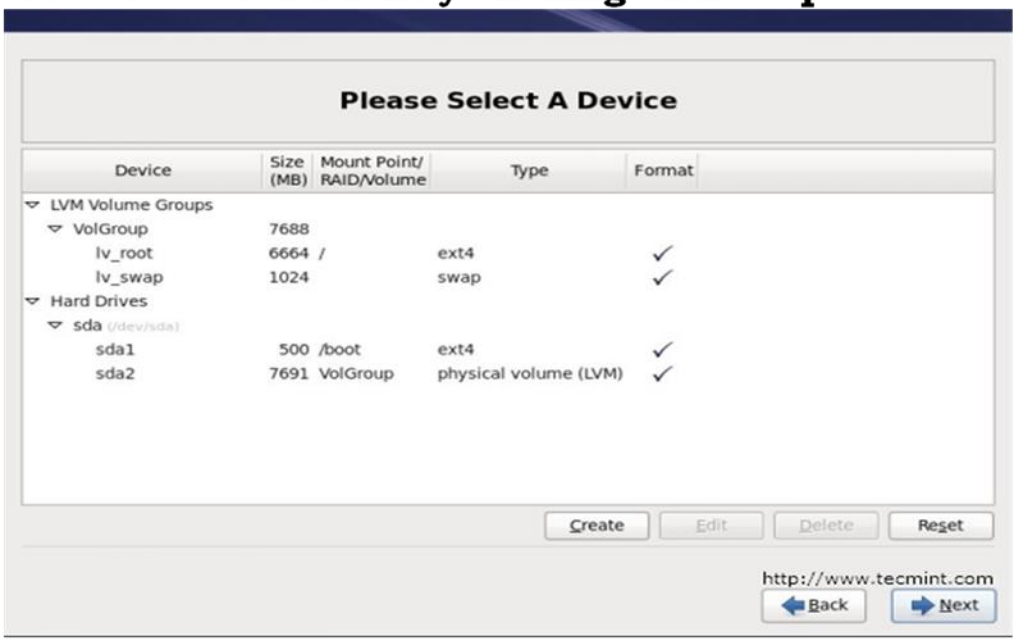
Selection of a standard installation type is now available.
The options include Personal Desktop, Workstation, Server, or Custom.
For this particular guide, I selected Personal Desktop and then Next.

9. Select type of Installation and review partitioning layout



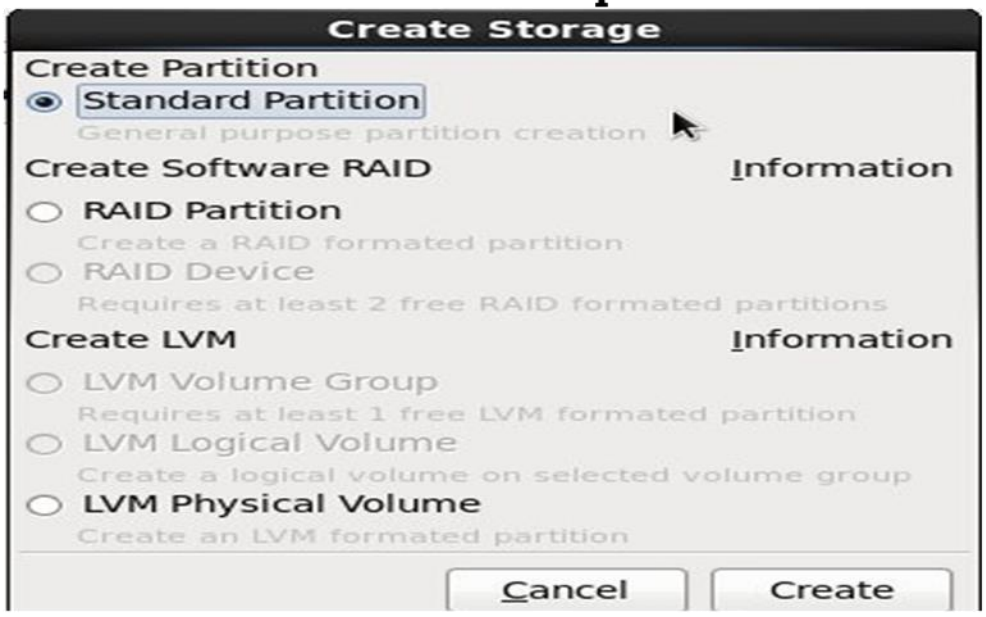
➔ **Set RHEL 6 Partition Layout as Create Custom Layout**

10. Create Partition by clicking on free space

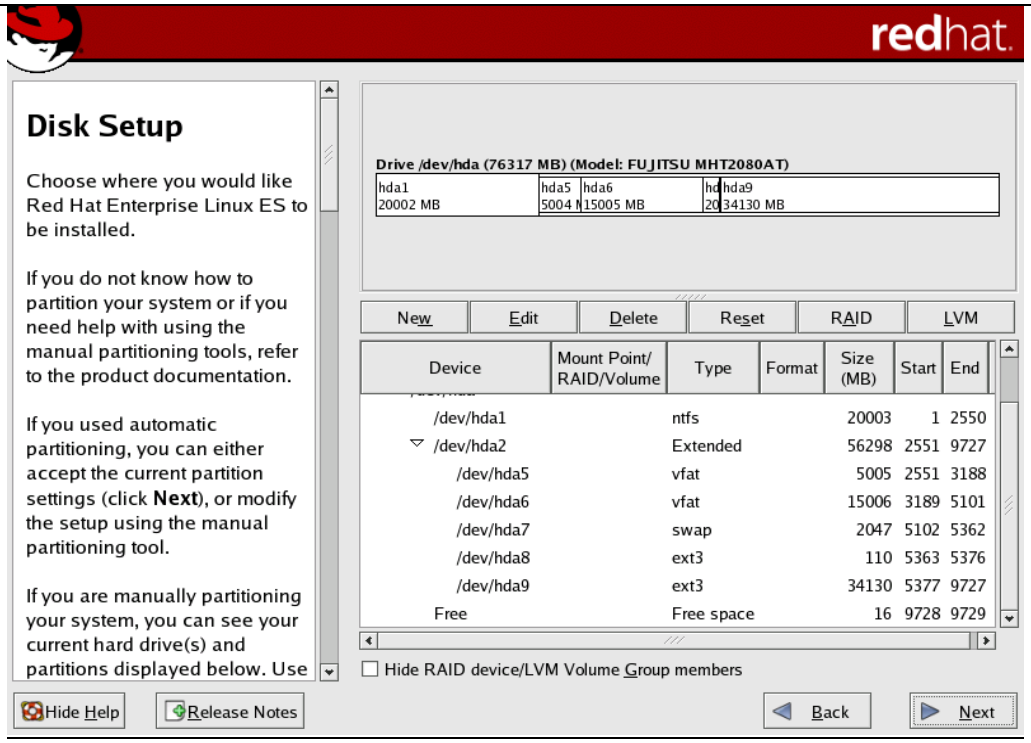


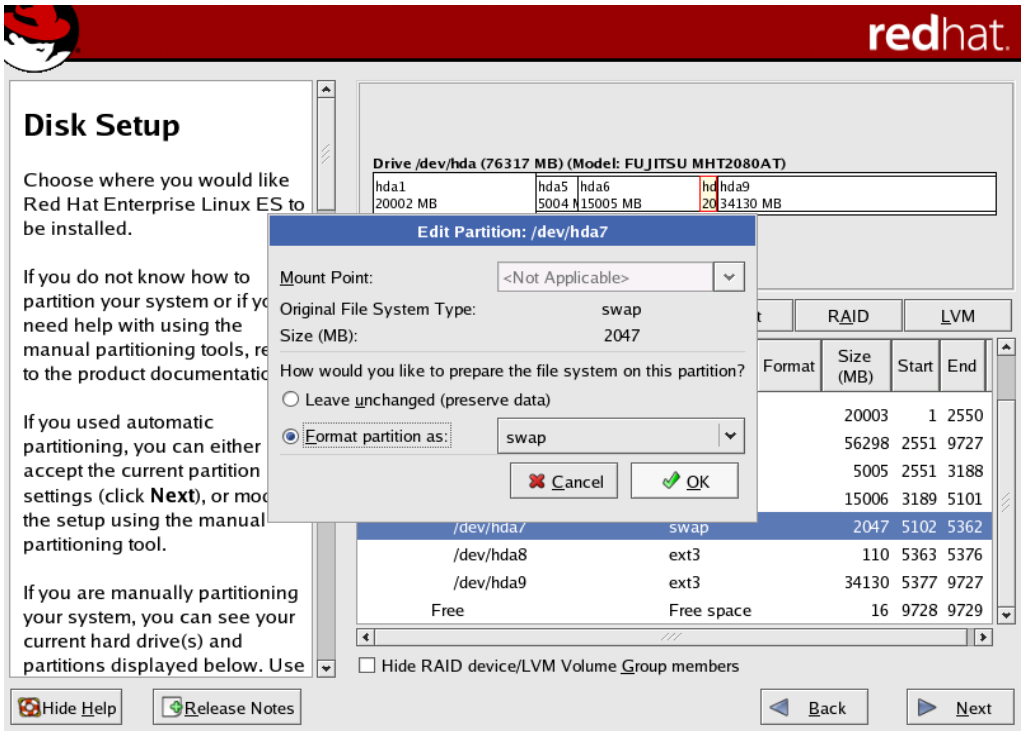
➔ Click on Create Button

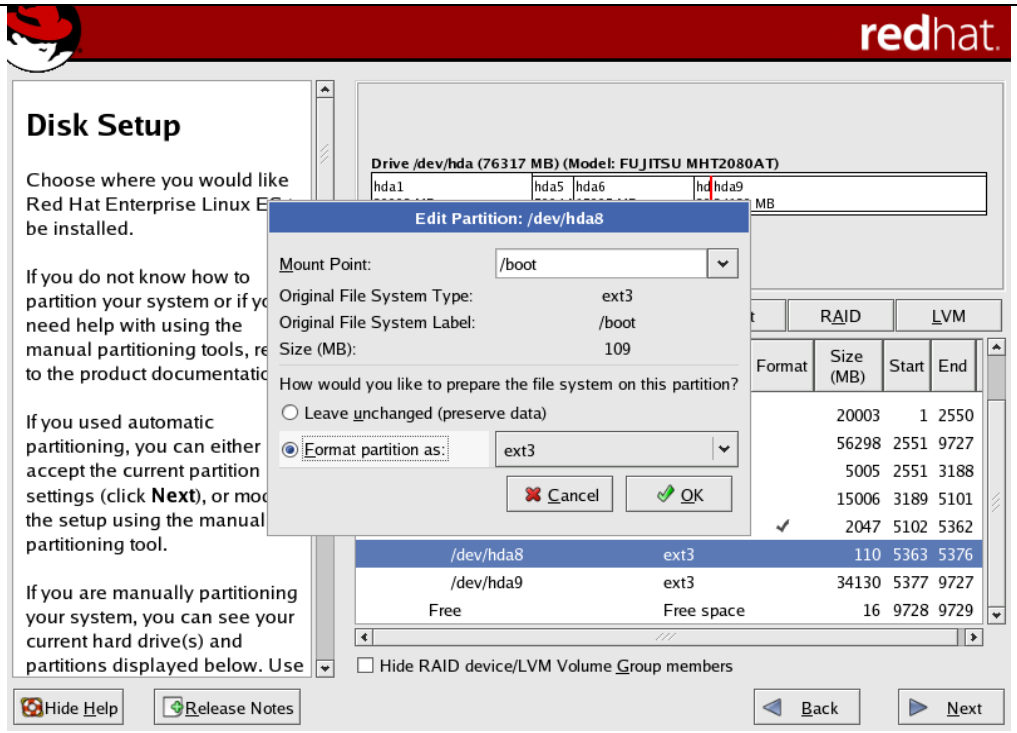
11. Select Standard Partition option

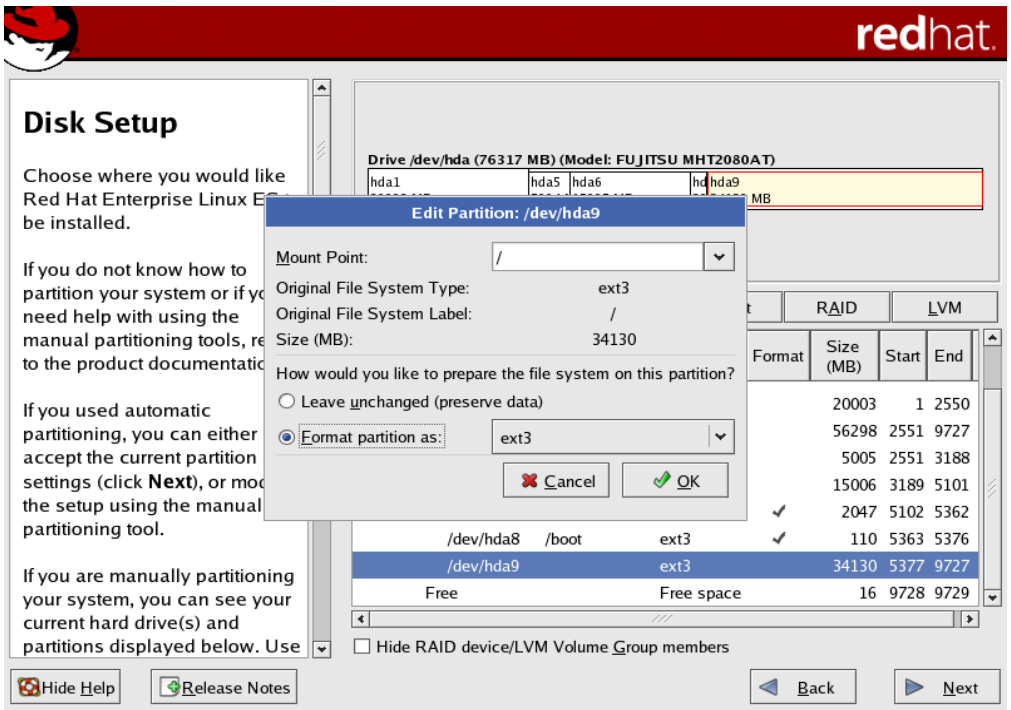


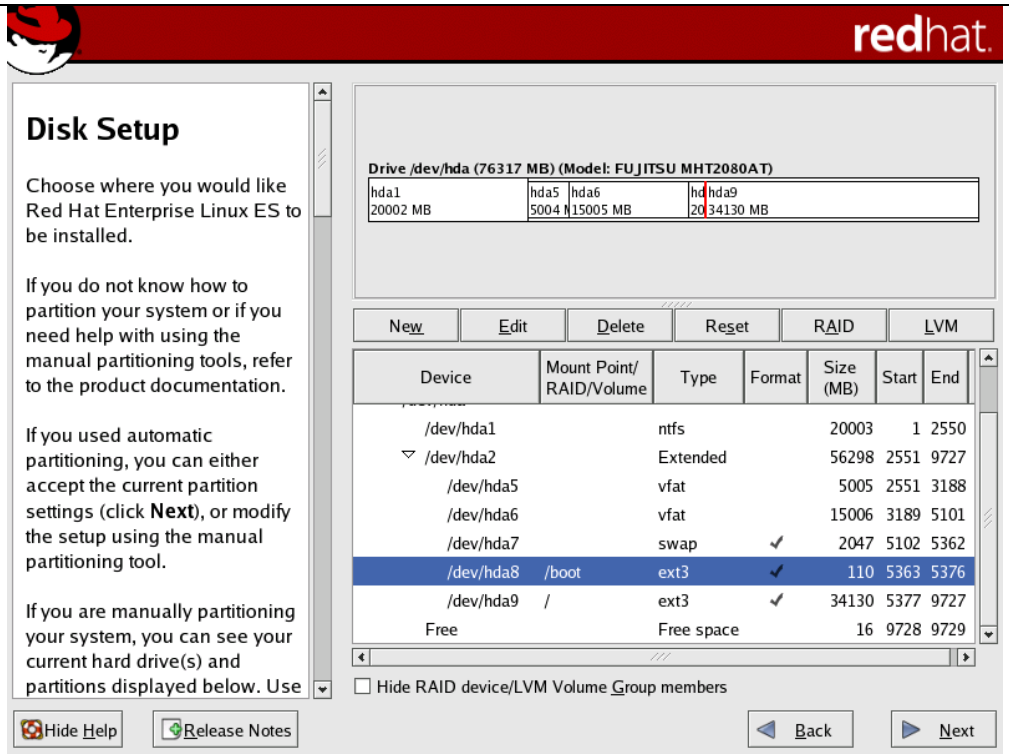
➔ Click on Create Button

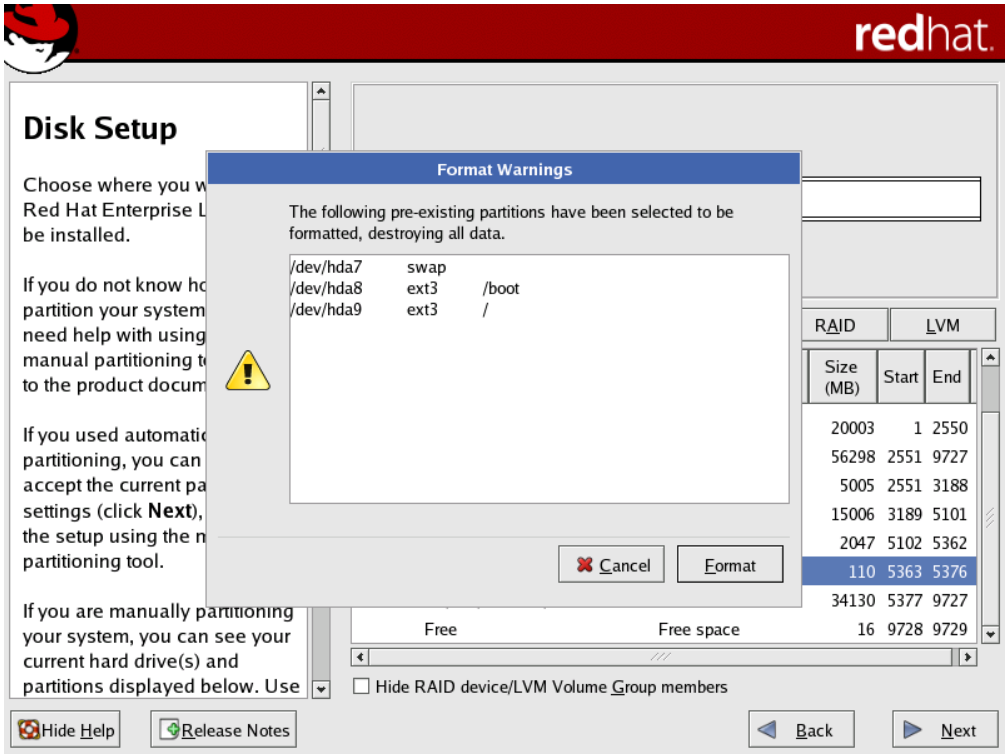

redhat.
Disk Setup
Choose where you would like Red Hat Enterprise Linux ES to be installed.
If you do not know how to partition your system or if you need help with using the manual partitioning tools, refer to the product documentation.
If you used automatic partitioning, you can either accept the current partition settings (click Next), or modify the setup using the manual partitioning tool.
If you are manually partitioning your system, you can see your current hard drive(s) and partitions displayed below. Use
Drive /dev/hda (76317 MB) (Model: FUJITSU MHT2080AT)
hda1 20002 MB
hda5 5004 M
hda6 15005 MB
hda9 34130 MB
New
Edit
Delete
Reset
RAID
LVM
Device
Mount Point/ RAID/Volume
Type
Format
Size (MB)
Start
End
/dev/hda1 ntfs 20003 1 2550
/dev/hda2 Extended 56298 2551 9727
/dev/hda5 vfat 5005 2551 3188
/dev/hda6 vfat 15006 3189 5101
/dev/hda7 swap 2047 5102 5362
/dev/hda8 ext3 110 5363 5376
/dev/hda9 ext3 34130 5377 9727
Free Free space 16 9728 9729
Hide RAID device/LVM Volume Group members
Hide Help
Release Notes
Back
Next


redhat.
Disk Setup
Choose where you would like Red Hat Enterprise Linux ES to be installed.
If you do not know how to partition your system or if yo need help with using the manual partitioning tools, re to the product documentatic
If you used automatic partitioning, you can either accept the current partition settings (click Next), or mod the setup using the manual partitioning tool.
If you are manually partitioning your system, you can see your current hard drive(s) and partitions displayed below. Use
Drive /dev/hda (76317 MB) (Model: FUJITSU MHT2080AT)
hda1 20002 MB
hda5 5004 M
hda6 15005 MB
hda9 34130 MB
Edit Partition: /dev/hda7
Mount Point: <Not Applicable>
Original File System Type: swap
Size (MB): 2047
How would you like to prepare the file system on this partition?
Leave unchanged (preserve data)
Format partition as: swap
Cancel
OK
RAID
LVM
Format
Size (MB)
Start
End
20003 1 2550
56298 2551 9727
5005 2551 3188
15006 3189 5101
/dev/hda7 swap 2047 5102 5362
/dev/hda8 ext3 110 5363 5376
/dev/hda9 ext3 34130 5377 9727
Free Free space 16 9728 9729
Hide RAID device/LVM Volume Group members
Hide Help
Release Notes
Back
Next

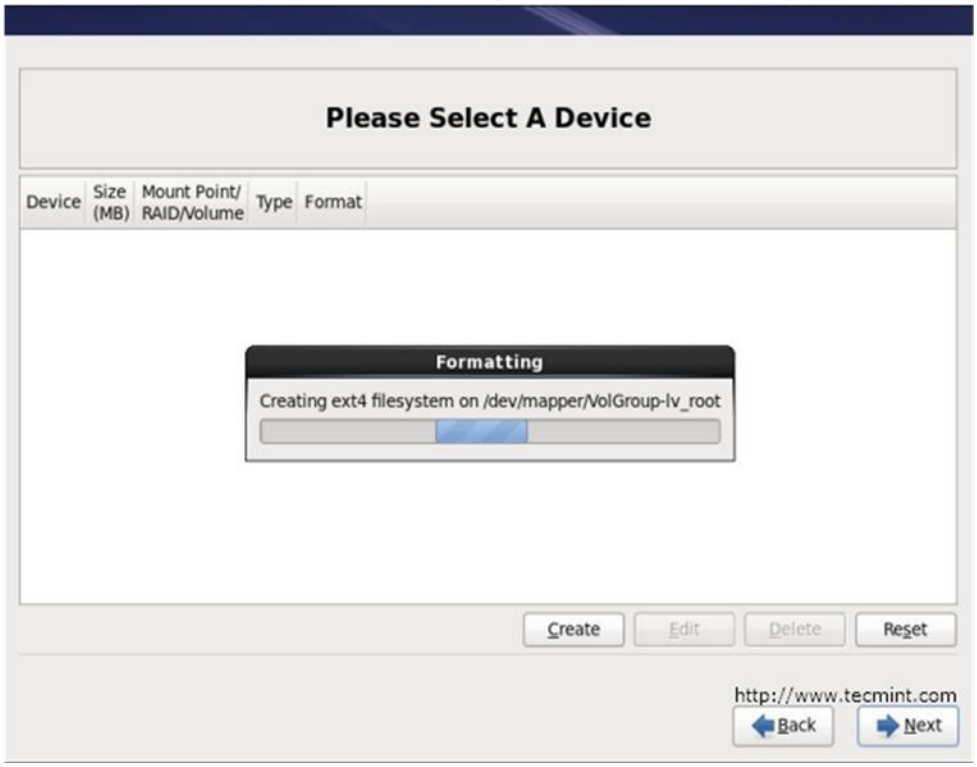

redhat.
Disk Setup
Choose where you would like Red Hat Enterprise Linux E be installed.
If you do not know how to partition your system or if y need help with using the manual partitioning tools, re to the product documentatio
If you used automatic partitioning, you can either accept the current partition settings (click Next), or mod the setup using the manual partitioning tool.
If you are manually partitioning your system, you can see your current hard drive(s) and partitions displayed below. Use
Drive /dev/hda (76317 MB) (Model: FUJITSU MHT2080AT)
hda1
hda5
hda6
hda9
MB
Edit Partition: /dev/hda8
Mount Point: /boot
Original File System Type: ext3
Original File System Label: /boot
Size (MB): 109
How would you like to prepare the file system on this partition?
Leave unchanged (preserve data)
Format partition as: ext3
Cancel
OK
RAID
LVM
Format
Size (MB)
Start
End
20003 1 2550
56298 2551 9727
5005 2551 3188
15006 3189 5101
2047 5102 5362
/dev/hda8 ext3 110 5363 5376
/dev/hda9 ext3 34130 5377 9727
Free Free space 16 9728 9729
Hide RAID device/LVM Volume Group members
Hide Help
Release Notes
Back
Next


redhat.
Disk Setup
Choose where you would like Red Hat Enterprise Linux E be installed.
If you do not know how to partition your system or if y need help with using the manual partitioning tools, re to the product documentatio
If you used automatic partitioning, you can either accept the current partition settings (click Next), or mod the setup using the manual partitioning tool.
If you are manually partitioning your system, you can see your current hard drive(s) and partitions displayed below. Use
Drive /dev/hda (76317 MB) (Model: FUJITSU MHT2080AT)
hda1
hda5
hda6
hda9
MB
Edit Partition: /dev/hda9
Mount Point: /
Original File System Type: ext3
Original File System Label: /
Size (MB): 34130
How would you like to prepare the file system on this partition?
Leave unchanged (preserve data)
Format partition as: ext3
Cancel
OK
RAID
LVM
Format
Size (MB)
Start
End
20003 1 2550
56298 2551 9727
5005 2551 3188
15006 3189 5101
2047 5102 5362
/dev/hda8 /boot ext3 110 5363 5376
/dev/hda9 ext3 34130 5377 9727
Free Free space 16 9728 9729
Hide RAID device/LVM Volume Group members
Hide Help
Release Notes
Back
Next

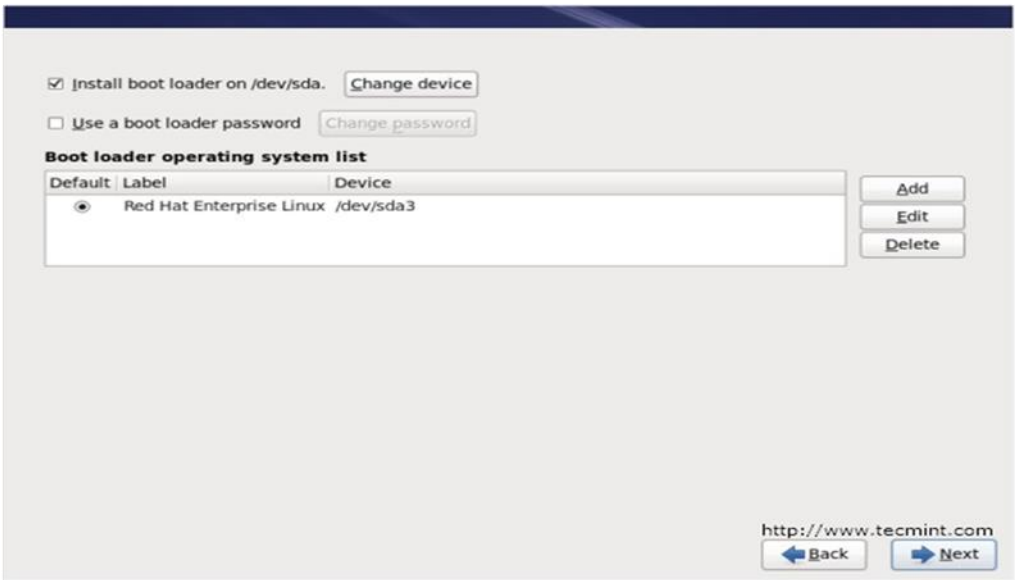

redhat
Disk Setup
Choose where you would like Red Hat Enterprise Linux ES to be installed.
If you do not know how to partition your system or if you need help with using the manual partitioning tools, refer to the product documentation.
If you used automatic partitioning, you can either accept the current partition settings (click Next), or modify the setup using the manual partitioning tool.
If you are manually partitioning your system, you can see your current hard drive(s) and partitions displayed below. Use
Drive /dev/hda (76317 MB) (Model: FUJITSU MHT2080AT)
hda1 20002 MB
hda5 5004 M
hda6 15005 MB
hda9 34130 MB
New
Edit
Delete
Reset
RAID
LVM
Device
Mount Point/ RAID/Volume
Type
Format
Size (MB)
Start
End
/dev/hda1 ntfs 20003 1 2550
/dev/hda2 Extended 56298 2551 9727
/dev/hda5 vfat 5005 2551 3188
/dev/hda6 vfat 15006 3189 5101
/dev/hda7 swap 2047 5102 5362
/dev/hda8 /boot ext3 110 5363 5376
/dev/hda9 / ext3 34130 5377 9727
Free Free space 16 9728 9729
Hide RAID device/LVM Volume Group members
Hide Help
Release Notes
Back
Next


redhat
Disk Setup
Format Warnings
The following pre-existing partitions have been selected to be formatted, destroying all data.
/dev/hda7 swap
/dev/hda8 ext3 /boot
/dev/hda9 ext3 /
Cancel
Format
RAID
LVM
Size (MB)
Start
End
Free Free space 16 9728 9729
Hide RAID device/LVM Volume Group members
Hide Help
Release Notes
Back
Next

12 . It will start formatting of Linux File System

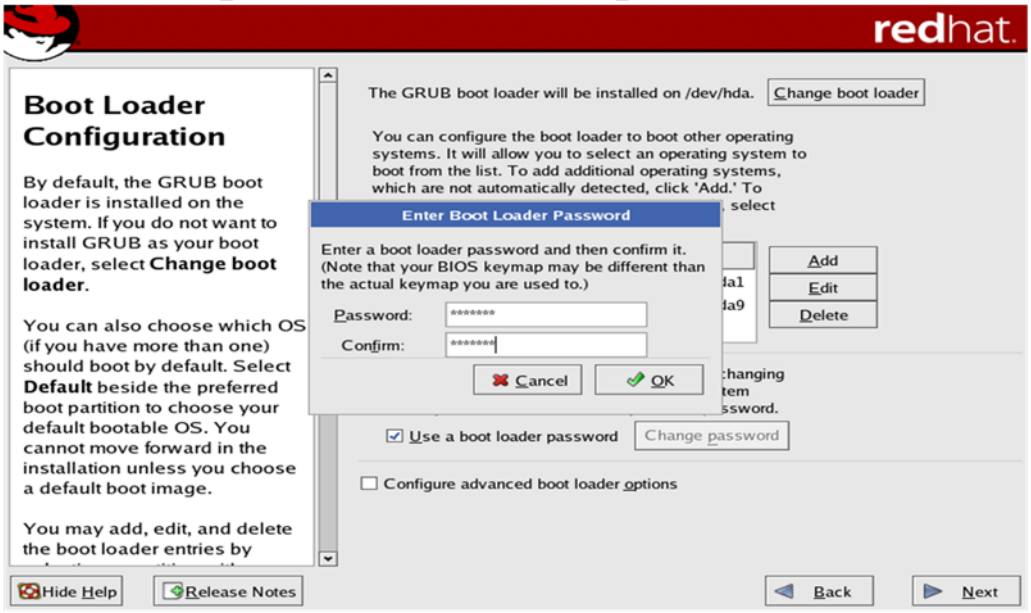


13 . Configuring boot loader options, also can give boot loader password for security reason.



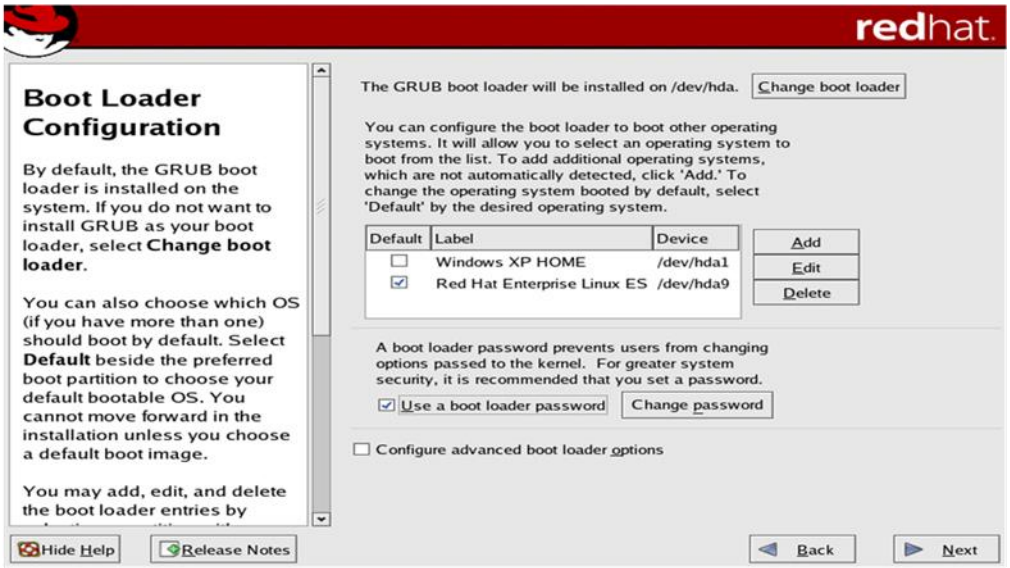
➔ Click on check Box Use a boot loader password

14 . Enter password & confirm password.



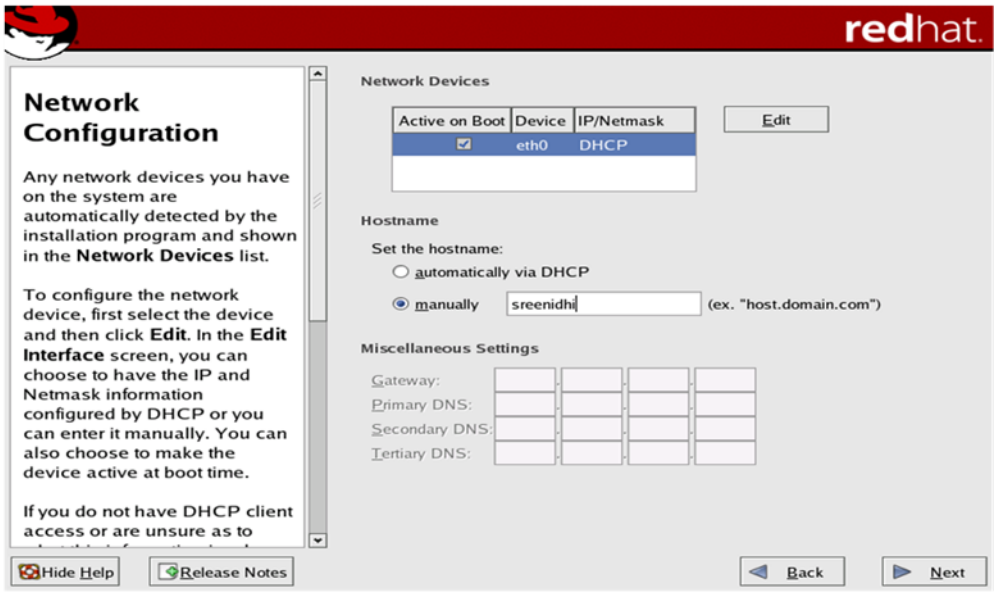
➔ Click on ok & can change the Labels Of RedHat & windows

15 . Select the Default BootLoader & can change label if needed.



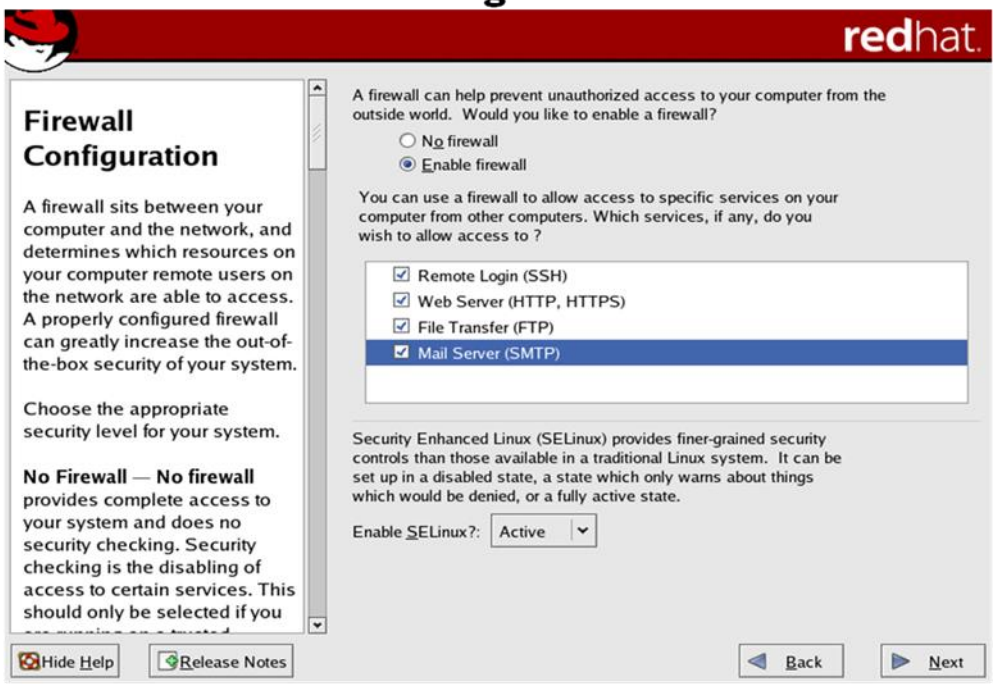
➔ Click on Edit to change Label & then click on Next

16 . Select Mannual or Automatically through DHCP.



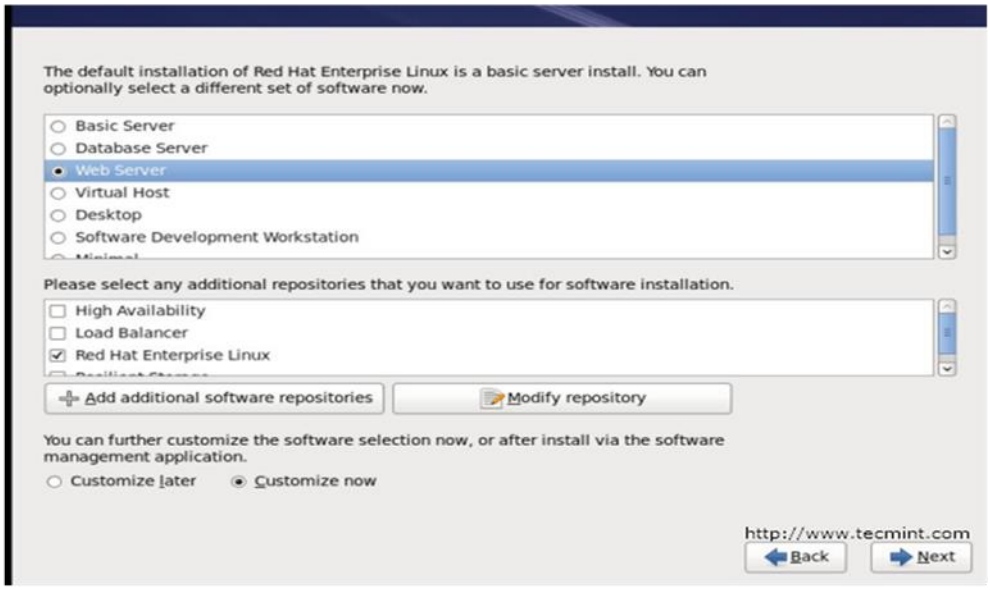
➔ Click on automatically via DHCP

17 . Select Firewall Configuration



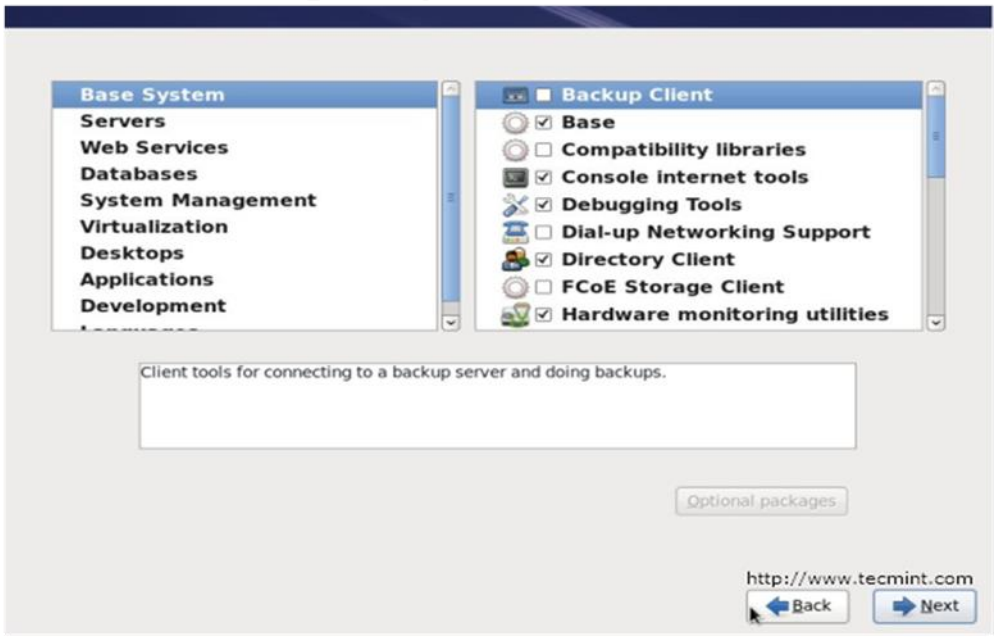
➔ Click on Next after selecting checkboxes of ftp,smtp etc.

18 . Select application to install by Selecting Web Server Option & click on Customize now option button.



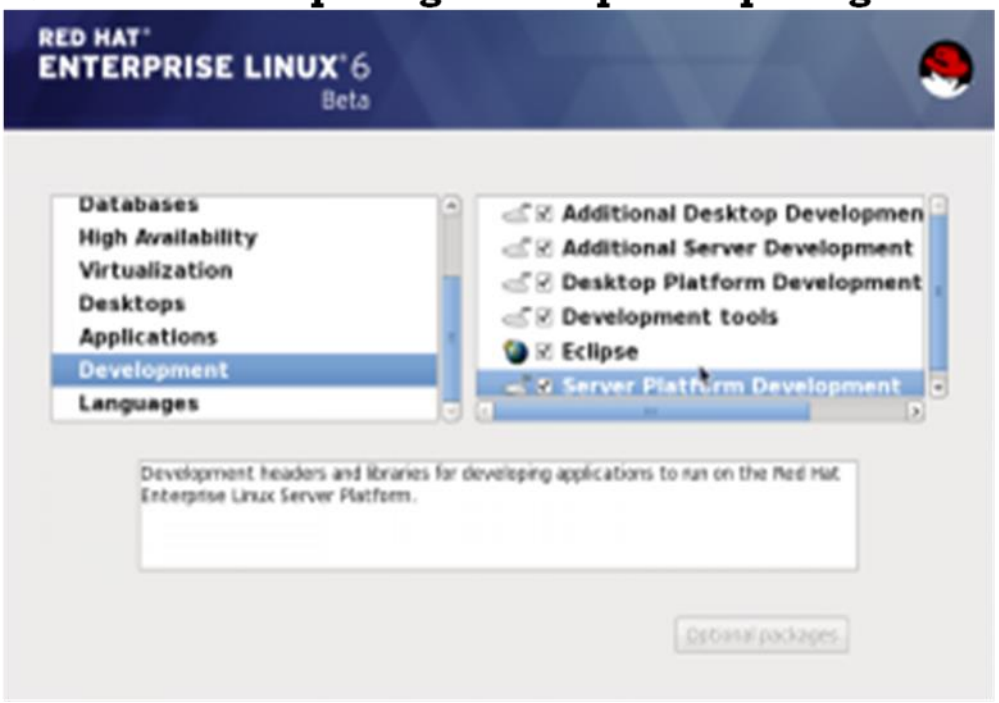
➔ Click on Next button to start of selecting software pkgs,

19 . Customize package selection



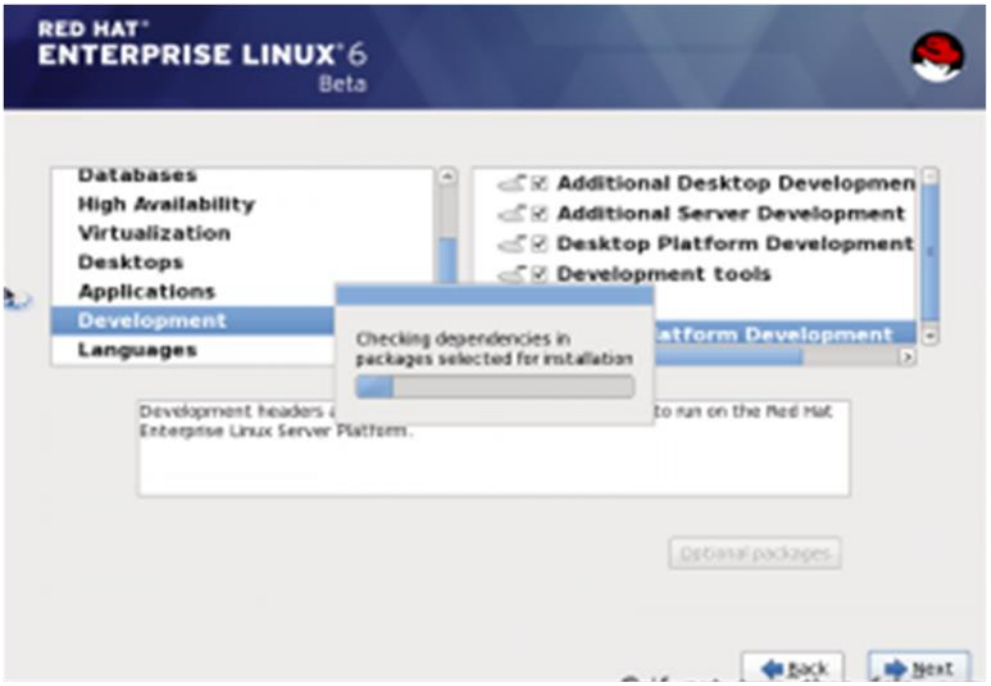
➔ Click on Next button to start of selecting software pkgs,

20. Select each package & its optional packages



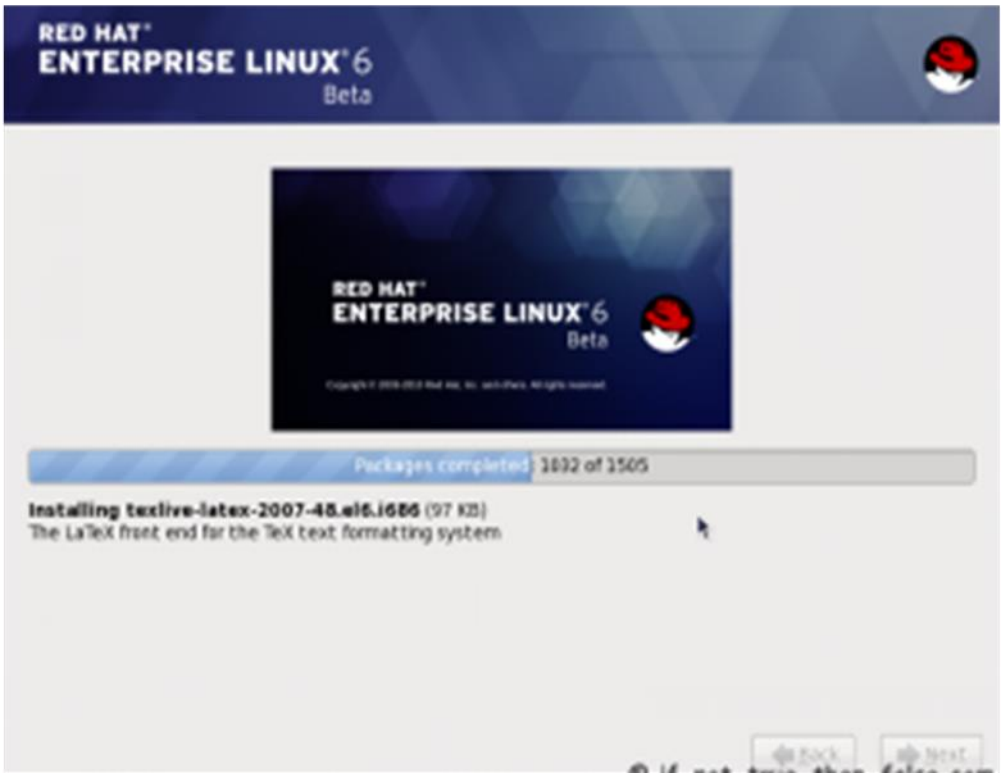
➔ Click on Next button to start of selecting software pkgs,

21. Checking dependencies for installation



➔ Click on Next button to start of selecting software pkgs,

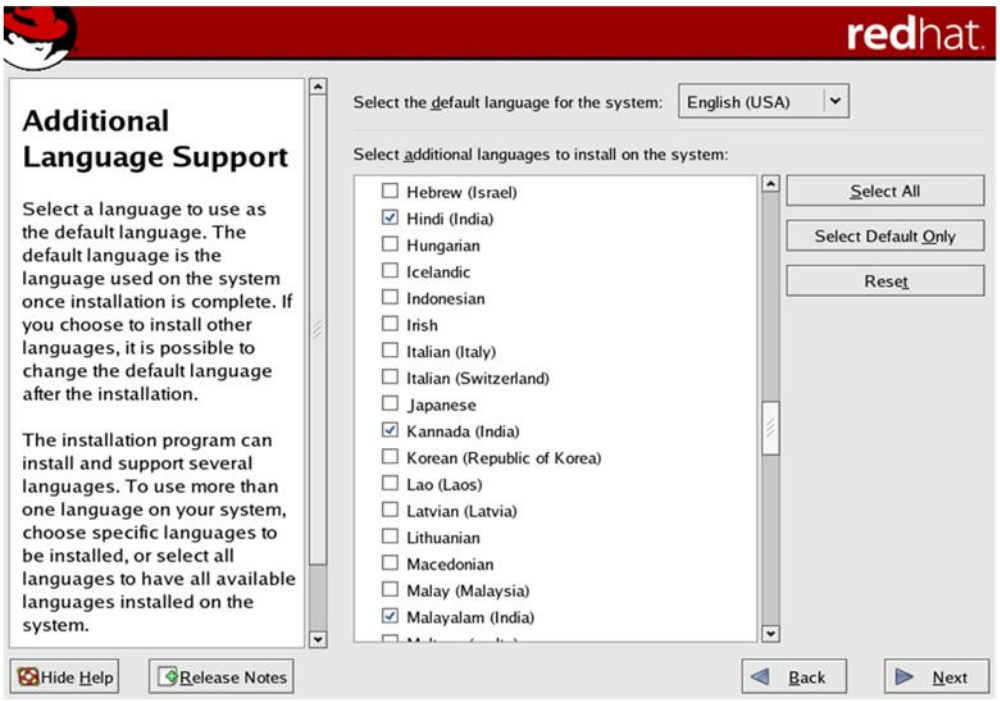
22 . Starting installation process



23. Installation is complete

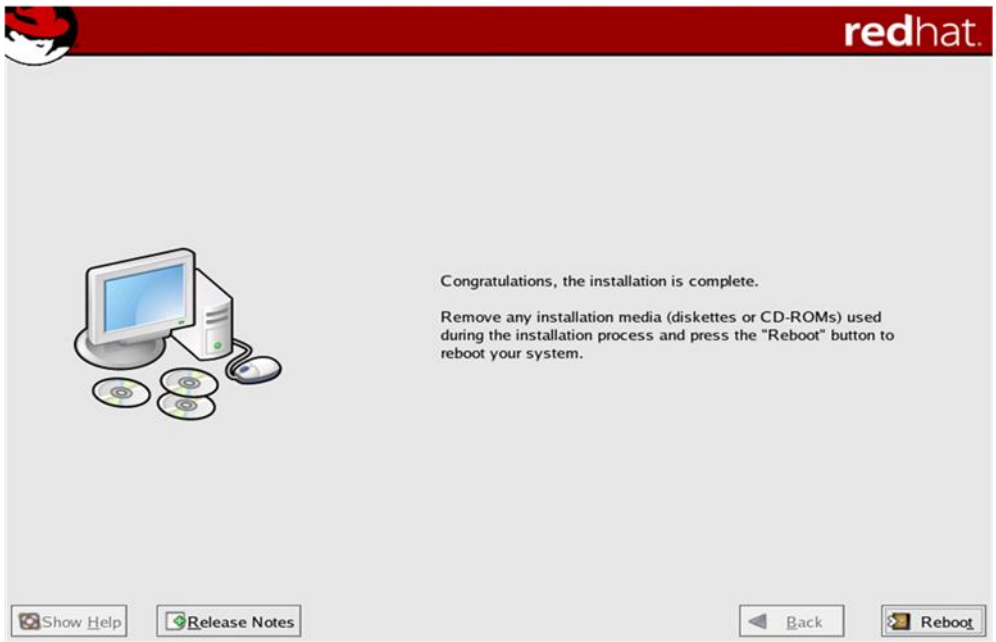
➔ Click on Next button

24. Select any Additional language if needed.



➔ Click on Next button

25. Click reboot button and remove installation Media.



➔ It will restart the machine

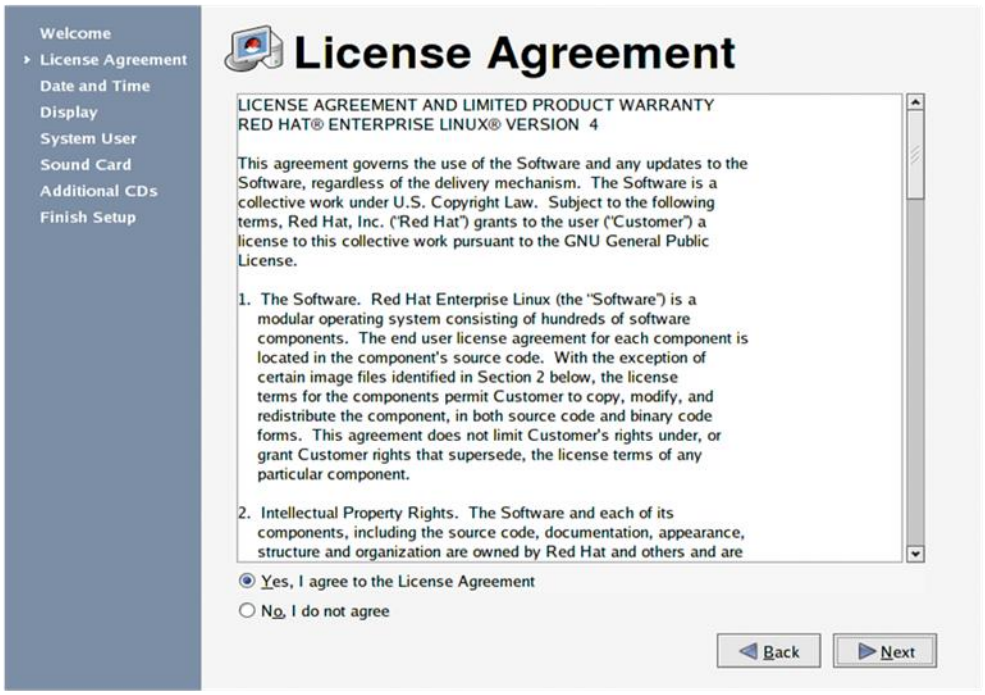
26. Selecting RHEL 6 from GRUB Loader

➔ Select redhat option to complete RedHat Installation

27. Displays Welcome screen

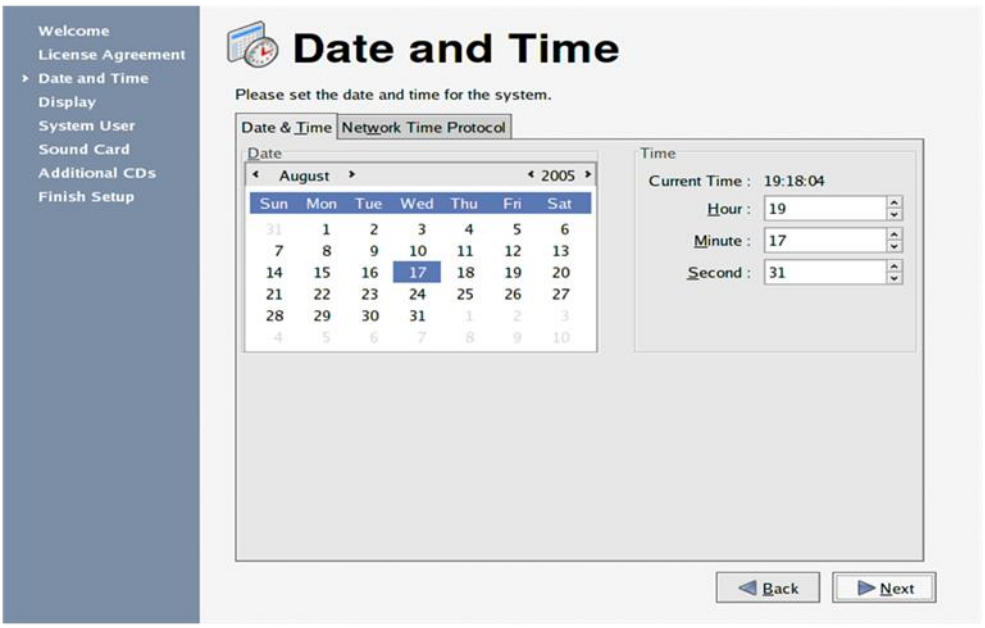
➔ Click on Next button

28. Displays License Agreement window



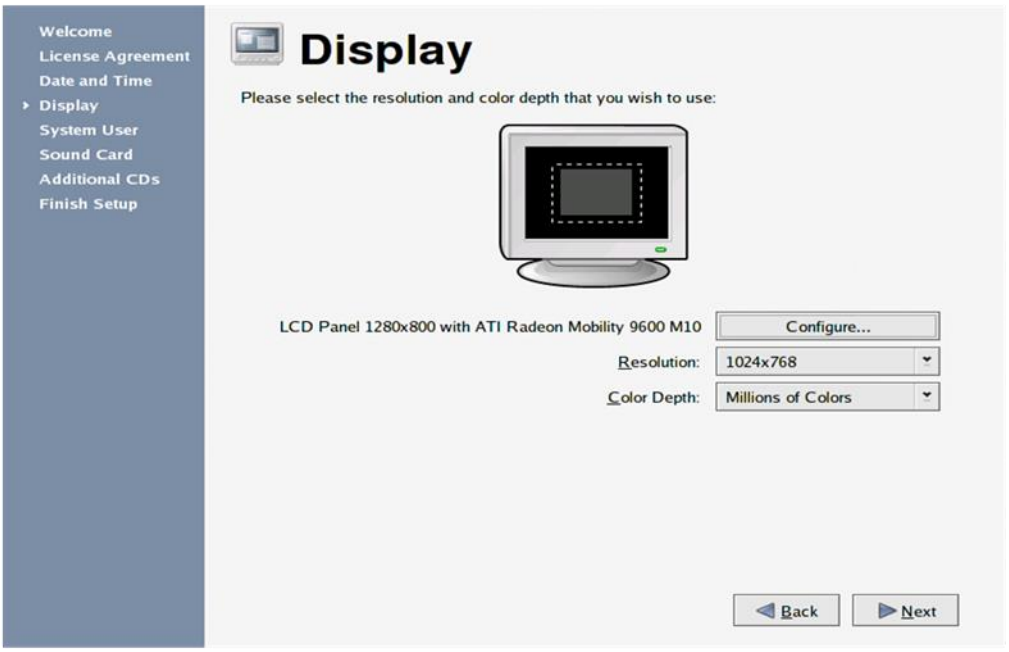
➔ Click Yes, I agree option button & then click on Next button

29 . Setup date and time and keep up-to-date with NTP



➔ Click on Next button

30. Identifying Display card

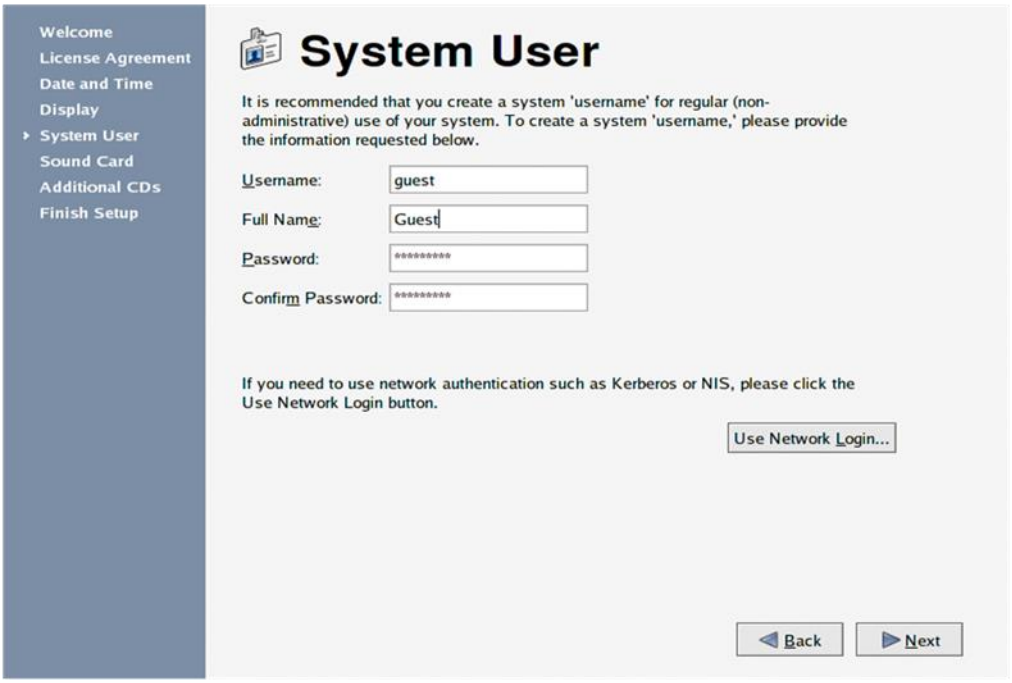


➔ Click on Next button

31. Detecting Sound card

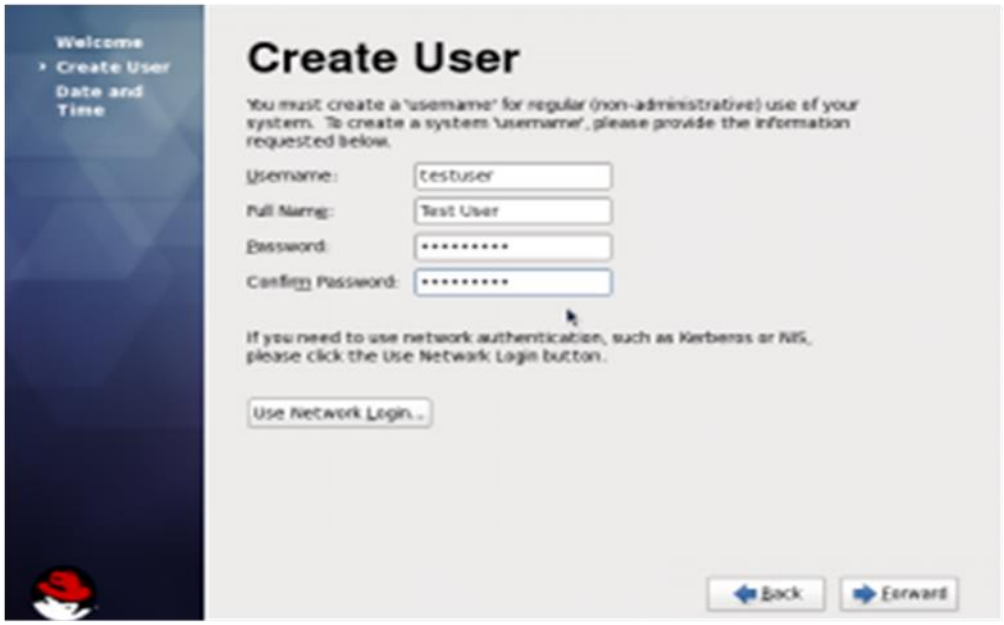
➔ Click on Next button

32. Can Create System user, Enter username & password



➔ Click on Next button

33. Can create normal user

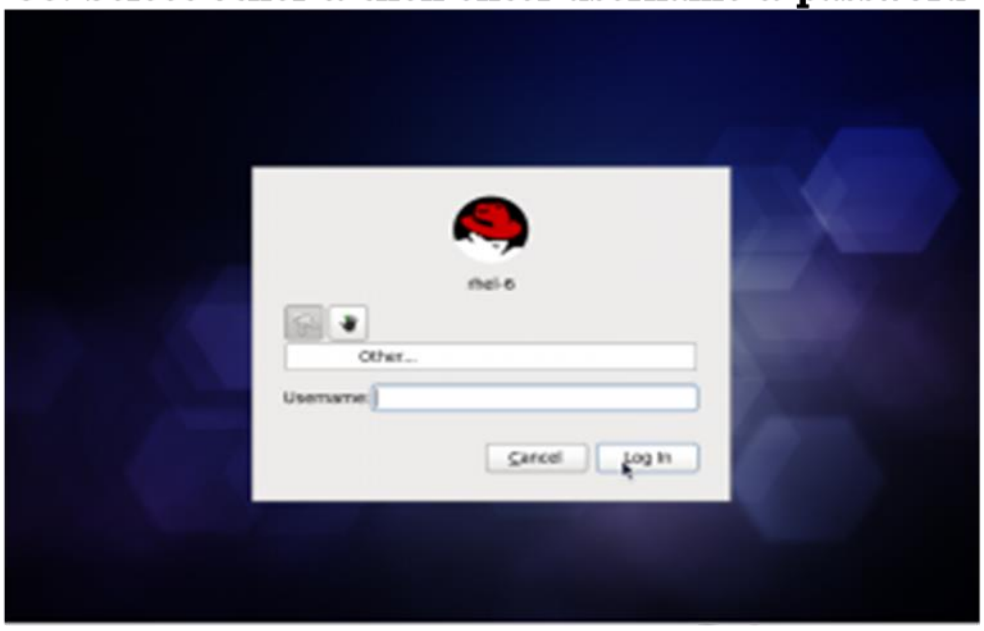


➔ Click on forward button.

34. Additional CDs

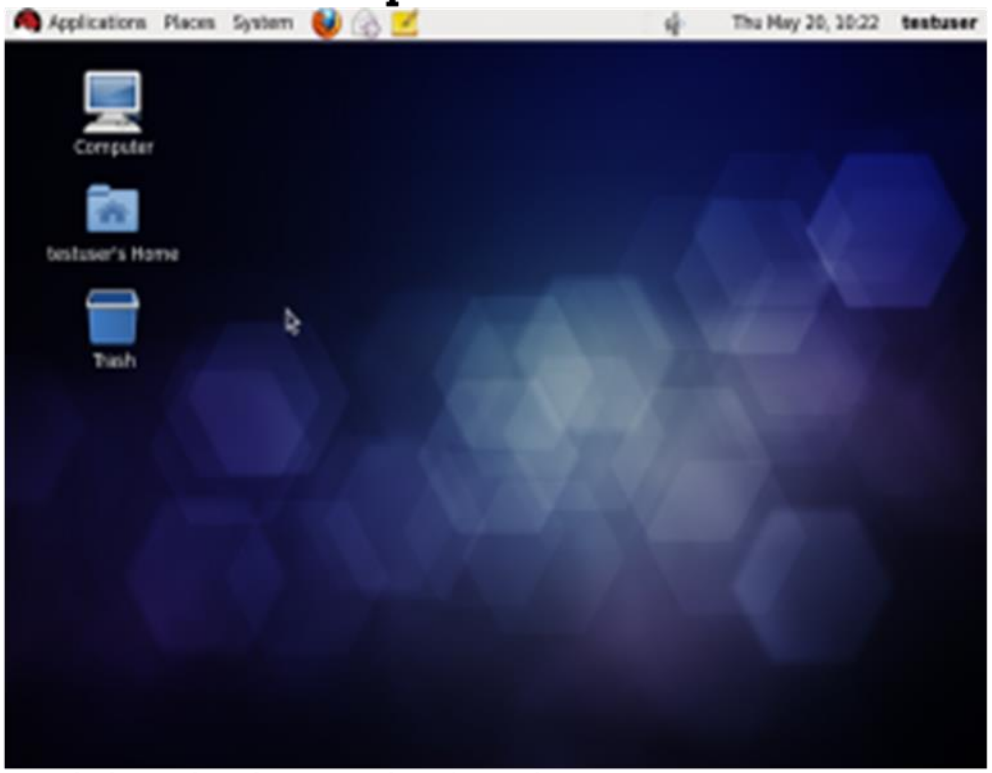
➔ Click on Next button

35. Select other & then enter username & password



➔ Click on Login to switch to 2nd Application window.

36 . This is Desktop



➔ Select shutdown option from System menu.

Practical 3:
Introduction to GRUB.CONF

GRUB.CONF

1) Grub.confg file is available in /boot directory
2) grub stands for **GRand Unified Bootloader**.
3) It is a new boottime operating system loader that Provides Dual Booting options for Red Hat 7.2 & windows and provides authority to the user to make a choice (either windows or Linux).
4) By default in current versions of fedora & Redhat GRUB Loader is used & in the previous versions boot loader used was

LILO (Linux Loader).

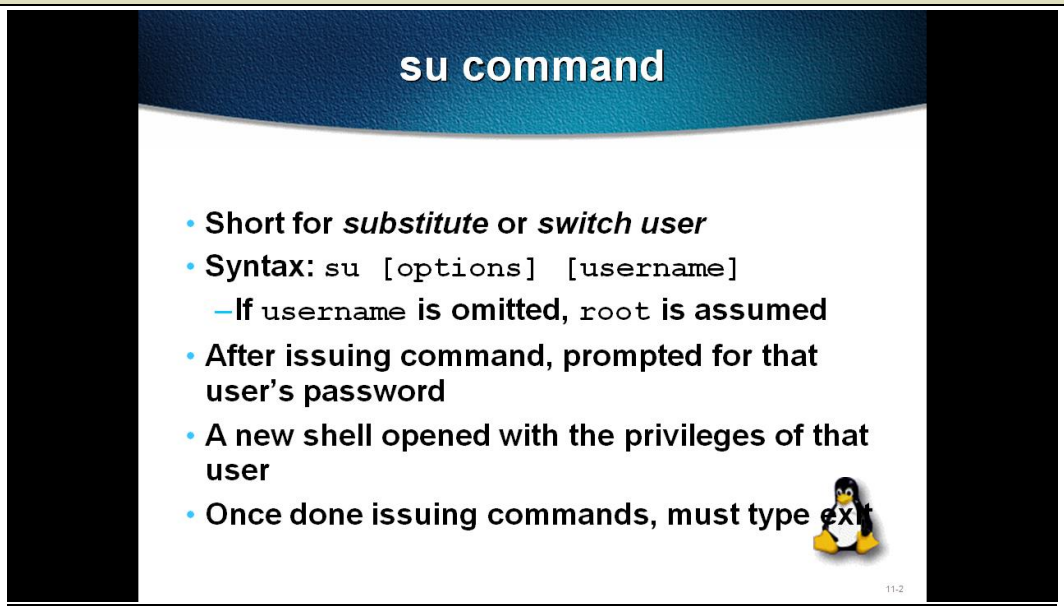
Here is a typical grub.conf file:

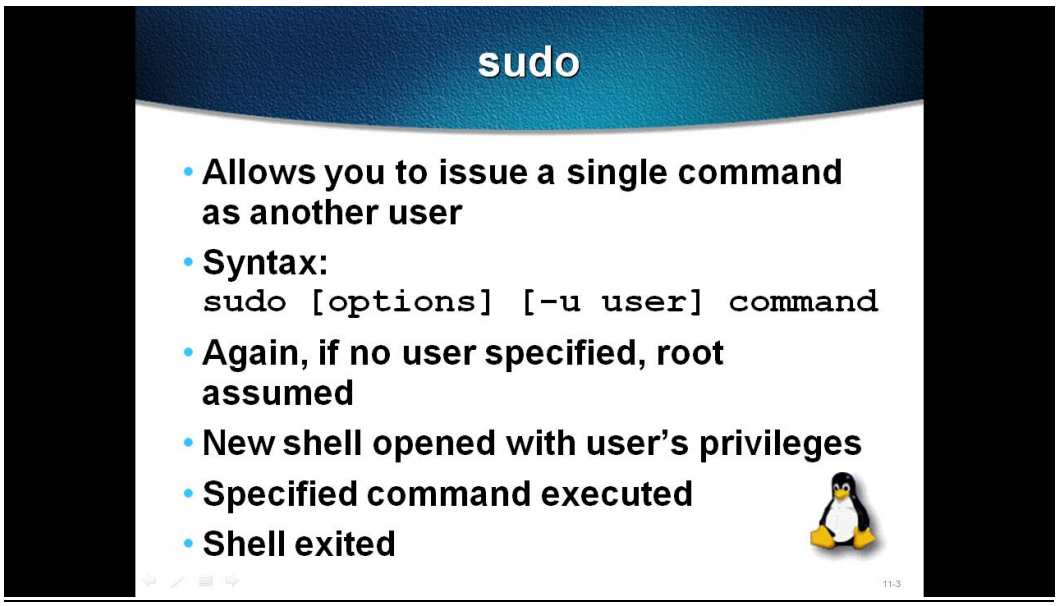
```
# grub.conf generated by anaconda
# Note that you do not have to rerun grub after making changes to this file
# NOTICE: You have a /boot partition. This means that
# all kernel and initrd paths are relative to /boot/, eg.
# root (hd0,1)
# kernel /vmlinuz-version ro root=/dev/sda8
# initrd /initrd-version.img
#boot=/dev/sda
default=0
timeout=10
title Red Hat Linux (2.4.16)
root (hd0,1)
kernel /vmlinuz-2.4.16 ro root=/dev/hda5 hdd=ide-scsi
initrd /initrd-2.4.9-13.img
title DOS
chainloader +1
```

1) default=0 line indicates that the first kernel section should be booted by default. grub starts its counting at 0 instead of 1.
2) The title line contains the label that will be shown in the boot menu for that kernel.
3) The root line specifies that Linux will be booted off the first hard drive.
4) The kernel line indicates the kernel's location on the file system.

In the DOS title section, notice that grub is calling a chain loader to be used for loading DOS. This is because grub doesn't support loading DOS. grub uses a chain loader to load any operating system that it doesn't support.

4.	**Practical 4:** **Linux System Administration**	





sudoers

- Must configure a user to run commands as another user when using `sudo`
- Permissions stored in `/etc/sudoers`
- Use utility `visudo` to edit this file (run as root)
- Permissions granted to users or groups, to certain commands or all, and with or without password being required

11-4

Temporarily changing user identity with su command

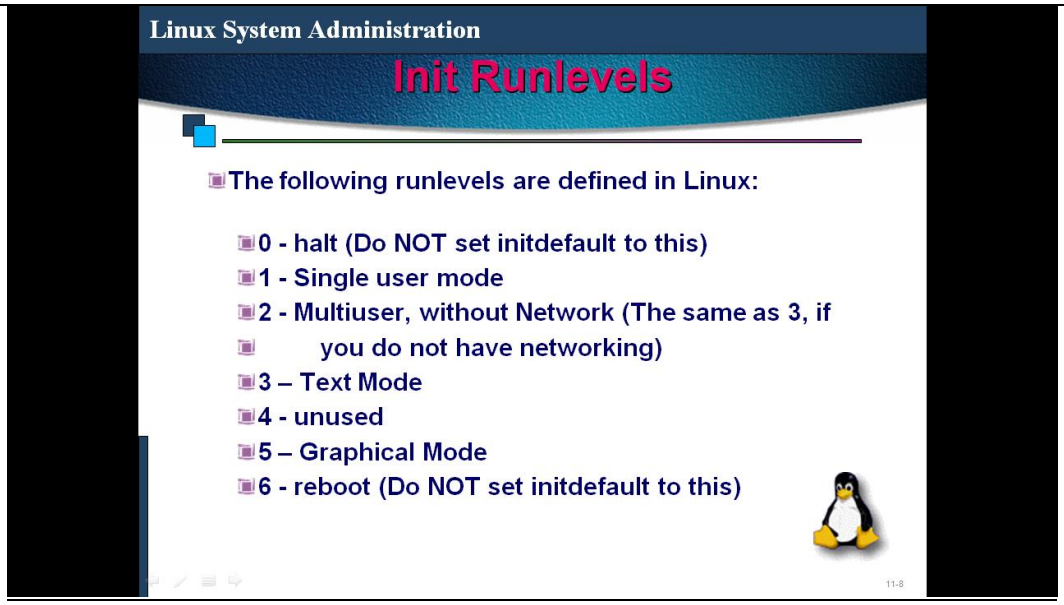
1. Login into another user account

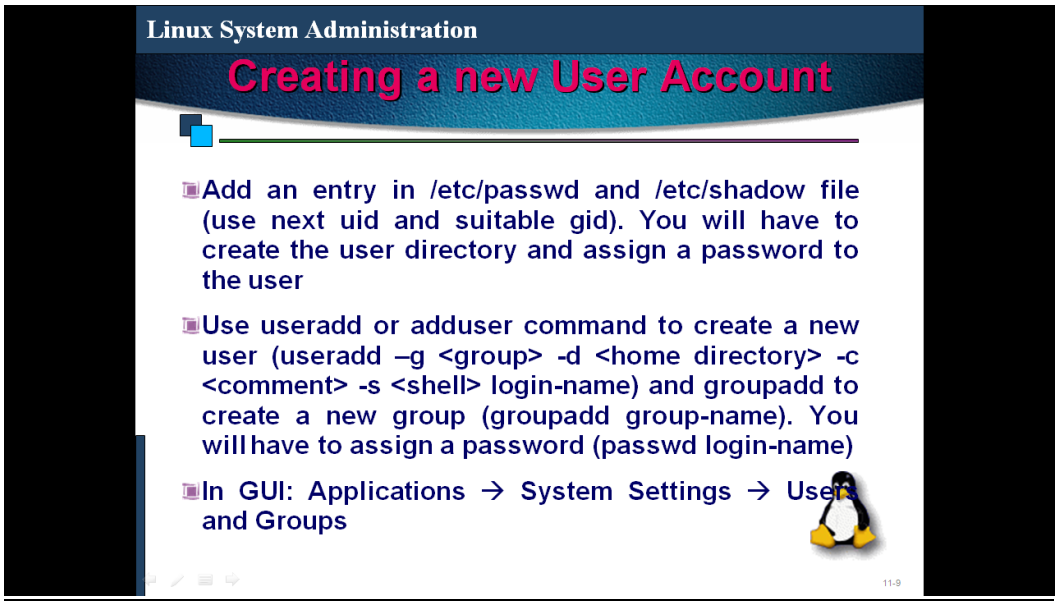
By simply passing the username to the su command, it will provide the login session after the password authentication as shown below

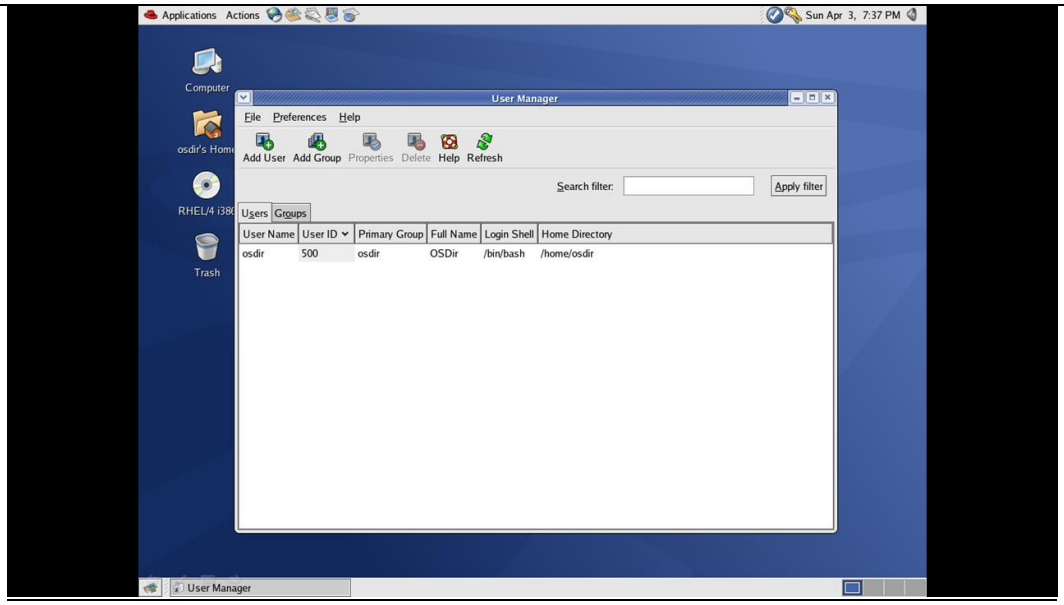
```
$ su guest
Password:

$
```

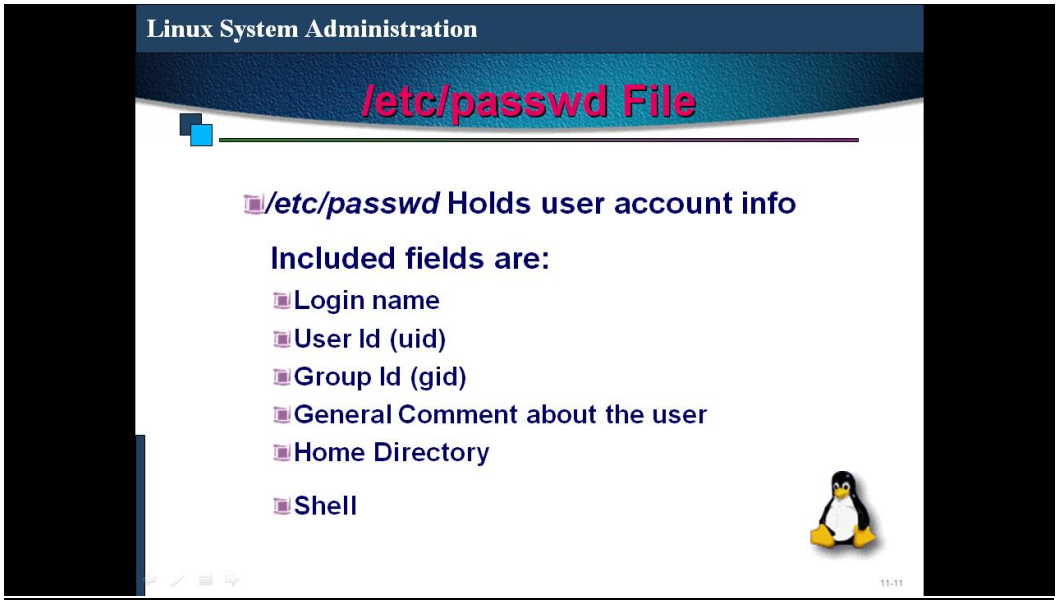
```
$    su root
 Password
#
```

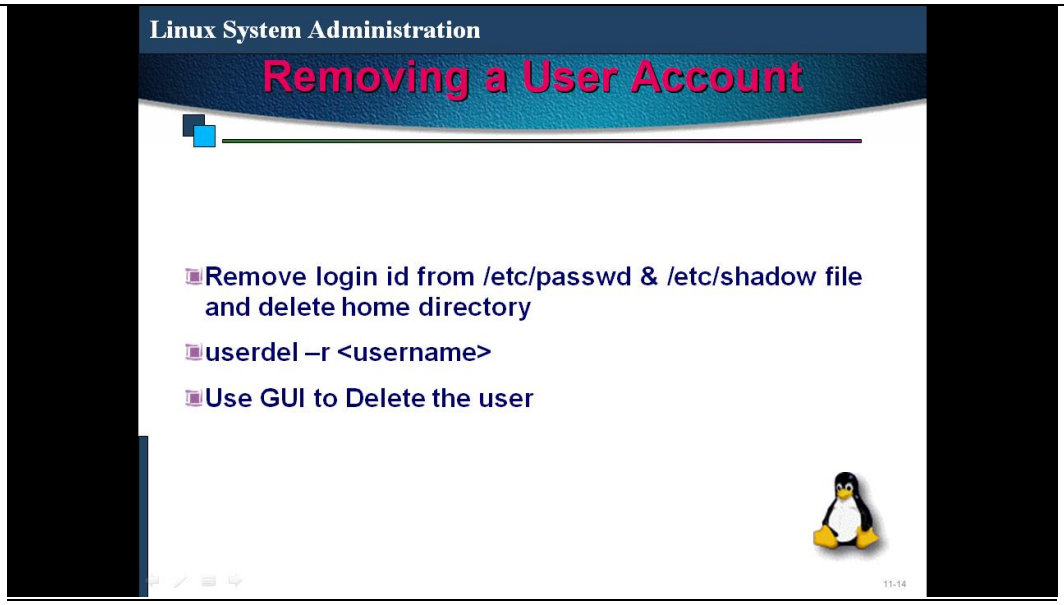
11-5

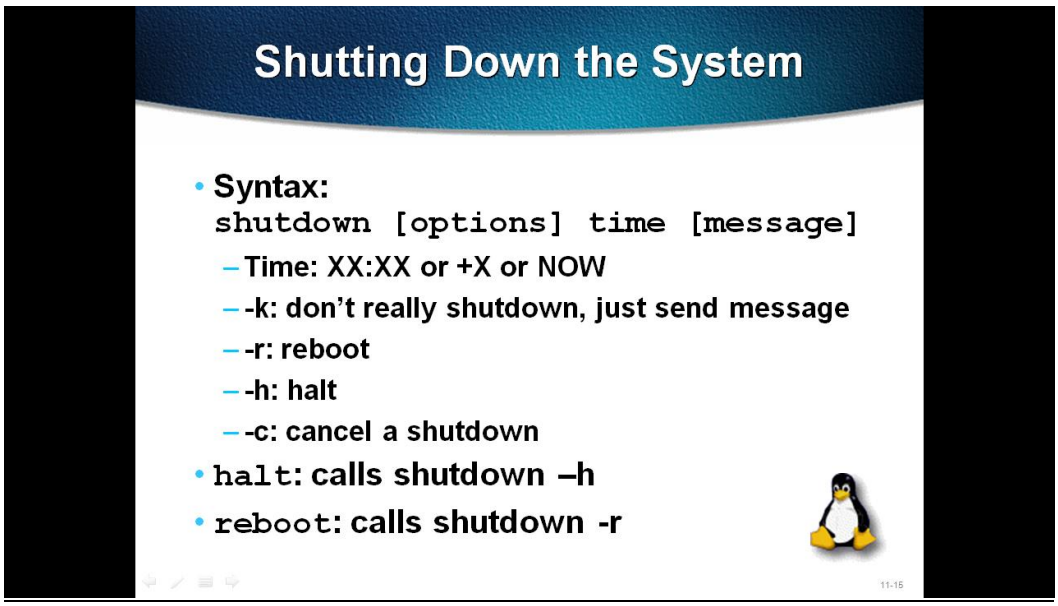

Linux System Administration
Init Runlevels
The following runlevels are defined in Linux:
0 - halt (Do NOT set initdefault to this)
1 - Single user mode
2 - Multiuser, without Network (The same as 3, if you do not have networking)
3 – Text Mode
4 - unused
5 – Graphical Mode
6 - reboot (Do NOT set initdefault to this)
11-8

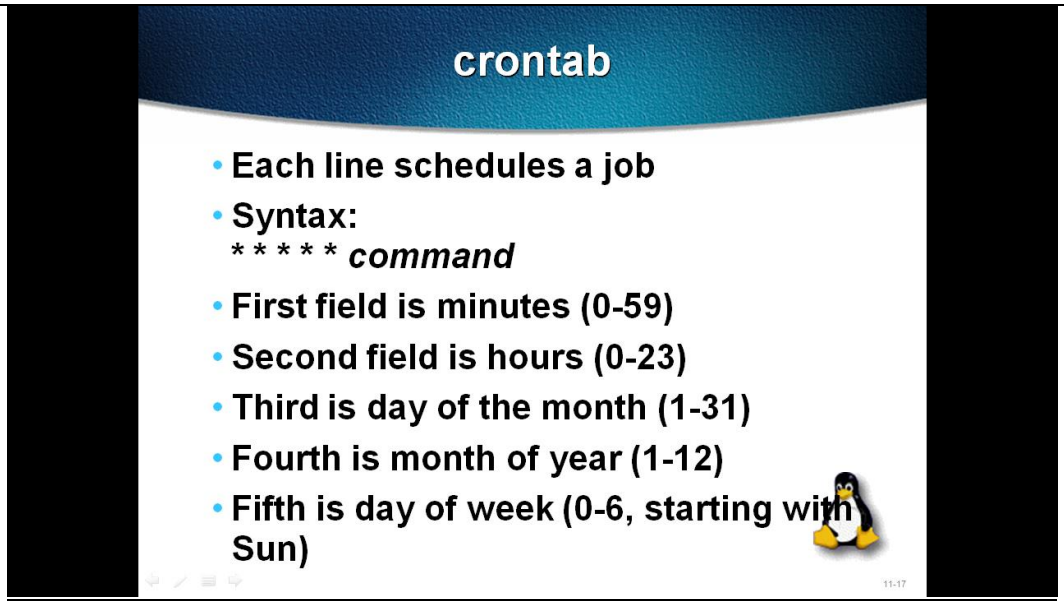

Linux System Administration
Creating a new User Account
Add an entry in /etc/passwd and /etc/shadow file (use next uid and suitable gid). You will have to create the user directory and assign a password to the user
Use useradd or adduser command to create a new user (useradd –g <group> -d <home directory> -c <comment> -s <shell> login-name) and groupadd to create a new group (groupadd group-name). You will have to assign a password (passwd login-name)
In GUI: Applications → System Settings → Users and Groups
11-9

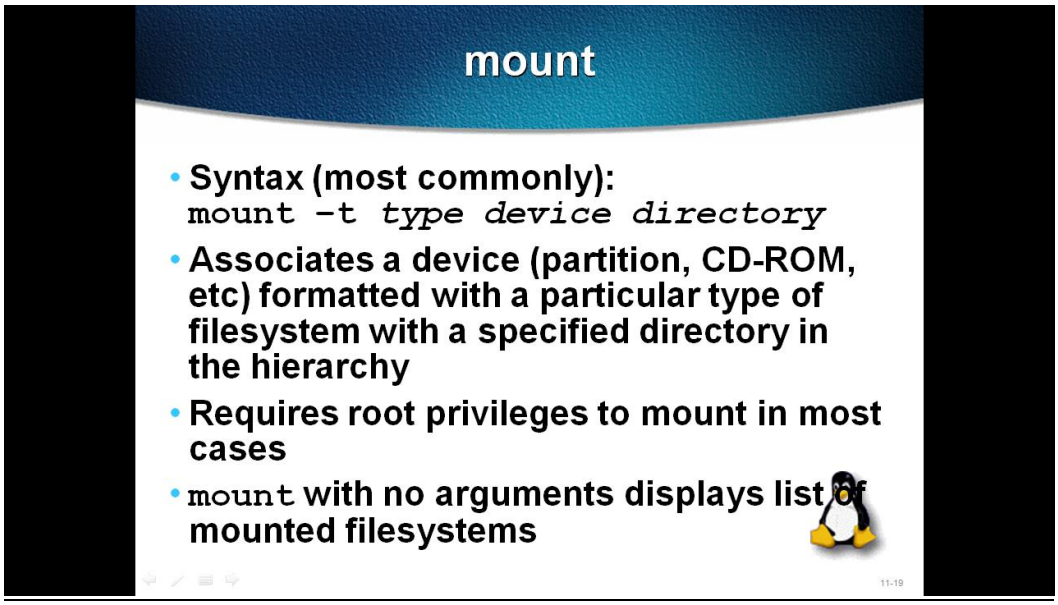

Applications Actions
Sun Apr 3, 7:37 PM
Computer
osdir's Home
RHEL/4 i386
Trash
User Manager
File Preferences Help
Add User Add Group Properties Delete Help Refresh
Search filter:
Apply filter
Users Groups
User Name User ID Primary Group Full Name Login Shell Home Directory
osdir 500 osdir OSDir /bin/bash /home/osdir
User Manager


Linux System Administration
/etc/passwd File
/etc/passwd Holds user account info
Included fields are:
Login name
User Id (uid)
Group Id (gid)
General Comment about the user
Home Directory
Shell
11-11


Linux System Administration
Removing a User Account
Remove login id from /etc/passwd & /etc/shadow file and delete home directory
userdel –r <username>
Use GUI to Delete the user
11-14


Shutting Down the System
Syntax:
shutdown [options] time [message]
Time: XX:XX or +X or NOW
-k: don't really shutdown, just send message
-r: reboot
-h: halt
-c: cancel a shutdown
halt: calls shutdown –h
reboot: calls shutdown -r
11-15


crontab
Each line schedules a job
Syntax:
* * * * * command
First field is minutes (0-59)
Second field is hours (0-23)
Third is day of the month (1-31)
Fourth is month of year (1-12)
Fifth is day of week (0-6, starting with Sun)
11-17


mount
Syntax (most commonly):
mount -t type device directory
Associates a device (partition, CD-ROM, etc) formatted with a particular type of filesystem with a specified directory in the hierarchy
Requires root privileges to mount in most cases
mount with no arguments displays list of mounted filesystems
11-19

umount

- Syntax:
  `umount` *`directory`* | *`device`*
- Removes that association
- Cannot `umount` if device is still being accessed (i.e. open files)
- Again, most likely requires root privileges

11-20

fstab

- For filesystems that should be mounted on boot every time, put them in `/etc/fstab`
- Basically a tab delimited file that contains the command line parameters you'd give to mount
  - Device
  - Mount point (directory)
  - FS type
  - Options (Readonly, attributes, etc)

11-21

Creating New Filesystems

- First use `fdisk` *`device`* to create a partition
  - Similar in function to old fdisk from DOS
  - Use ? to display commands, p to display partition info
- Once partition created, must be formatted
  - `mkfs -t` *`type filesystem`*
- Once formatted, you can mount it

11-22

Filesystem Integrity

- Filesystem problems? Corrupt files? Forced into single user mode to fix errors?
- `fsck`
- Syntax:
  `fsck [options] -t` *`type filesystem`*
- Again, usually need root permissions
- Also, filesystem should NOT be mounted while running `fsck` – can cause damage

11-23

RPM Package Names

<name>-<version>-<release>.<arch>.rpm

Name: name of the software package.

Version: version of the software package.

Release: release version of the RPM.

Arch: architecture (`i386`, `noarch`, `ppc`, etc.)

If Arch is `src`, RPM contains source code for building the package.

11-25

RPM options

Syntax

- rpm -i [options]
- (also rpm --install)
- rpm -U [options] (also rpm --upgrade)
- rpm -e [options] (also rpm --uninstall)
- rpm -q [options] (also rpm --query)
- rpm -V [options] (also rpm --verify)

11-28

Querying the RPM Database

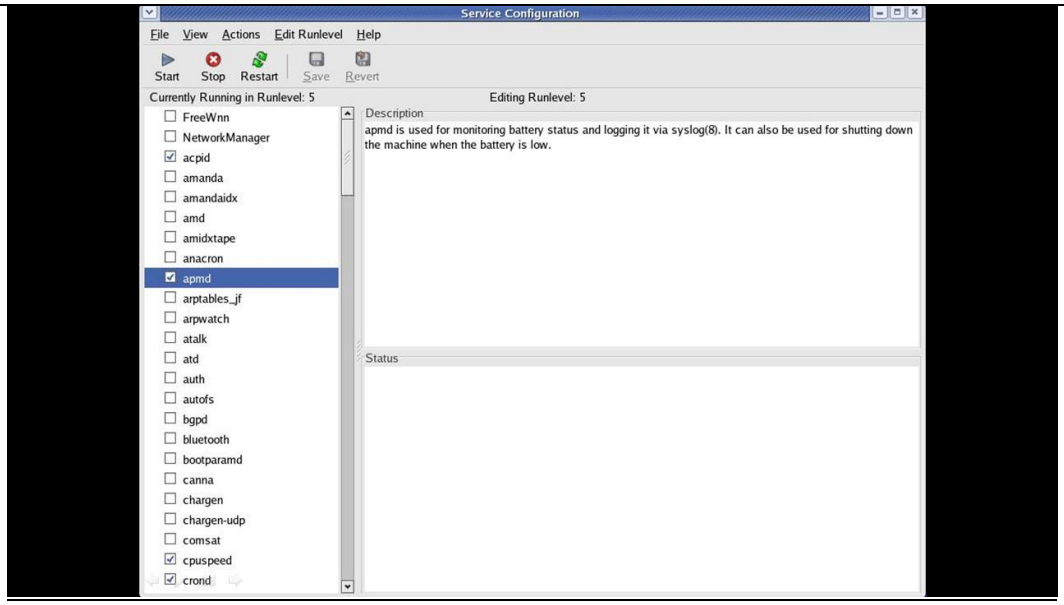
```
# rpm -q telnet
telnet-0.17-31.EL4.3
# rpm -ql telnet
/usr/bin/telnet
/usr/share/man/man1/telnet.1.gz
# rpm -qi telnet
Name        : telnet                       Relocations: (not)
Version     : 0.17                         Vendor: CentOS
Release     : 31.EL4.3                     Build Date: Tue 14 Jun 2005
Install Date: Sat 11 Feb 2006              Build Host: build5
Group       : Applications/Internet        Src RPM: telnet-0.17.src.rpm
Size        : 87254                        License: BSD
Signature   : DSA/SHA1, Tue 14 Jun 2005, Key ID a53d0bab443e1821
Packager    : Johnny Hughes <johnny@centos.org>
Summary     : Client program for telnet remote login protocol.
Description :
Telnet is a popular protocol for logging into remote systems
over the Internet. The telnet package provides a command line
telnet client.
```

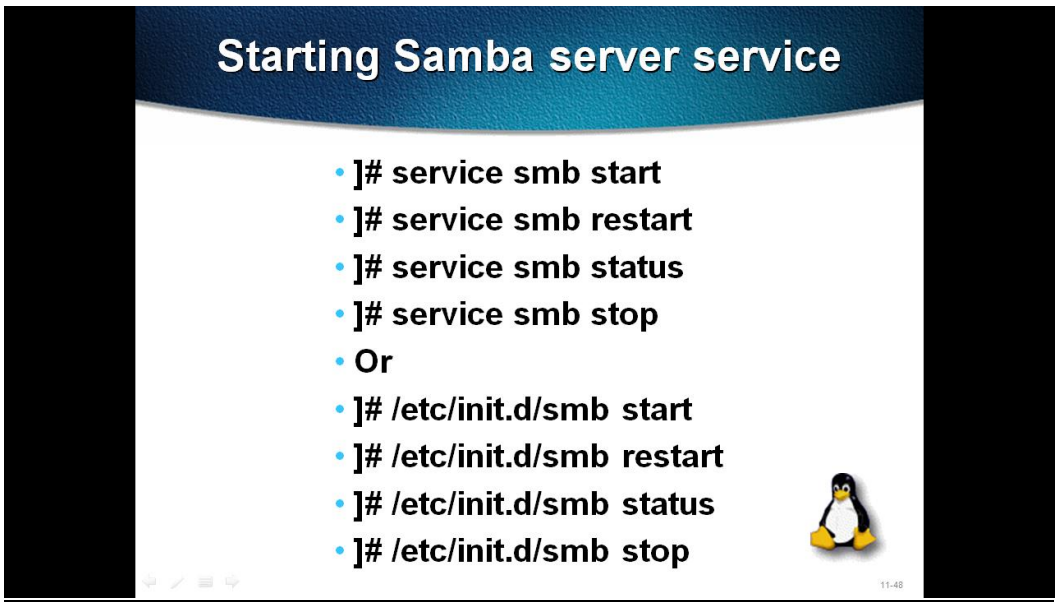
11-34

RPM Command Options

-i : install a package
-v : verbose
-h : print hash marks as the package archive is unpacked.
-q query operation
-a queries all installed packages
-f : file name
-d : refers documentation.
-p : specify a package name
l : list the files in the package

11-42


Service Configuration
File View Actions Edit Runlevel Help
Start Stop Restart Save Revert
Currently Running in Runlevel: 5
Editing Runlevel: 5
FreeWnn
NetworkManager
acpid
amanda
amandaidx
amd
amidxtape
anacron
apmd
arptables_jf
arpwatch
atalk
atd
auth
autofs
bgpd
bluetooth
bootparamd
canna
chargen
chargen-udp
comsat
cpuspeed
crond
Description
apmd is used for monitoring battery status and logging it via syslog(8). It can also be used for shutting down the machine when the battery is low.
Status


Starting Samba server service
]# service smb start
]# service smb restart
]# service smb status
]# service smb stop
Or
]# /etc/init.d/smb start
]# /etc/init.d/smb restart
]# /etc/init.d/smb status
]# /etc/init.d/smb stop

5.	**Practical 5:** **Setting up Linux as a Proxy server**	

Configure Linux SQUID Server Step By Step Guide Example and Implementation

Proxy servers operate as an intermediary between a local network and Internet. Requests from local clients for web services can be handled by the proxy server. Squid is a high-performance HTTP and FTP caching proxy server. It is also known as a Web proxy cache. As it stores data from frequently used Web pages and files, it can often give your users the data they need without their systems having to look to the Internet.

From squid web proxy server you can control what should be access on your network from internet. It could be act as a filter that could filter everything from porn site to advertise , videos.

In our example we will configure squid web proxy server and filter sites and deny permission to specific host from accessing internet.

Configure squid web proxy server

squid rpm is required to configure squid web proxy server check it for install if not found install it.

```
[root@server ~]# rpm -qa squid
squid-2.6.STABLE6-4.el5
[root@server ~]# _
```

Check the hostname and ip address of server it will be use in editing of squid.conf

```
[root@server ~]# hostname
server
[root@server ~]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 08:00:27:AF:A4:BB
          inet addr:192.168.1.3  Bcast:192.168.1.255  Mask:255.
          inet6 addr: fe80::a00:27ff:feaf:a4bb/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:466 errors:7 dropped:0 overruns:0 frame:0
          TX packets:54 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:40674 (39.7 KiB)  TX bytes:9019 (8.8 KiB)
          Interrupt:10 Base address:0xd020

[root@server ~]# _
```

Main Squid configuration file is squid.conf in the **/etc/squid/** directory. This file contains over 4000 lines in it, but only a few are active by default. Most of this file is filled with comments that describe most directives and associated options. To make editing easier use show line numbers options and locate desire tag from line number. We suggest you not to cram line number use them only to locate the desire tag as a simple enter can change the number of all lines in file. (set the line numbers by : set nu)

open /etc/squid/squid.conf for editing.

```
[root@server ~]# vi /etc/squid/squid.conf_
```

Show hidden line with : set nu option on vi command mode
You need to add three lines to the squid.conf file in the /etc/squid/ directory before activating Squid

First editing is about hostname locate visible_hostname tag near about line no 2835

```
2835 #  TAG: visible_hostname
2836 #       If you want to present a specia
     ,
2837 #       define this.  Otherwise, the re
```

Go in the end of this tag near about line no and add the hostname which you have checked in pervious command

```
2842 #Default:
2843 # none
2844 visible_hostname server_
2845 #  TAG: unique_hostname
```

By default squid works on port no 3128 but can change this. Port tag is located near line no 73

```
70 #       visible on the internal address.
71 #
72 # Squid normally listens to port 3128
73 http_port 3128
74
75 #  TAG: https_port
```

For our example we using the default port.

Next editing is to create access control list. Access control tag is located near the line no 2226

```
2226 # ACCESS CONTROLS
2227 # -----------------------------------
     -------
2228
2229 #  TAG: acl
2230 #       Defining an Access List
```

We will create three access list.

- First to block host with ip address 192.168.1.7 from accessing internet.
- Second to block a particular site.
- Third to allow our lab network for accessing internet.

Go in the end of access control tag near about line 2410 and create access list as show here

```
2408 acl Safe_ports port 777          # multiling http
2409 acl CONNECT method CONNECT
2410 acl deny_host  src 192.168.1.7
2411 acl allow_network src 192.168.1.0/24
2412 acl web_deny  dstdomain  "/etc/squid/web_deny"
2413
2414 #  TAG: follow_x_forwarded_for
```

Final editing is to implement whatever access list you have configured in access

list tag go to http access tag near line no 2482

```
2482 #  TAG: http_access
2483 #       Allowing or Denying access based on defined
2484 #
2485 #       Access to the HTTP port:
2486 #       http_access allow|deny [!]aclname ...
```

In the end of this tag near line no 2529 apply the configured access list

```
2528 # And finally deny all other access to this proxy
2529 http_access allow localhost
2530 http_access deny deny_host
2531 http_access deny web_deny
2532 http_access allow allow_network
2533 http_access deny all
2534 _
2535 #  TAG: http_access2
```

Be very careful about the order of access list alway put http_access deny all line in the end of all line. Whatever access list defined below the http_access deny all line will never be checked.

You have made necessary changed in squid.conf now save it and return to command prompt.

We have created a access list web_deny to filter the web traffic. We have set http_access deny web_deny tag in squid.conf. Now you can add the url of those websites in this file which you want block.

Now create /etc/squid/web_deny file.

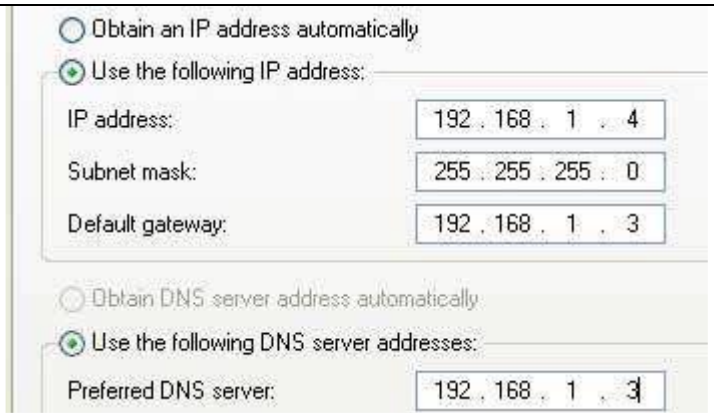
```
[root@server ~]# vi /etc/squid/web_deny
```

for testing purpose in our example we are blocking
www.google.com
you can add any sites url in this file which you want to block.

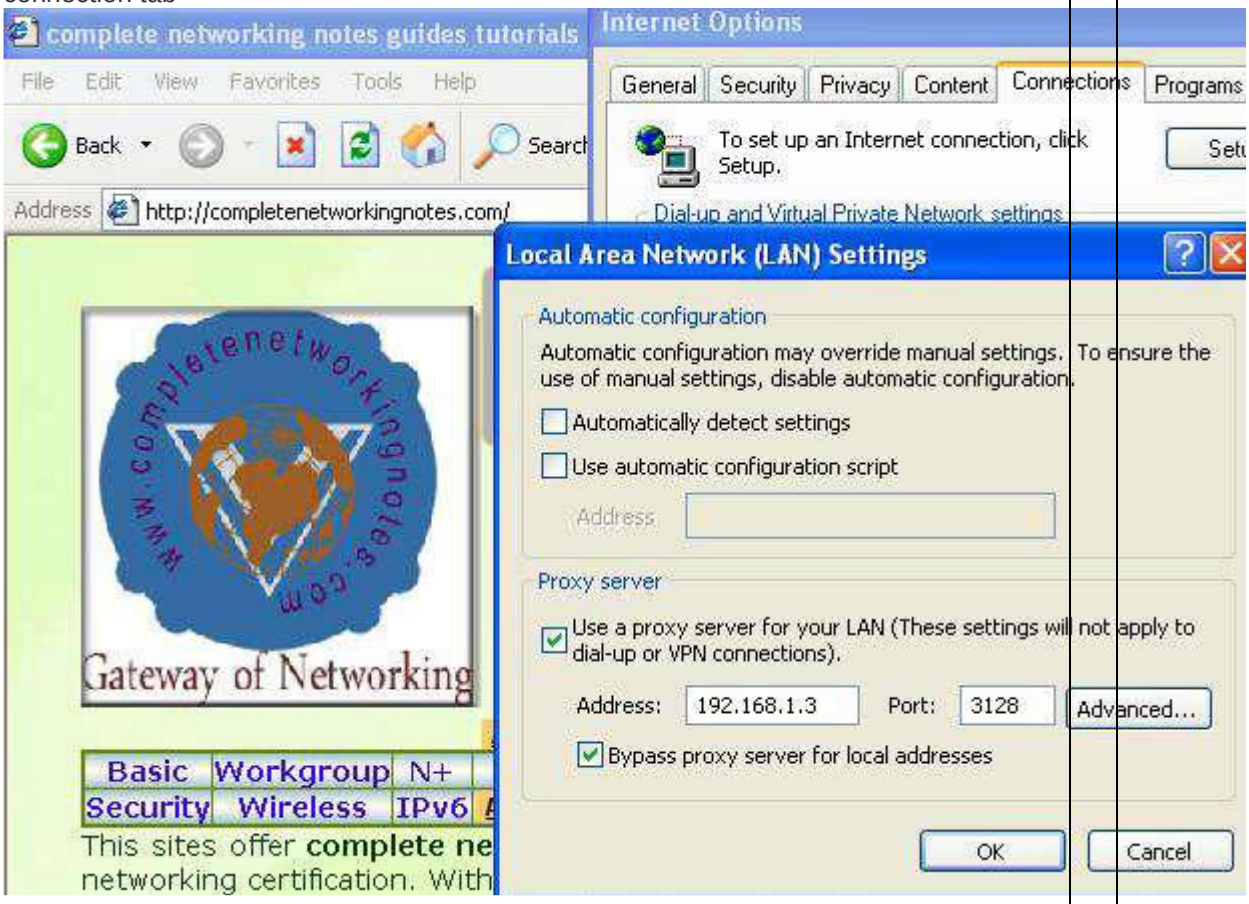
You have completed all necessary steps now start the squid service.

```
[root@server ~]# service squid restart
Stopping squid:                                            [  OK  ]
Starting squid: .                                          [  OK  ]
[root@server ~]# pgrep squid
2842
2845
[root@server ~]# _
```

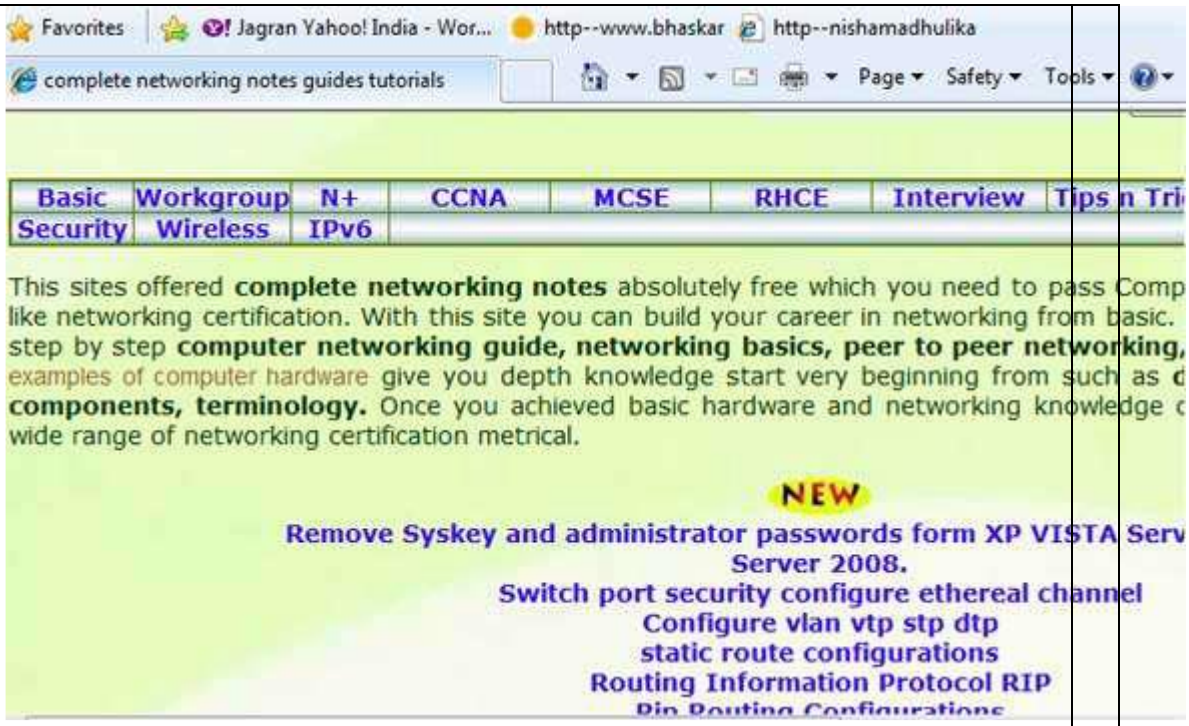
Squid client configuration
On client set the ip configuration. Set proxy servers ip 192.168.1.3 to default getway and dns server ip on client system.



Now open the web browser and set the port number and ip address of proxy server in connection tab



If you can successful retrieve website mean squid is working correctly



Now try to open www.google.com

Now go system which ip address is 192.168.1.7 and try to access after doing same setting


ERROR: The requested URL could not be retrieved - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://www.microsoft.com/isapi/redir.dll?prd=ie&pver=6&ar=msnhome
ERROR
The requested URL could not be retrieved

6.	**Practical 6:** **Setting up Samba Server**	

Samba Server

Samba is an Open Source Suite, that provides file and print services to SMB/CIFS clients. CIFS – Common Internet File System is a protocol that is basically an updated SMB.

Samba is freely available.

With Samba, you can share a Linux filesystem with Windows 95, 98, 2000 and NT and vice versa.

You can also share printers connected to either Linux or a system with Windows 95, 98, 2000 or NT.

Samba server uses smb service, SMB stands for – Server Message Block.

It is a protocol by which a lot of PC-related machines share files and printers and other information such as lists of available files and printers.

11-2

Samba Server Configuration

SAMBA SERVER INSTALLATION

[root@localhost home]# Now we will install samba server, since with the help of samba server we can share the linux files into windows machine

root@localhost home]# yum install samba
Loaded plugins: refresh-packagekit, rhnplugin
This system is not registered with RHN.
RHN support will be disabled.
Setting up Install Process
Nothing to do

root@localhost home]# rpm -qa | grep samba
sblim-cmpi-samba-1.0-1.el6.x86_64
samba-common-3.5.4-68.el6.x86_64
samba-winbind-clients-3.5.4-68.el6.x86_64
samba-3.5.4-68.el6.x86_64
samba-client-3.5.4-68.el6.x86_64

11-10

Samba Server Configuration Contd..

```
[root@localhost home]# pwd
/home
[root@localhost home]# mkdir kiransmb
[root@localhost home]# cd kiransmb
[root@localhost kiransmb]# touch f1 f2 f3
[root@localhost kiransmb]#
cat>kiransamba.txt
hello to every one at workshop, welcome to
all
[root@localhost kiransmb]# ls
f1  f2  f3  kiransamba.txt
```

11-11

Samba Server Configuration Contd..

```
root@localhost kiransmb]# vi /etc/samba/smb.conf
```

Workgroup = MYGROUP (windows machine workgroup name)
Uncomment line interface lo eth0 192.168.1.
Hosts allow 192.168.1.
At end of the file Add
[kiransmb]
comment = Kiran samba sharing Stuff
path = /home/kiransmb
public=yes
writable=yes
Broweasable = yes
write list = Administrator

```
[root@localhost kiransmb]# service smb start
Starting SMB services:
[root@localhost kiransmb]# service smb restart
Shutting down SMB services:                [  OK  ]
Starting SMB services: chkconfig           [  OK  ]
```

11-12

Samba Server Configuration Contd..

```
[root@localhost kiransmb]# chkconfig smb on
[root@localhost kiransmb]# testparm
Load smb config files from /etc/samba/smb.conf
rlimit_max: rlimit_max (1024) below minimum Windows
limit (16384)
Processing section "[homes]"
Processing section "[printers]"
Processing section "[kiransmb]"
Loaded services file OK.
Server role: ROLE_STANDALONE
Press enter to see a dump of your service definitions
[global]
        workgroup = MYGROUP
        server string = Samba Server Version %v
        interfaces = lo, eth0, 192.168.1.
        log file = /var/log/samba/log.%m
        max log size = 50
        hosts allow = 192.168.1.
        cups options = raw
```

11-13

Samba Server Configuration Contd..

```
[homes]
        comment = Home Directories
        read only = No
        browseable = No
[printers]
        comment = All Printers
        path = /var/spool/samba
        printable = Yes
        browseable = No
[kiransmb]
        comment = Kiran samba sharing Stuff
        path = /home/kiransmb
        write list = Administrator
        read only = No
        guest ok = Yes
[root@localhost kiransmb]# service iptables stop
[root@localhost kiransmb]# service iptables status
iptables: Firewall is not running.
```

11-14

Samba Server Configuration Contd..

```
[root@localhost kiransmb]# setenforce 0
[root@localhost kiransmb]# getsebool -a
abrt_anon_write --> off
allow_console_login --> on
allow_corosync_rw_tmpfs --> off
samba_enable_home_dirs --> off
[root@localhost kiransmb]# setsebool
samba_enable_home_dirs=1
[root@localhost kiransmb]# getsebool -a
exim_manage_user_files --> off
exim_read_user_files --> off
samba_enable_home_dirs --> on
[root@localhost kiransmb]# useradd kiransmbuser
[root@localhost kiransmb]# smbpasswd -a kiransmbuser
New SMB password:
Retype new SMB password:
Added user kiransmbuser.
[root@localhost kiransmb]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:E0:4C:A0:1A:AD
          inet addr:192.168.1.12  Bcast:192.168.1.255  Mask:255.255.255.0
```

11-15

Samba Server Configuration Contd..

```
[root@localhost kiransmb]# smbclient
//192.168.1.12/home/kiransmb -U kiransmbuser
Enter kiransmbuser's password:
Domain=[MYGROUP] OS=[Unix] Server=[Samba 3.5.4-68.el6]
tree connect failed: NT_STATUS_BAD_NETWORK_NAME

[root@localhost kiransmb]# service smb restart
Shutting down SMB services:                [  OK  ]
Starting SMB services:                     [  OK  ]

[root@localhost kiransmb]# chmod -R 777 /home/kiransmb
```

11-16

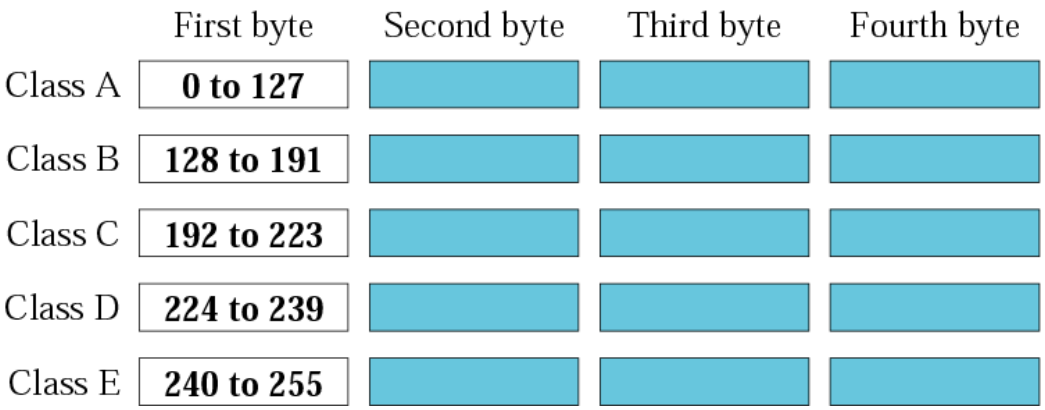
Samba Server Configuration Contd..

```
[root@localhost kiransmb]# Now our samba server is
configured
 [root@localhost kiransmb]# Now we will switch to
windows machine to see our folder or directory in
windows machine
 [root@localhost kiransmb]# ls
f1  f2  f3  kiransamba.txt  kiranwindows
[root@localhost kiransmb]# Now we are in Linux machine
back since we had created kiranwindows folder in
windows machine. Now we can view kiranwindows folder
here in Linux machine
[root@localhost kiransmb]# pwd
/home/kiransmb
[root@localhost kiransmb]# ls
f1 f2 f3 kiransamba.txt kiranwindows
```

11-17

7. Practical 7: Setting up Local area Network IP Addresses Classes & LAN Topology & Networking (TCP/IP) through manual (Statically) by using setup command or through Wizard.

IP ADDRESSES CLASSES



Class	1st Octet Decimal Range	1st Octet High Order Bits	Network/Host ID (N=Network, H=Host)	Default Subnet Mask	Number of Networks	Hosts per Network (Usable Addresses)
A	1 – 126*	0	N.H.H.H	255.0.0.0	126 (2^7 – 2)	16,777,214 (2^{24} – 2)
B	128 – 191	10	N.N.H.H	255.255.0.0	16,382 (2^{14} – 2)	65,534 (2^{16} – 2)
C	192 – 223	110	N.N.N.H	255.255.255.0	2,097,150 (2^{21} – 2)	254 (2^8 – 2)

D	224 – 239	1110	Reserved for Multicasting
E	240 – 254	1111	Experimental; used for research

Note: Class A addresses 127.0.0.0 to 127.255.255.255 cannot be used and is reserved for loopback and diagnostic functions.

Private IP Addresses

Class	Private Networks	Subnet Mask	Address Range
A	10.0.0.0	255.0.0.0	10.0.0.0 - 10.255.255.255
B	172.16.0.0 - 172.31.0.0	255.240.0.0	172.16.0.0 - 172.31.255.255
C	192.168.0.0	255.255.0.0	192.168.0.0 - 192.168.255.255

Linux IP Configuration and Network Configuration Commands Descriptions

Every node participating in networking needs a valid IP address. On Linux command prompt IP address is assigned by a network configuration window. This window can be invoked by selecting **network configuration** sub menu form **setup** command or directly executing **system-config-network** commands.

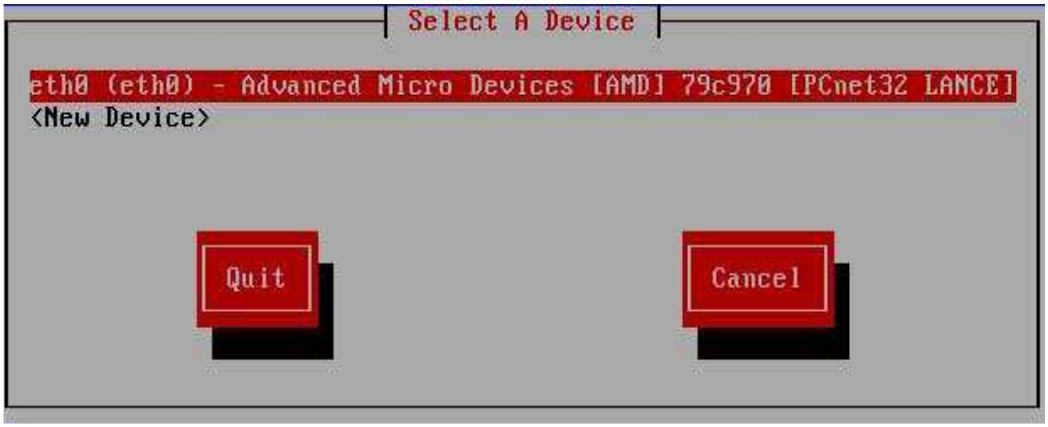
Run setup command form root user

```
#setup
```

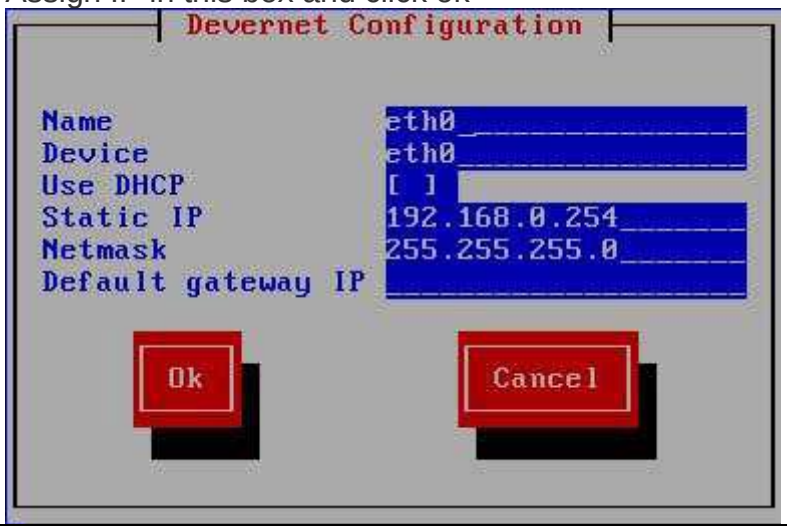
```
[root@localhost Server]# setup_
```

This will launch a new window select **network configuration**

Now a new window will show you all available LAN card select your LAN card (if you don't see any LAN card here mean you don't have install driver)



Assign IP in this box and click ok



Click on ok, quit and again quit to come back on root prompt.
Alternately you can use **system-config-network** command directly to invoke this setup window

```
#system-config-network
[root@Server ~]# system-config-network_
```

Whatever change you made in network configuration will not take place till you restart the LAN card

```
#service network restart
```

```
[root@localhost Server]# service network restart
Shutting down interface eth0:                              [  OK  ]
Shutting down loopback interface:                          [  OK  ]
Bringing up loopback interface:                            [  OK  ]
Bringing up interface eth0:                                [  OK  ]
[root@localhost Server]# _
```

Ifconfig

```
[root@Server devices]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0C:29:11:AD:E1
          inet addr:192.168.0.254  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe11:ade1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:195 errors:0 dropped:0 overruns:0 frame:0
          TX packets:232 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:36848 (35.9 KiB)  TX bytes:39270 (38.3 KiB)
          Interrupt:67 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:1602 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1602 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:3404782 (3.2 MiB)  TX bytes:3404782 (3.2 MiB)

[root@Server devices]# _
```

The **ifconfig** command will display the configuration of all active Ethernet card. Without specifying any parameter this command will show all active Ethernet card. if you want to see the configuration of any specific Ethernet card then use the name of that card as the command line arguments. for example to show the IP configuration on loop back Interface execute this command.

```
#ifconfig lo
```

```
[root@Server devices]# ifconfig lo
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:1602 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1602 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:3404782 (3.2 MiB)  TX bytes:3404782 (3.2 MiB)

[root@Server devices]# _
```

ifup/ifdown

```
[root@Server devices]# ifdown eth0
[root@Server devices]# ifup eth0
[root@Server devices]# _
```

Each installed network adapter has a corresponding **ifcfg-*** file in **/etc/sysconfig/network-scripts.** You can activate or deactivate that adapter with the ifup and ifdown commands. Either of the following commands will activate the eth0 network adapter:

```
#ifup ifcfg-eth0 #ifup eth0
```

netstat

```
[root@Server devices]# netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
192.168.0.0     *               255.255.255.0   U         0 0          0 eth0
169.254.0.0     *               255.255.0.0     U         0 0          0 eth0
[root@Server devices]# _
```

The netstat program provides real-time information on the status of your network connections, as well as network statistics and the routing table. The netstat command has several options you can use to bring up different sorts of information about your network.

arp

```
[root@Server devices]# arp
Address                  HWtype  HWaddress           Flags Mask            Iface
Client1                  ether   00:0C:29:62:28:1A   C                     eth0
[root@Server devices]# _
```

The Address Resolution Protocol associates the hardware address of a network adapter with an IP address. The arp command (in the /sbin directory) displays a table of hardware and IP addresses on the local computer. With arp, you can detect problems such as duplicate addresses on the network, or you can manually add arp entries as required.

mii-tool

```
[root@Server devices]# mii-tool
SIOCGMIIPHY on 'eth0' failed: Operation not supported
no MII interfaces found
[root@Server devices]# _
```

mii-tool command is used to check the link is activated or not. Most use of mii-tool command is to check to physical link of Ethernet card on command line. With this command you can check on command prompt that cable is plugged in LAN card or not.

ping

ping command is used to check the physical connectivity. If you get reply mean everything is ok. If you get request time out response means there is some problem it could be unplugged cable power off switch or enabled firewall on destination node. If you get Destination host unreachable means remote node is not in your network.

Use CTRL+C to abort the ping sequence

```
[root@Server devices]# ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
64 bytes from 192.168.0.1: icmp_seq=1 ttl=64 time=1.21 ms
64 bytes from 192.168.0.1: icmp_seq=2 ttl=64 time=0.289 ms

--- 192.168.0.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.289/0.750/1.212/0.462 ms
[root@Server devices]# _
```

service network restart

Whatever change you made in network configuration files will not take place until you restart the network services.
To implement change this command is used.

```
[root@localhost Server]# service network restart
Shutting down interface eth0:                              [  OK  ]
Shutting down loopback interface:                          [  OK  ]
Bringing up loopback interface:                            [  OK  ]
Bringing up interface eth0:                                [  OK  ]
[root@localhost Server]# _
```

How to use Linux as Router, Create Virtual LAN Card Example and Implementations

I will show you that how can you use Linux as a router. Routers are the devices those are used to connect two different networks. Routers are very costly devices. Linux could be a cost effective solution of routing in a small company.

Exam question
Your system is going use as a router for 192.168.0.0/24 and 192.168.1.0/24. Enable the IP forwarding.

Linux as a Router

In this practical we are using three computers. One Linux system will be use for routing and reset two will remain in two different networks. First we will configure the system which is going to play the role of router.

How to create virtual LAN card

Configure server system

You need two LAN card for routing between two networks or you can create virtual LAN card instead of deploying them physically.

To create virtual Ethernet card change directory to **/etc/sysconfig/network-scripts**

```
[root@Server ~]# cd /etc/sysconfig/network-scripts/
[root@Server network-scripts]# ls
ifcfg-eth0     ifdown-isdn     ifup-aliases  ifup-plip
ifcfg-lo       ifdown-post     ifup-bnep     ifup-plusb
ifdown         ifdown-ppp      ifup-eth      ifup-post
ifdown-bnep    ifdown-routes   ifup-ippp     ifup-ppp
ifdown-eth     ifdown-sit      ifup-ipsec    ifup-routes
ifdown-ippp    ifdown-sl       ifup-ipv6     ifup-sit
ifdown-ipsec   ifdown-tunnel   ifup-ipx      ifup-sl
ifdown-ipv6    ifup            ifup-isdn     ifup-tunnel
[root@Server network-scripts]# _
```

ifcfg-eth0 is the necessary script file for Ethernet 0. Copy this file to the same folder to create new virtual LAN cards.

```
[root@Server network-scripts]# cp ifcfg-eth0 ifcfg-eth0.1
[root@Server network-scripts]# _
```

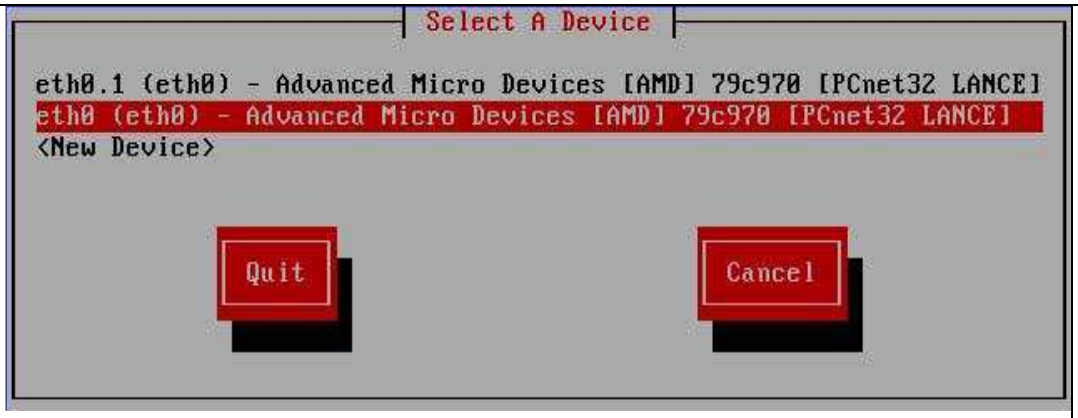
Now on this newly created virtual LAN card. It could be done by **service network restart**

```
[root@Server network-scripts]# service network restart
Shutting down interface eth0:                              [  OK  ]
Shutting down loopback interface:                          [  OK  ]
Bringing up loopback interface:                            [  OK  ]
Bringing up interface eth0:                                [  OK  ]
Bringing up interface eth0.1:                              [  OK  ]
[root@Server network-scripts]# _
```

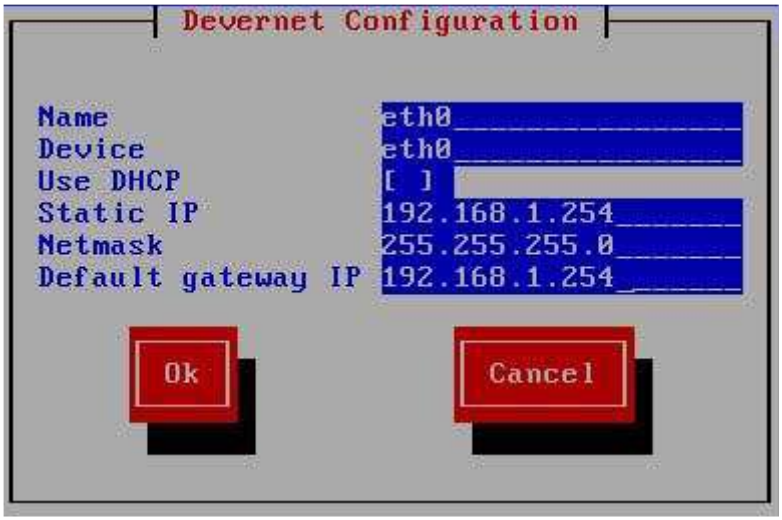
Run **setup** command and select network configuration sub window from list

```
[root@Server network-scripts]# setup_
```

You have two LAN card here, select **eth0** from list to assign IP

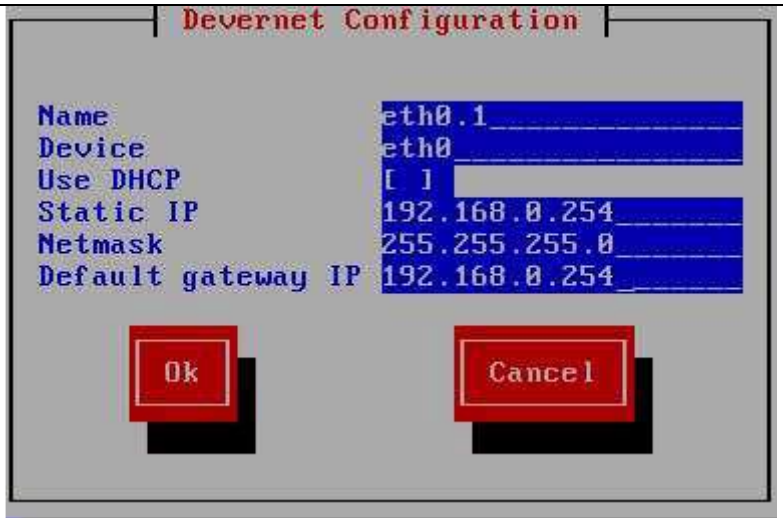


This Ethernet card will be the default gateway of first network set its IP to **192.168.1.254** and click on ok



Now select eth0.1 (This our virtual LAN card which we create in our last sections)

Set its IP to **192.168.0.254** This will be the default gateway of other network. Click on **OK** then **quit** and **quit** to come back on command prompt



IP forwarding can be enabled by editing in **/etc/sysctl.conf** file. open this file

```
[root@Server network-scripts]# vi /etc/sysctl.conf _
```

Locate the **net.ipv4. ip_forward = 0 tag**. and replace the value **0** to **1**. This will enable IP forwarding to permanently . But this require a system reboot.

```
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 1_

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Do not accept source routing
net.ipv4.conf.default.accept_source_route = 0
```

If don't want to restart the system you can tell running kernel directly by **echo** command and kernel will enable the IP forwarding

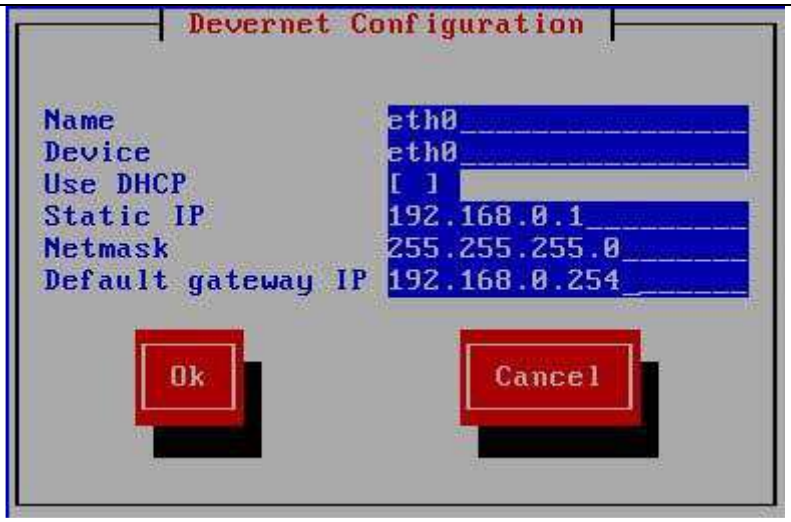
```
[root@Server network-scripts]# echo "1"> /proc/sys/net/ipv4/ip_forward
[root@Server network-scripts]# _
```

now configure our client system. we are using two system one from each network to test the connectivity .

Our first system is a Linux machine run **setup** command on it

[root@Client1] # setup_

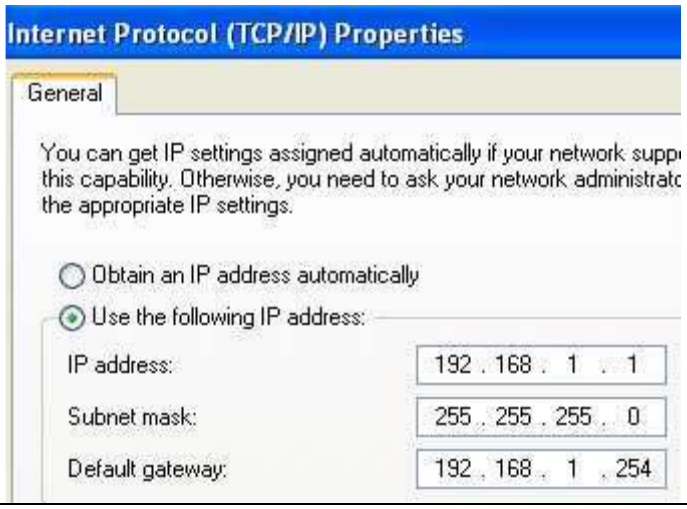
assign its IP address to **192.168.0.1** with a default gateway of **192.168.0.254**



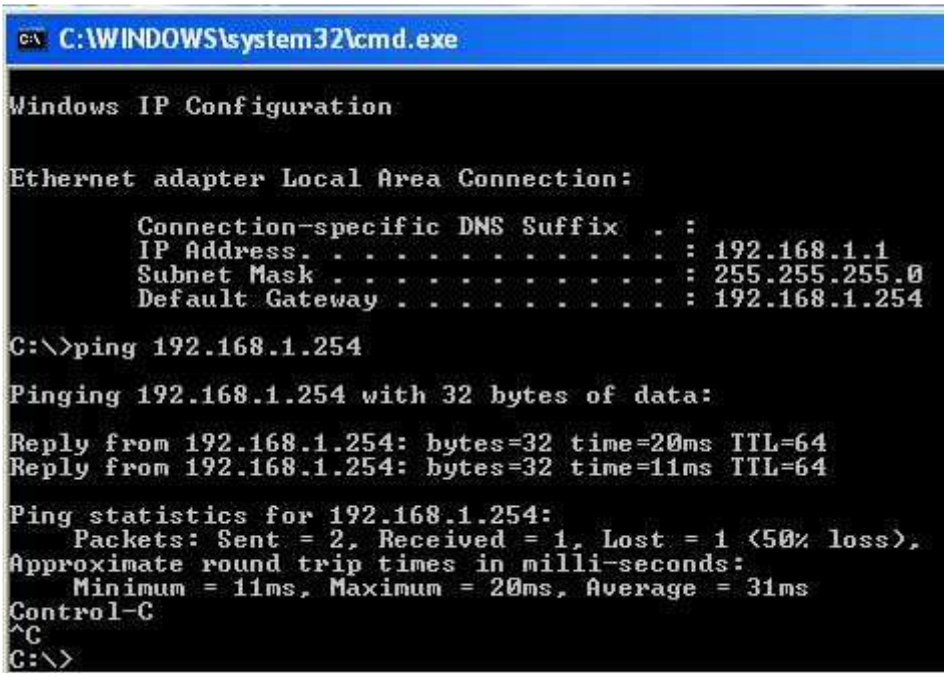
Now restart the network service and check connectivity form its default gateway
(Server IP)

```
[root@Client1 network-scripts]# service network restart
Shutting down interface eth0:                              [  OK  ]
Shutting down loopback interface:                          [  OK  ]
Bringing up loopback interface:                            [  OK  ]
Bringing up interface eth0:                                [  OK  ]
[root@Client1 network-scripts]# ping 192.168.0.254
PING 192.168.0.254 (192.168.0.254) 56(84) bytes of data.
64 bytes from 192.168.0.254: icmp_seq=1 ttl=64 time=3.97 ms
64 bytes from 192.168.0.254: icmp_seq=2 ttl=64 time=0.282 ms

--- 192.168.0.254 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 0.282/2.126/3.970/1.844 ms
[root@Client1 network-scripts]# _
```

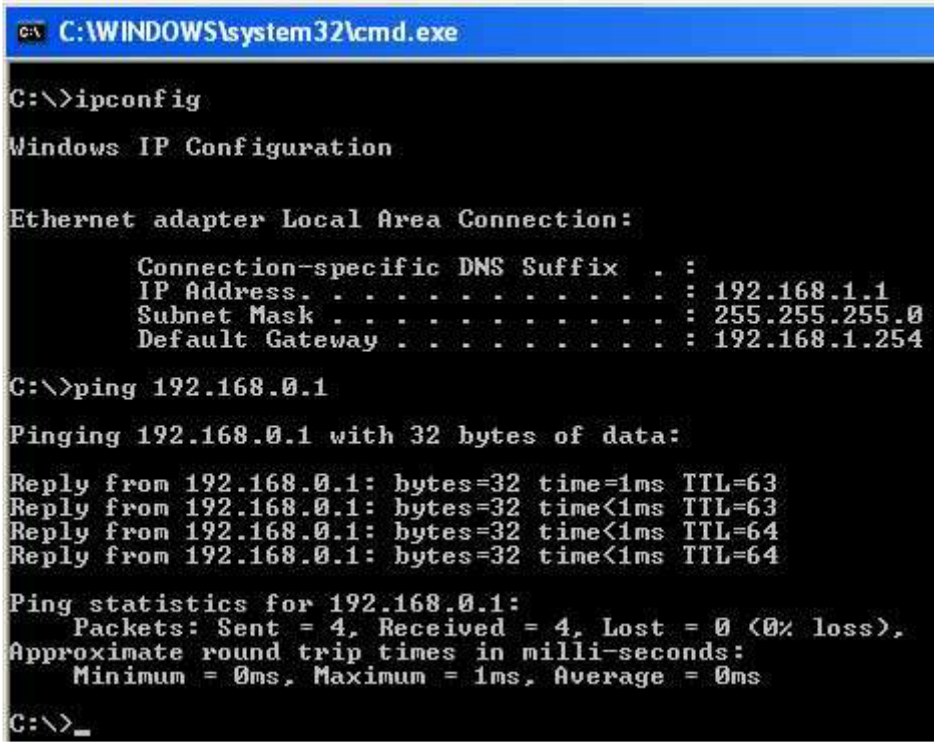
Now go on our other host which we are using a window machine (You can also use Linux host) and set IP address to **192.168.1.1** with a default gateway to **192.168.1.254**



Now open command prompt and test connectivity with default gateway



At this point you have completed all necessary step's to enable routing its time to verify this **Test from windows system ping** the Linux host located on other network



Test from Linux system

ping the Window host located on other network

```
[root@Client1 network-scripts]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:29:62:28:1A
          inet addr:192.168.0.1  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe62:281a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:344 errors:0 dropped:0 overruns:0 frame:0
          TX packets:436 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:46609 (45.5 KiB)  TX bytes:69077 (67.4 KiB)
          Interrupt:67 Base address:0x2000

[root@Client1 network-scripts]# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
From 192.168.0.254: icmp_seq=1 Redirect Host(New nexthop: 192.168.1.1)
64 bytes from 192.168.1.1: icmp_seq=1 ttl=127 time=0.808 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=127 time=0.525 ms

--- 192.168.1.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.525/0.666/0.808/0.143 ms
[root@Client1 network-scripts]# _
```

Basic Network Configurations Example and Implementations

In this article we will discuss all those necessary steps which you need to perform before solving networking related questions in RHCE exam. Don't skip this tutorial, giving few minutes to this could save you from huge problems in exam. All steps are given in a sequences don't skip them whether you them or not.

Check LAN card driver is installed or not.

LAN driver is the top most part for network. To check it run **setup** command

```
[root@localhost Server]# setup_
```

Select **network configuration** from list

If you see LAN card here mean you have LAN driver

If you don't see here anything and Linux drop you back on list manu means you don't have LAN driver. Install is first.

Check proper IP configuration

All systems on RHCE exam should have an properly configured IP address. During this entire practical we are using three systems. There description is given below Check these systems for properly configured IP address.

Node	Operating system	Name	IP
PC1	Linux RHEL	Server	192.168.0.254
PC2	Linux RHEL	Client1	192.168.0.1
PC3	Windows XP	Client2	192.168.0.2

Change host name

If you have skipped hostname during installation then it would be **localhost.localdomain.** You can change hostname with **hostname**command but this change would be temporary. To change hostname permanently do editing in **/etc/sysconfig/network.**

set computer name as shown in table.

#vi /etc/sysconfig/network

```
NETWORKING=yes
NETWORKING_IPV6=no
HOSTNAME=Server
```

On server system set it to **Server** and in client system set it **Client1**

```
[root@Client1 ~]# cat /etc/sysconfig/network
NETWORKING=yes
NETWORKING_IPV6=no
HOSTNAME=Client1

[root@Client1 ~]# _
```

Check /etc/hosts files for name resolution

Several Linux server depend name resolution. This file should have entry of all network systems. It will save you from naming related problem. In our network it should look like this on both Linux system **Server** and **Client1**

```
[root@Server devices]# cat /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       localhost.localdomain   localhost
::1     localhost6.localdomain6 localhost6
192.168.0.254   Server  Server
192.168.0.1     Client1 Client1
192.168.0.2     Client2 Client2

[root@Server devices]# _
```

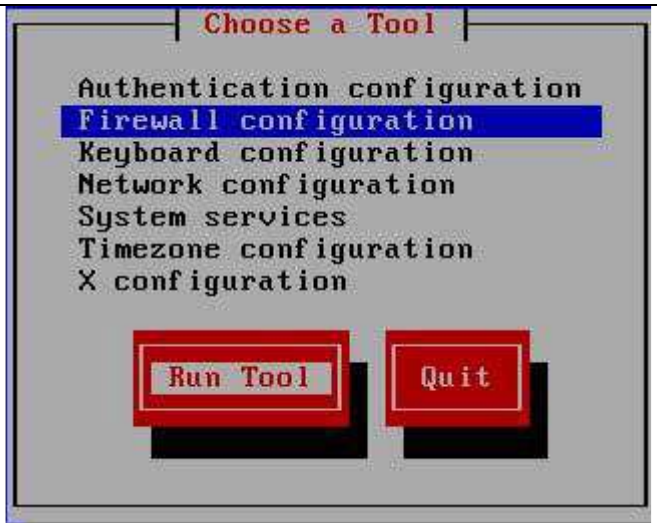
Check firewall status

Firewall is the necessary security part of Linux system which is connected to Internet. But in exam we are not going to use Internet so it's good practice to disable it.

To disable firewall run **setup** commands

```
[root@localhost Server]# setup_
```

Now select **firewall configuration** from list and click on **run tool**



Select **disable** and click on **ok** and **quit** to return on command prompt.

System reboot require to take effect so reboot system with **reboot -f** commands

```
[root@Server ~]# reboot -f_
```

Check portmap and xinetd package status

Almost every Linux server needs these two rpm to function properly. First check that these rpm are install or not.

If no rpm is install then install them via **rpm** commands.

```
[root@Server ~]# rpm -qa portmap
portmap-4.0-65.2.2.1
[root@Server ~]# rpm -qa xinetd
xinetd-2.3.14-10.el5
[root@Server ~]# _
```

If you have rpm then check there status via **setup** commands

```
[root@localhost Server]# setup_
```

Now select **system service** from menu

Put a **star** in front the **portmap** service

Now put **star** in front the **xinetd** service

Click on **ok** and select **quit** to come back on command prompt

Now restart these two service.

```
[root@Server ~]# service portmap restart
Stopping portmap:                                          [  OK  ]
Starting portmap:                                          [  OK  ]
[root@Server ~]# service xinetd restart
Stopping xinetd:                                           [  OK  ]
Starting xinetd:                                           [  OK  ]
[root@Server ~]# _
```

To keep on these services after reboot on then via **chkconfig** command

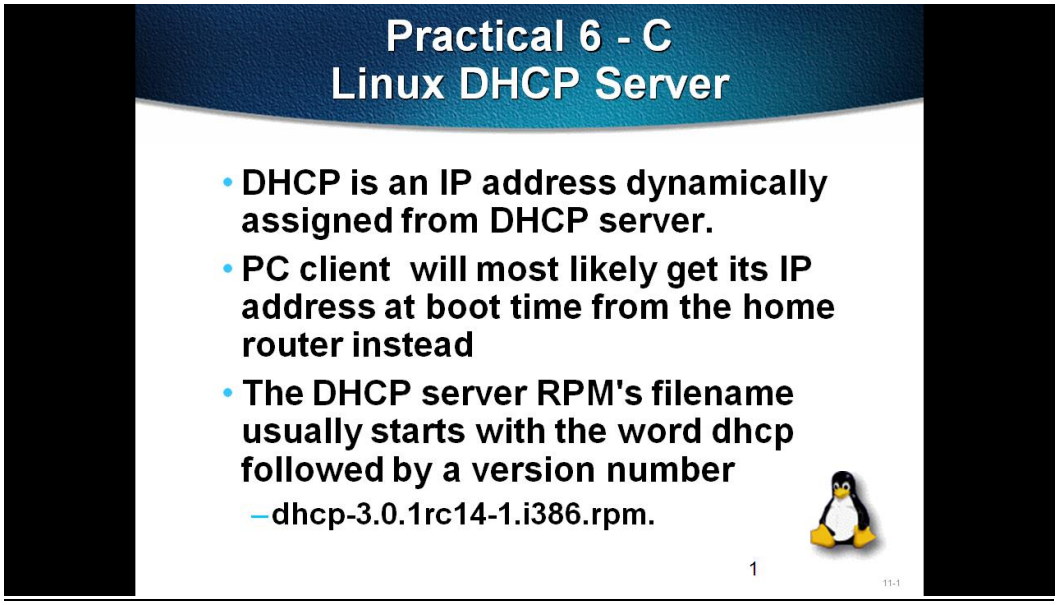
```
[root@Server ~]# chkconfig portmap on
[root@Server ~]# chkconfig xinetd on
[root@Server ~]# _
```

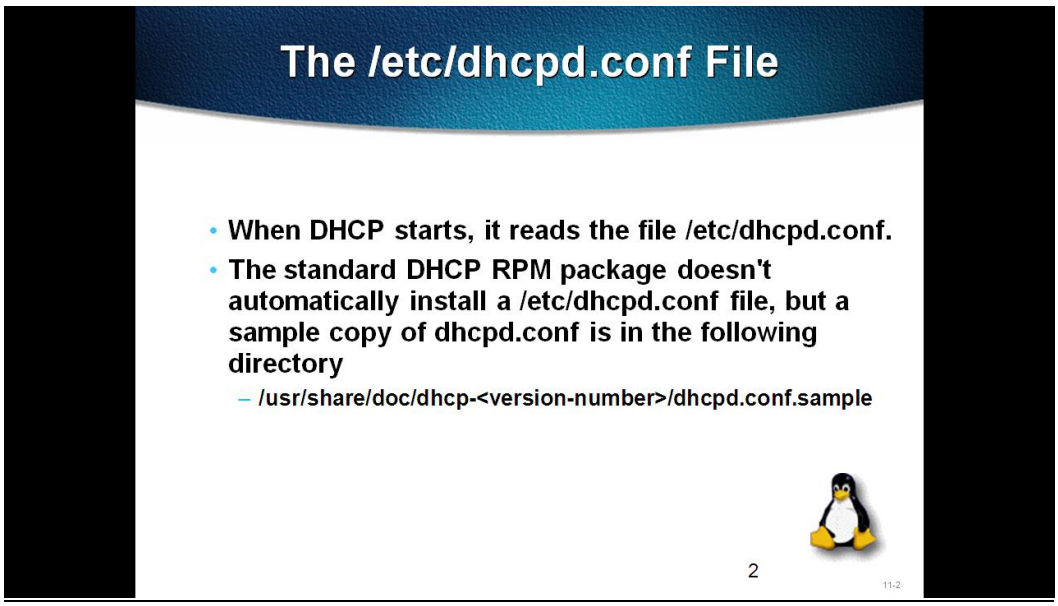
After reboot verify their status. It must be in running condition

```
[root@Server ~]# service portmap status
portmap (pid 3430) is running...
[root@Server ~]# service xinetd status
xinetd (pid 3462) is running...
[root@Server ~]# _
```

Once you have successfully completed these steps you are ready to configure the Linux server.

8.	**Practical 8:** **Assigning Dynamically IP Addresses by configuring DHCP Server**	

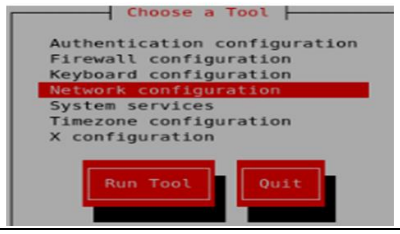




DHCP SERVER

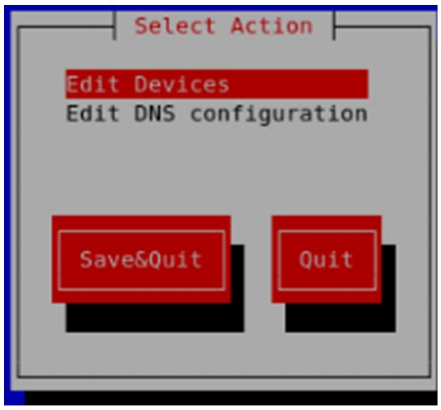
```
root@localhost ~]# Now we are configuring DHCP on Our machine
root@localhost ~]# First we have to check whether DHCP is avaiable
                   on our machine or not that we can check with rpm
                   command
root@localhost ~]# rpm -qa | grep dhcp
   dhcp-4.1.1-12.P1.el6.x86_64
   sblim-cmpi-dhcp-1.0-1.el6.x86_64
root@localhost ~]# hostname
    localhost.localdomain
root@localhost ~]# Now we have to change hostname of our machine as
        server.rkt.com
[root@localhost ~]# setup
```

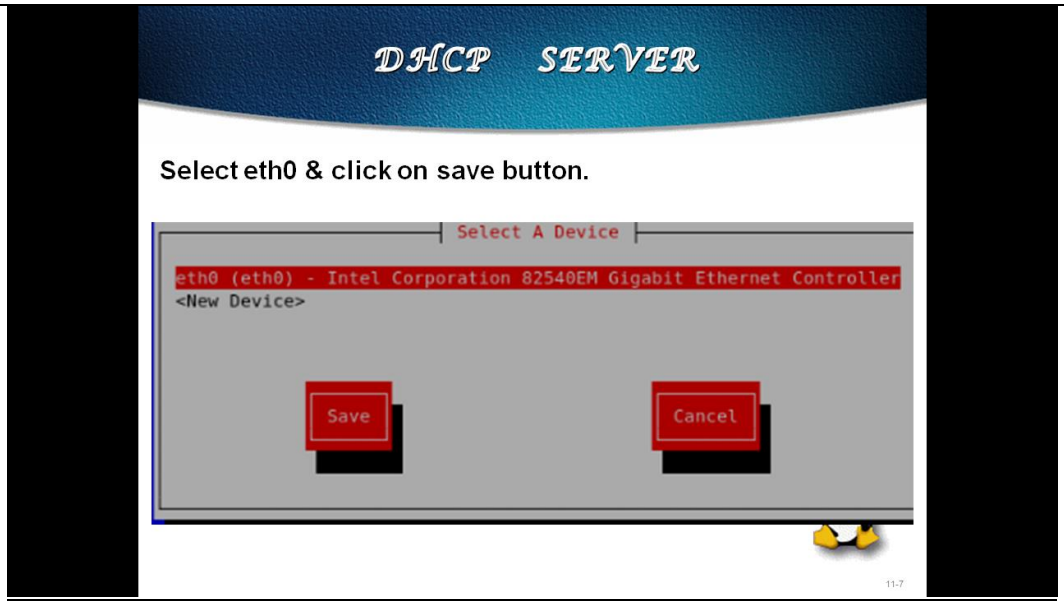
Click on Run Tool button

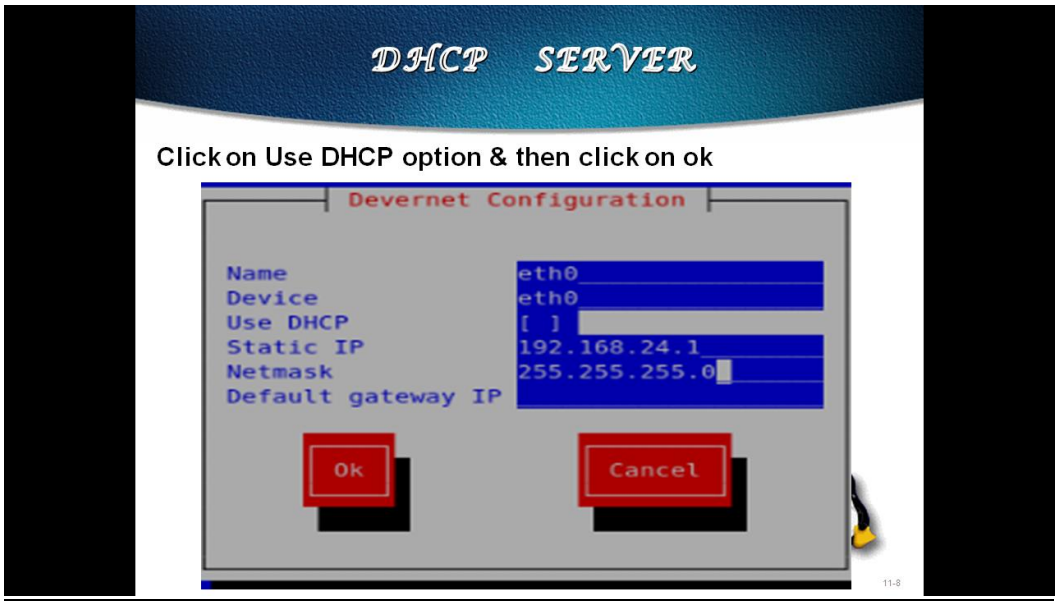


DHCP SERVER

Click on Edit Devices option




DHCP SERVER
Select eth0 & click on save button.
Select A Device
eth0 (eth0) - Intel Corporation 82540EM Gigabit Ethernet Controller
<New Device>
Save
Cancel
11-7


DHCP SERVER
Click on Use DHCP option & then click on ok
Devernet Configuration
Name
Device
Use DHCP
Static IP
Netmask
Default gateway IP
eth0
eth0
[ ]
192.168.24.1
255.255.255.0
Ok
Cancel
11-8

DHCP SERVER

```
root@localhost ~]# service network restart

root@localhost ~]# vi /etc/dhcp/dhcpd.conf
```

dhcpd.conf not found

```
root@localhost ~]# vi /usr/share/doc/dhcp-
4.1.1/dhcpd.conf.sample

root@localhost ~]# cp /usr/share/doc/dhcp-
4.1.1/dhcpd.conf.sample   /etc/dhcp/dhcp.conf
```

root@localhost ~]# vi /etc/dhcp/dhcpd.conf
In this file we have to do some changes

11-9

DHCP SERVER

- Change option domain-name "example.org" to option domain-name "rkt1.com"
- Change option domain-name-servers ns1.example.org, ns2.example.org; to option domain-name-servers server.rkt1.com;
- Change these lines

```
subnet 10.254.239.0 netmask 255.255.255.224
{
range 10.254.239.10 10.254.239.20;
option routers rtr-239-0-1.example.org, rtr-239-0-2.example.org
}
```

To following lines after changes

```
subnet 198.168.1.0 netmask 255.255.255.0
{
range 198.168.1.100 198.168.1.200;
option routers 192.168.1.1;
}
```

- Save this file.

11-10

DHCP SERVER

root@localhost ~]# service dhcpd start

root@localhost ~]# chkconfig dhcpd on

DHCP Server is configured properly, Now we will switch to DHCP Client.

11-11

DHCP CLIENT

DHCP CLIENT

[root@localhost network-scripts]# Now we are on client machine & we we will check whether through dhcp, ip address can be given to our machine or not Before that we have to check currently our machine is configured on mannual or dhcp

Through wizard we will check on network

Right click on Network icon at right top corner on desktop --> Edit Connection --> Select system eth0

--> click on Edit Button --> select IPV4 Setting option --> see the method is manual

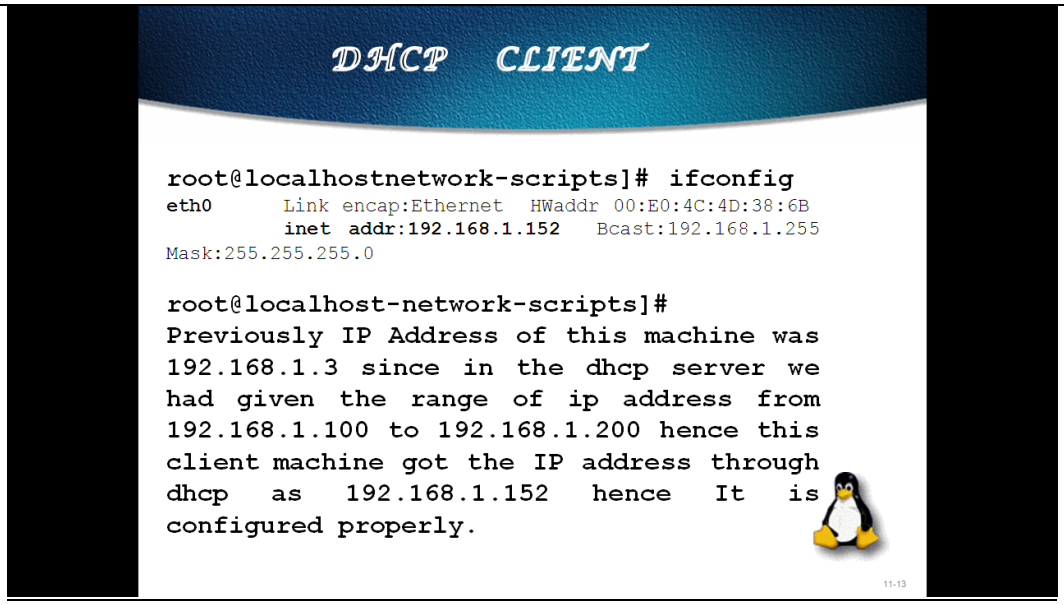
```
root@localhost network-scripts]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:E0:4C:4D:38:6B
          inet addr:192.168.1.3  Bcast:192.168.1.255  Mask:255.255.255.0
```

root@localhost network-scripts]# vi /etc/sysconfig/network-scripts/ifcfg-eth0

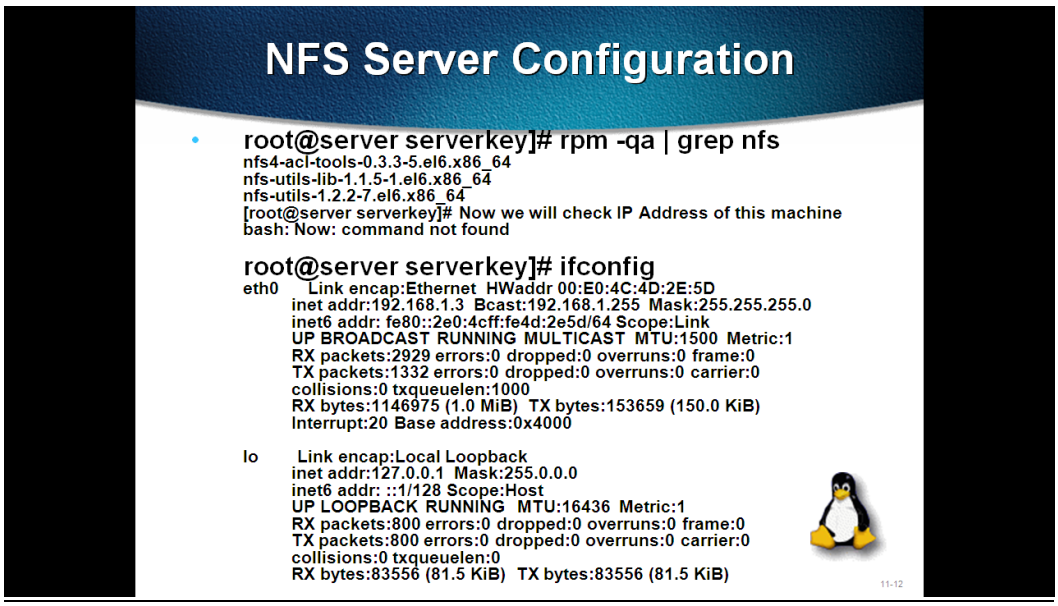
Change BOOTPROTO = dhcp

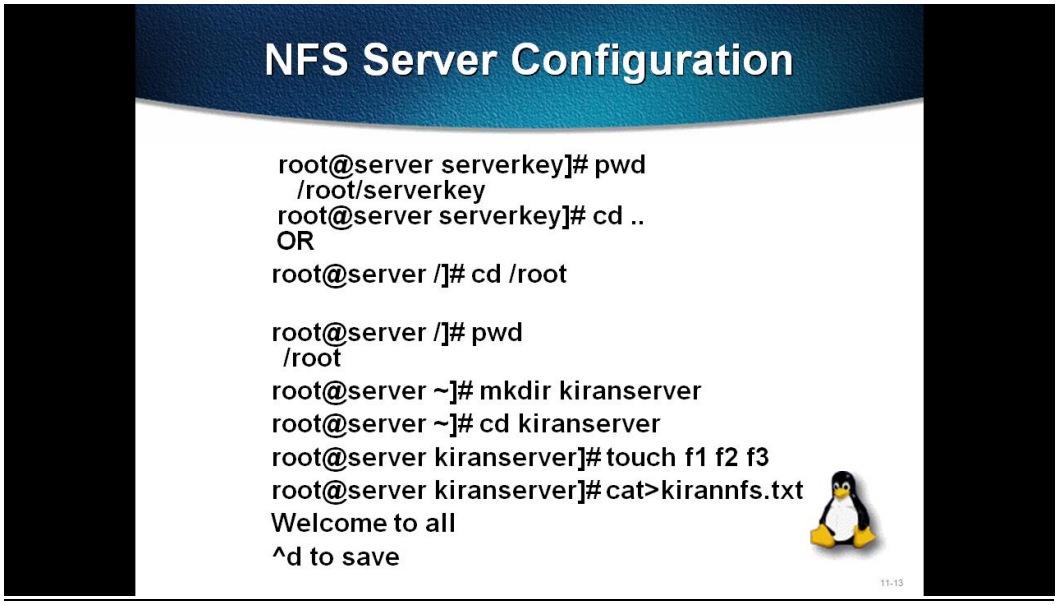
root@localhost network-scripts]# service network restart

11-12


DHCP CLIENT
root@localhostnetwork-scripts]# ifconfig
eth0 Link encap:Ethernet HWaddr 00:E0:4C:4D:38:6B
inet addr:192.168.1.152 Bcast:192.168.1.255
Mask:255.255.255.0
root@localhost-network-scripts]#
Previously IP Address of this machine was 192.168.1.3 since in the dhcp server we had given the range of ip address from 192.168.1.100 to 192.168.1.200 hence this client machine got the IP address through dhcp as 192.168.1.152 hence It is configured properly.
11-13

9.	**Practical 9:** **Setting up NFS File Server**	





NFS Server Configuration

- root@server kiranserver]# vi /etc/exports
- /root/kiranserver *(rw,sync)
- File → save

root@server kiranserver]# service nfs start

- root@server kiranserver]# service nfs restart

```
Shutting down NFS mountd:                  [  OK  ]
Shutting down NFS daemon:                  [  OK  ]
Shutting down NFS quotas:                  [  OK  ]
Shutting down NFS services:                [  OK  ]
Starting NFS services:                     [  OK  ]

Starting NFS quotas:                       [  OK  ]
Starting NFS daemon:                       [  OK  ]
Starting NFS mountd:                       [  OK  ]
```

11-14

NFS Server Configuration

- root@server kiranserver]# showmount -e 192.168.1.3

```
Export list for 192.168.1.3:
/root/kiranserver *
/root/serverkey   *
/root/mkey        *
/ravi             *
/home/kk          *
root@server kiranserver]# service iptables stop
root@server kiranserver]# service iptables status
iptables: Firewall is not running.
```

- root@server kiranserver]# chmod -R 777 /root/kiranserver

```
root@server kiranserver]# service vsftpd stop
Shutting down vsftpd:
```

- root@server kiranserver]# service vsftpd status

```
vsftpd is stopped
```

11-15

NFS Server Configuration

Now our NFs is configured on server, we can share kiranserver folder or directory on client machine,

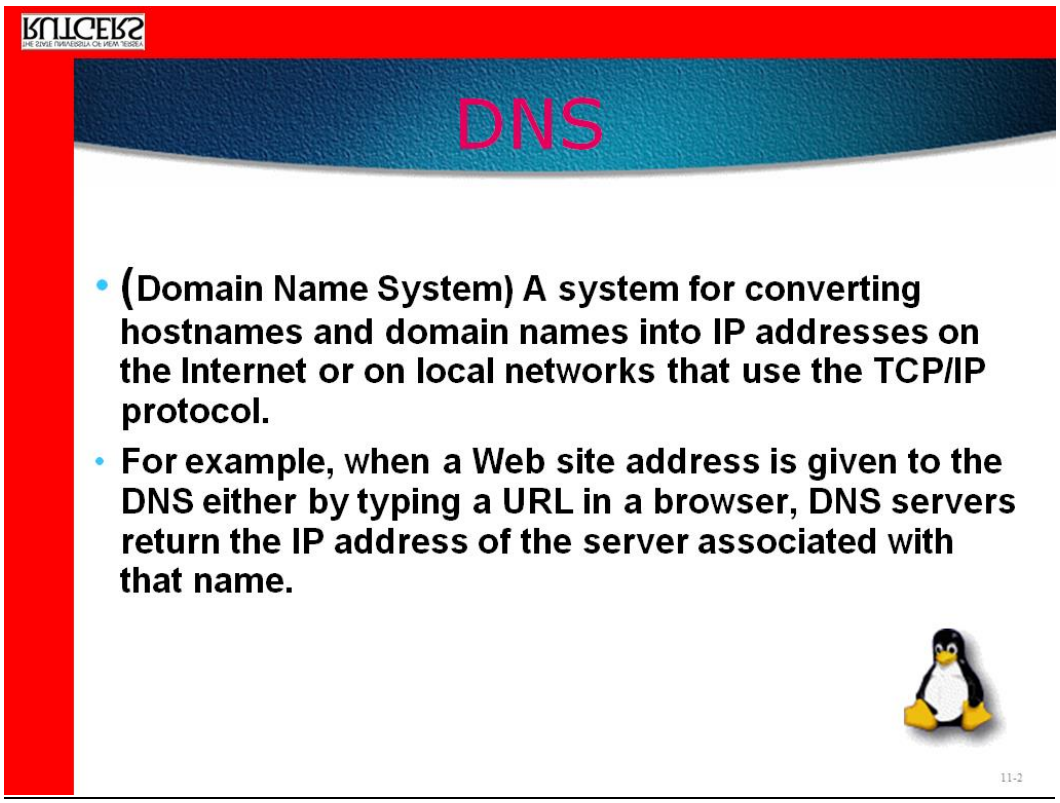
Now we will switch to client machine for sharing

11-16

NFS Client Configuration

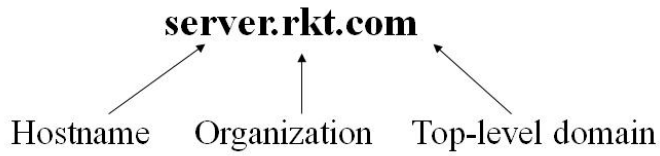
```
root@server ]# pwd
/root
root@server ]# cd /home
root@server ]# pwd
/home
root@server ]# service iptables stop
root@server ]# service vsftpd stop
root@server home ]# mkdir nfsclient
root@server ]# mount –t nfs 192.168.1.3:/root/kiranserver
  /home/nfsclient

root@server ]# cd /home/nfsclient
root@server ]# ls
        kirannfs.txt  f1 f2 f3
```

11-17

10.	**Practical 10:** **Creation of Any Domain Name System**	
		

Networking Basics: DNS

- **IP addresses are usually paired with more human-friendly names: Domain Name System (DNS).**

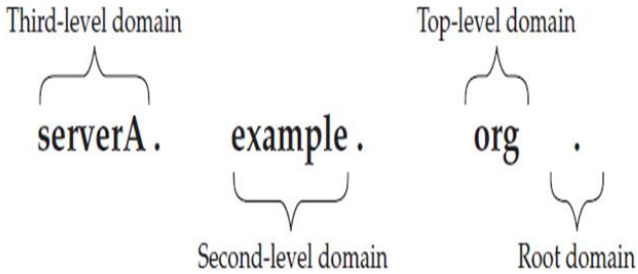


- Other top-level domains include .com, .gov, .org, etc. There are also country-specific domains like .uk, .ca, .jp, etc.

11-3

Example of domain

serverA.example.org.in



11-4

Types of Servers used in DNS

DNS servers come in three flavors: primary, secondary, and caching.

Another special class of name servers consists of the so-called "root name servers."

Primary servers *are the ones considered authoritative for a particular domain.*

authoritative server *is the one on which the domain's configuration files reside.*

Secondary servers *work as backups and as load distributors for the primary name servers.*

Primary servers know of the existence of secondaries and send them periodic updates to the name tables.

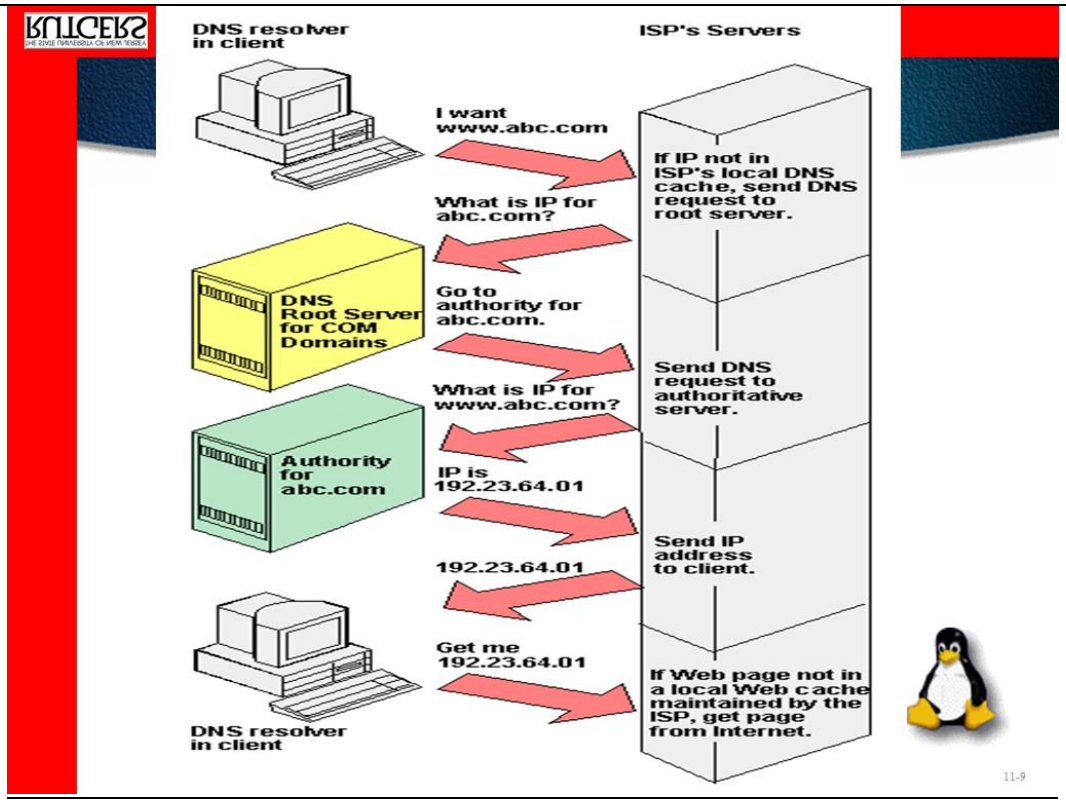
11-6

Types of Servers Contd…

Caching servers *are just that: caching servers. They contain no configuration files for* any particular domain.

when a client host requests a caching server to resolve a name, that server will check its own local cache first. If it cannot find a match, it will find the primary server and ask it. This response is then cached. Practically caching servers work quite well because of the temporal nature of DNS requests.

11-7



DNS FLOW

1.Normally Ip is group of number not easy to remember , instead we use names DNS (Domain Name Server).

server.abc.edu.**in**

in → Root domain

edu → Top level domain.

abc →second level domain

server → third Level domain

2.When you type hostname. It will use resources to resolve hostname to IP Address for that it uses the file /etc/nswitch.conf, nswitch is a name server search.

nswitch has hosts line which indicate first DNS has to search in local file i.e /etc/hosts

11-10

DNS FLOW Contd…

3.It will check in Local DNS /etc/hosts (IP addr & hosts – hostname) If domain name is available in /etc/hosts then it resolve its IP Address Else move to resolve.conf

4. If not available then it will search in /etc/resolve.conf
This file contains → search server.rkt.com
nameserver 192.168.1.3
(In this, advisible maximum three different nameservers can be defined. If one is not reachable it may search for 2nd after 3rd it will stop searching.)

11-11

DNS FLOW Contd…

5. We have to install BIND DNS Server → Berkeley Internet Name Daemon
→ Yum install Bind*

7. Then it will detect the type of server from 3 types of servers (Master, slave & cache server)

8. BIND Server (Berkeley Internet Name Daemon) will not have direct access to /etc, hence it is available /var/named/chroot.

11-12

DNS FLOW Contd…

9. Before beginning to configure your server you have to identify number of files depends on BIND server, we need 5 files.

10. 5 files are needed to setup named server, out of that 3 files are required for Master, Slave & Caching servers & remaining two files are used by Master Server.

11. Three files are
    named.conf
    named.ca
    named.local

11-13

DNS FLOW Contd…

12. named.conf is available in /etc.
    This file stores addresses of other configuration files, also it has global properties & source of configuration files.
13. named.ca is available in /var/named directory.
    It stores names & addresses of Root Server.
14. named.local is available in /var/named directory.
    This file stores information for resolving loopback address for the localhost.
15. Master needs two more files, They are called zone files.
    zone ➔ It stores names & addresses of servers & workstation in the local domain & map names to IP addresses.

11-14

DNS FLOW Contd…

reverse.zone → It stores IP Addresses & names.
It maps IP Address to nameserver (ie. Domain name)
ie. This file provides information to map IP Addresses to names.

named.ca file

The first zone file is known as the cache file, and it references a file called named.ca, which contains information about the world's root name servers. This information changes and needs to be updated periodically.

11-15

Sample Master named.conf

- listings in /etc/named.conf shown here

These zone statements refer to files that are called zone files.

Additional options for zone statements exist.

Each zone statement begins with the word zone followed by the domain name and the data class.

- The four data classes are in, hs, hesiod, and chaos. If no type is specified, the default is in, for Internet.

```
1. options {
directory "/var/named";
dump-file "/var/named/data/cache_dump.db"};
2. controls {
inet 127.0.0.1 allow { localhost; } keys { rndckey; };
};
3. zone "." {
        type hint;
        file "named.ca";
};
4. zone "0.0.127.in-addr.arpa" {
        type master;
        file "named.local";
        allow-query {any;};
};
5. zone "iitk.ac.in" {
        type master;
        file "hosts.db";
        allow-query {any;};
};
6. zone "95.200.203.IN-ADDR.ARPA" {
        type master;
        file "hosts.rev.203.200.95";
        allow-query {any;};
};
7. zone "iitk.ernet.in" {
        type slave;
        file "hosts.iitk.ernet.in";
        masters { 202.141.40.10; };
        allow-query {any;};
```

SOA (START OF AUTHORITY)

SOA (START OF AUTHORITY)

The start of authority (SOA) is the first line in the zone file. The SOA identifies the name server as the authoritative source for information about this domain. Each zone file has only one SOA, and it contains the following data:

@ IN SOA main.tactechnology.com. mail.tactechnology.com. OR
@ IN SOA localhost root (
42 ; serial
3H ; refresh
15M ; retry
1W ; expiry
1D) ; minimum
IN NS localhost
localhost IN A 127.0.0.1 };

11-31

SOA (START OF AUTHORITY)

■■ **Refresh** — The amount of time the server should wait before refreshing its data.

■■ **Retry** — The amount of time the server should wait before attempting
to contact the primary server if the previous attempt failed.

■■ **Expire** — Means that if the secondary master is unable to contact a primary master during the specified period, the data expires and should be purged.

■■ **TTL** — Specifies the time to live for the data. This parameter is intended for caching name servers and tells them how long to hold the data in their cache.

11-34

Records for DNS

Configuring BIND: The Domain Name System 459

All of the information contained by the SOA may be placed on one line, but it is usually written as shown previously. The order of the items is significant in the SOA header. Following the SOA header information are lines containing additional server information. Two of these lines, containing the abbreviations NS and A are shown in Listing 20-2. These abbreviations are explained here:

- **NS** — Name servers in this domain.
- **A**— The IP address for the name server.
- **PTR** — Pointer for address name mapping.
- **CNAME** — Canonical name, the real name of the host.
- **MX** — The mail exchange record. The MX record specifies the mail

11-35

Records for DNS

servers for the domain. If more than one MX server is present priority is determined by the address with the lowest number receiving the highest priority.

- **TXT** — Text information. You can enter descriptive information here.
- **WKS** — Well-known service. You can enter descriptive information here.
- **HINFO** — Host Information usually shows type of hardware and software.

11-36

DNS Configuration

1)[root@server ~]# ifconfig

IP 192.168.1.3

Ethernet card number

eth0 ➔ Ethernet card number

2) [root@server ~]# Vim /etc/sysconfig/network-scripts/ifcfg-eth0

DNS1 = 192.168..1.3 → type DNS IP addr same as machine IP addr

IP: 192.168.1.3

Save file

3) [root@server ~]# vim /etc/hosts

Add in this file

IP addr of current m/c | Hostname | Hostname

192.168.1.3 server.rkt.com server

Domain name which we want to create

Save this file

4) [root@server ~]# vi /etc/sysconfig/network

networking = yes

Change this line ➔ Default is hostname = localhost.localdomain

hostname = server.rkt.com

Save this file

5) [root@server ~]# vi /etc/resolv.conf

Type the following :-

Domain name which we want to create

search rkt.com

nameserver 192.168.1.3 — IP addr of current m/c

Save this File

6) [root@server ~]# service network restart

7) [root@server ~]# yum install bind*

(Now system is Registered with RHN – Red Hat Network)

[root@server ~]# vi /etc/named.conf

Options

{

listen – on port 53{ 192.168.1.3; } ; — Change this from 127.0.0.1 to current machine IP addr

comment this line by # ➔ listen – on –v6 port 53 { : : 1 ; } ;

allow-query { any ; } ; — Change from localhost to any

check & notedown this filename for master zone ➔ **include " /etc/named.rfc1912.zones ";**

Save this File

8) [root@server ~]# vi /etc/named.rfc1912.zones

Change localhost.localdomain with rkt.com

zone "rkt.com " IN {

type master ;

Change named.localhost with forward.zone

file " forward.zone" ;
allow-update { none ; } ;
} ;

Change 1.0.0.127 with 1.168.192.

zone " 1.168.192. in-addr.arpa " IN {
type master ;

Change named.loopback with reverse.zone

file " reverse.zone " ;
allow-update { none ; } ;
Save this File

9)[root@server ~]# cd /var/named/
➔ (It has
named.ca,named.localhost,named.loopback,
named.empty,forward.zone,reverse.zone,chroot,
dynamic,data,slaves)

10)[root@server named]# cp named.localhost forward.zone
11)[root@server named]# cp named.loopback reverse.zone

12)[root@server named]# vi forward.zone

$TTL 1D

Change @ to server.rkt.com.

@ IN SOA server.rkt.com. root.server.rkt.com. {
}
IN NS server.rkt.com.

Change rname.invalid. to server.rkt.com.

Change NS @ to IN NS server.rkt.com.

Server IN A 192.168.1.3

Change A 127.0.0.1 to Server IN A 192.168.1.3

hostname

Delete this line ➔ (AAAA : : 1)
Save this File

13)[root@server named]# vim reverse.zone

$TTL 1D

Change @ to server.rkt.com.

@ IN SOA server.rkt.com. root.server.rkt.com. {

Change rname.invalid. to root.server.rkt.com.

IN NS server.rkt.com.

Change NS @ to IN NS server.rkt.com.

Delete this line ➔ A 127.0.0.1
Delete this line ➔ AAAA :: 1

3 IN PTR server.rkt.com.

Change PTR localhost to 3 IN PTR server.rkt.com.

Last digit of IP address
of Machine 192.168.1. **3**
Save this File

14)[root@server named]# chgrp named forward.zone

15)[root@server named]# chgrp named reverse.zone

16)[root@server named]# service named start

To check DNS is working properly or not type dig command

17)[root@server named]# dig server.rkt.com

In this command we are writing IP addr & It will resolve domain name

;; ->>HEADER<<- opcode: QUERY**:, status: NOERROR**, id: 55281
;; flags: qr rd ra; **QUERY: 1**, **ANSWER: 1**, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;server.rkt.com. 86400 IN A

;;ANSWER SECTION:
;server.rkt.com. 86400 IN A 192.168.1.3

In this command we are writing IP addr & It will resolve domain name

;; AUTHORITY SECTION:
rkt.com. 86400 IN NS server.rkt.com.

18)[root@server named]# dig -x 192.168.1.3

```
;; QUESTION SECTION:
;  3.1.168.192. in-addr.arpa. IN    PTR

;; ANSWER SECTION:
   3.168.192.in-addr.arpa.    86400 IN    NS
      server.rkt.com.

;; AUTHORITY SECTION:
   rkt.com        86400    IN    NS    server.rkt.com
```

19)[root@server named]# nslookup

>server.rkt.com
server : 192.168.1.3
Address : 192.168.1.3 #53

Type server.rkt.com

name : server.rkt.com
Address : 192.168.1.3
> 192.168.1.3
Server : 192.168.1.3
Address : 192.168.1.3 #53
3.1.168.192.in-addr.arpa name= server.rkt.com

Type 192.168.1.3

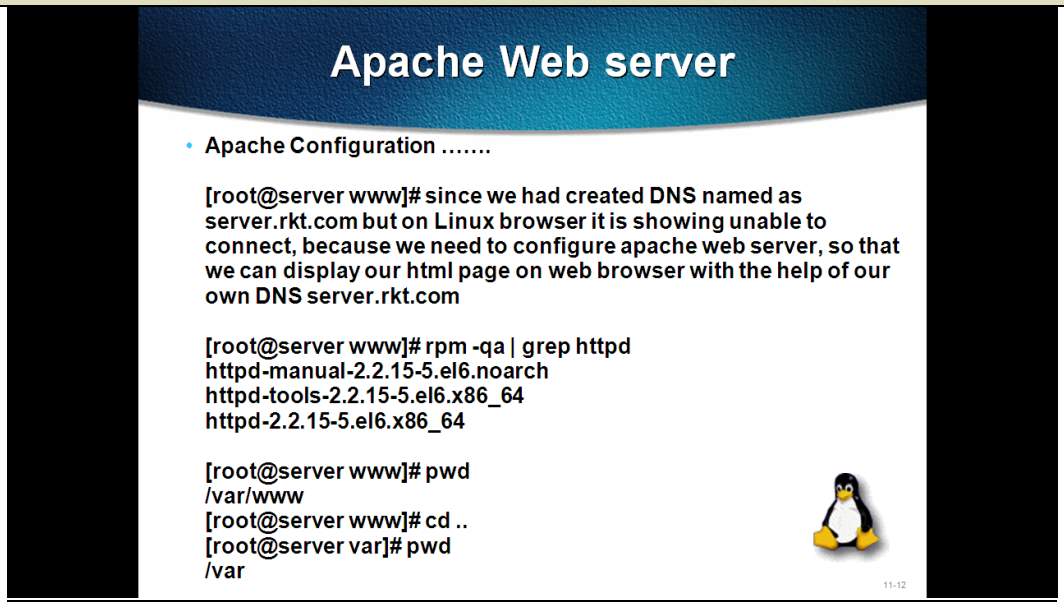
> exit

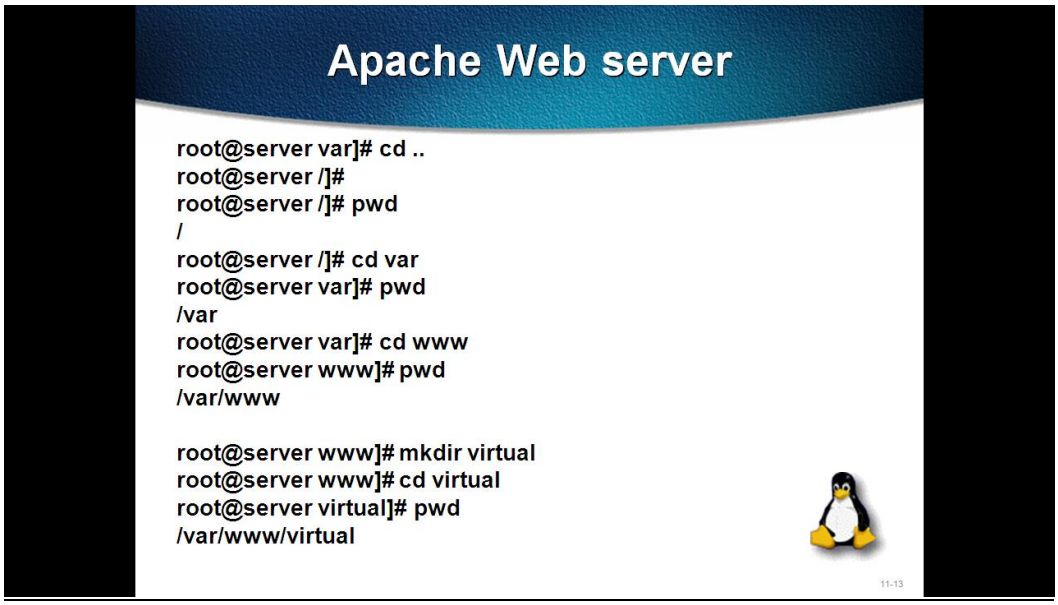
Type exit

Open Mozilla Web Browser & type on Address Bar➔ `server.rkt.com` It will provide msg

➔ Unable to connect
Therfore, we have to configure apache web server to display index.html file with our DNS

11. Practical 11:
The Apache web Server





Apache Web server

```
[root@server virtual]# mkdir www.rkt.com
[root@server virtual]# cd www.rkt.com
[root@server www.rkt.com]# pwd
/var/www/virtual/www.rkt.com
[root@server www.rkt.com]# mkdir html
[root@server www.rkt.com]# pwd
/var/www/virtual/www.rkt.com
[root@server www.rkt.com]# cd html

[root@server html]# vi index.html
<html>
<head>
<title> This is the First page on apache on our domain </title>
</head>
<body>
Welcome to the First page created on apache web server & can be
viewed with our domain created.
</body>
</html>
```

11-14

Apache Web server

- root@server html]# ls
  index.html
  root@server html]# Now we have our DNS is working or not, It can be done with the help of dig commnad for using dig command we need to know the IP Address of our machine hence with ifconfig

```
 root@server html]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:E0:4C:4D:2E:5D
          inet addr:192.168.1.3  Bcast:192.168.1.255
Mask:255.255.255.0
```

  root@server html]# now we will use dig command to check our DNS server.rkt.com is running or not

```
root@server html]# dig -x 192.168.1.3
```

11-15

Apache Web server

root@server html]# vim /etc/httpd/conf/httpd.conf

Add at the end of file, copy these block of lines

```
<VirtualHost *:80>
    ServerAdmin webmaster@dummy-host.example.com
    DocumentRoot /www/docs/dummy-host.example.com
    ServerName dummy-host.example.com
    ErrorLog logs/dummy-host.example.com-error_log
    CustomLog logs/dummy-host.example.com-access_log common
</VirtualHost>
```

Change to following lines :-

```
<VirtualHost *:80>
    ServerAdmin root@www.rkt.com
    DocumentRoot /var/www/virtual/www.rkt.com/
    ServerName www.rkt.com
    ErrorLog logs/rkt.com-error_log
    CustomLog logs/rkt.com-access_log common
</VirtualHost>
```

Apache Web server

2) Remove the comment of following line in httpd.conf file

NameVirtualHost *:80

3) Change the Line

Document Root "/var/www/html" to

Document Root "/var/www/virtual/www.rkt.com/html"

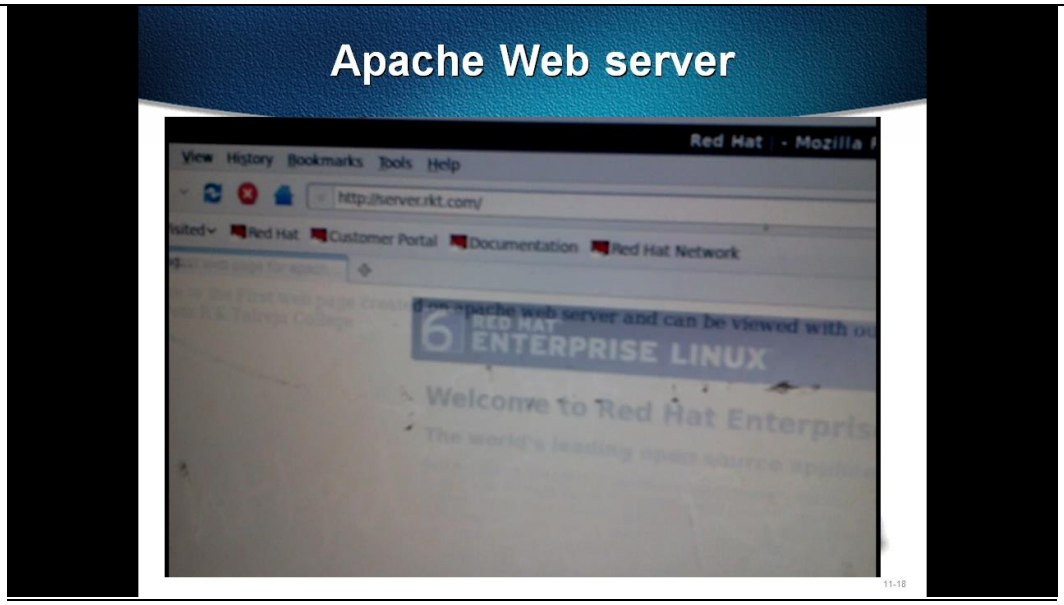
4) Save the Changes.

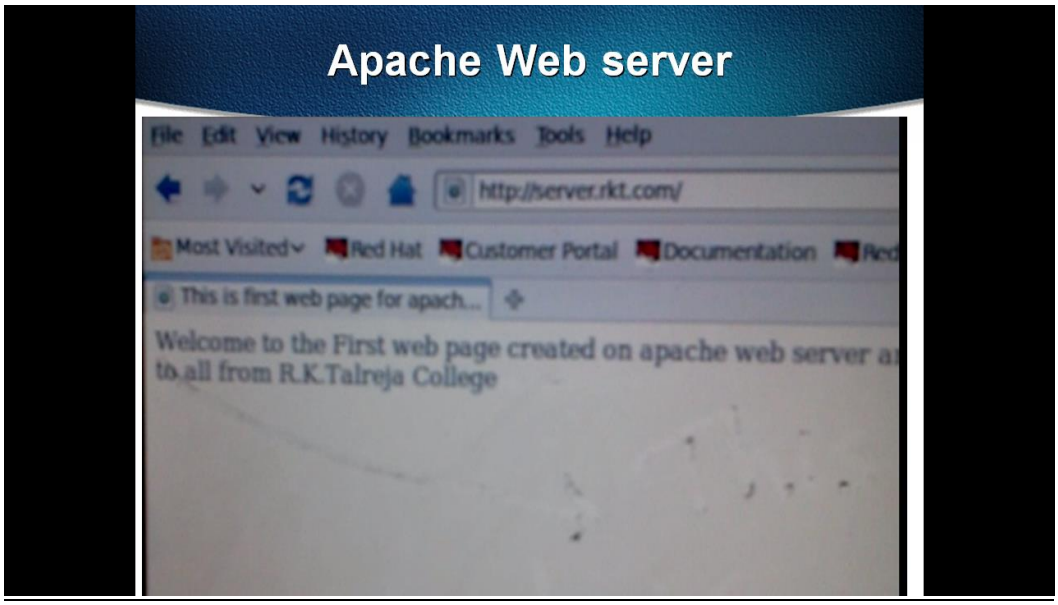
root@server html]# service httpd start

Starting httpd: [OK]

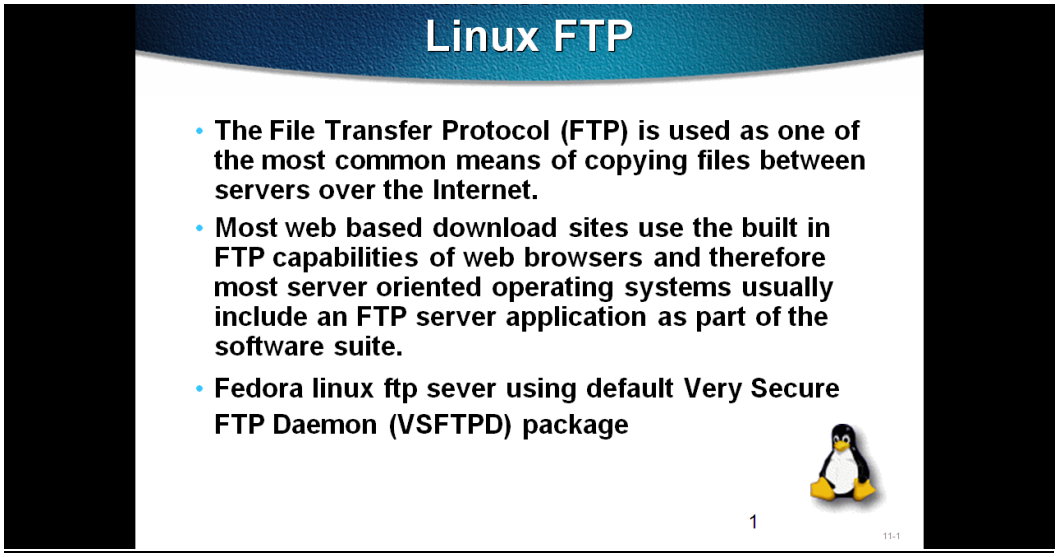
root@server html]# chmod -R 777 /var/www/virtual/www.rkt.com/html

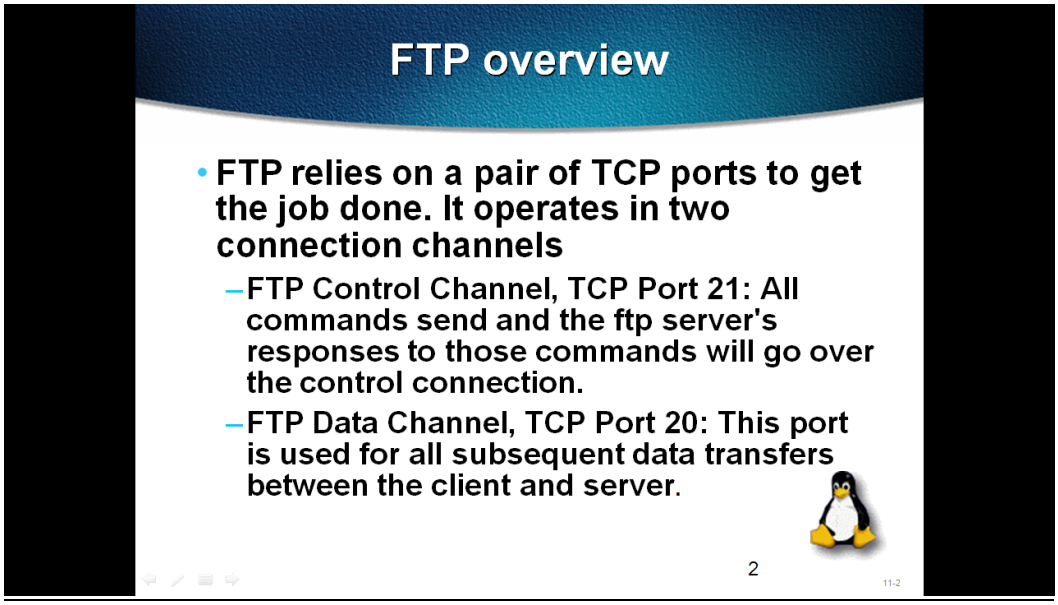
11-17

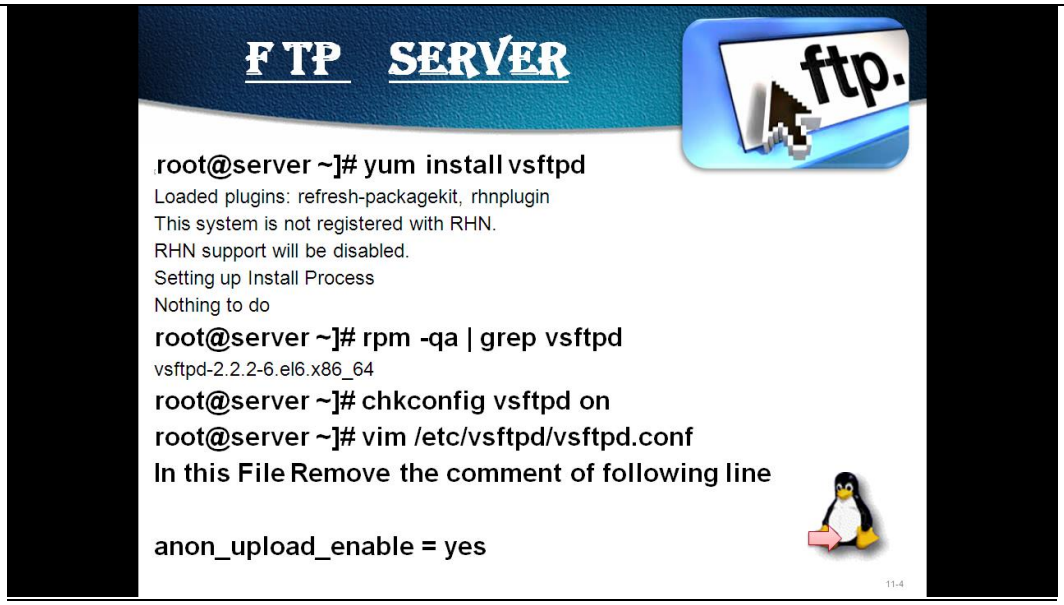

Apache Web server
View History Bookmarks Tools Help
http://server.rkt.com/
Red Hat
Customer Portal
Documentation
Red Hat Network
RED HAT
ENTERPRISE LINUX
Welcome to Red Hat Enterpris

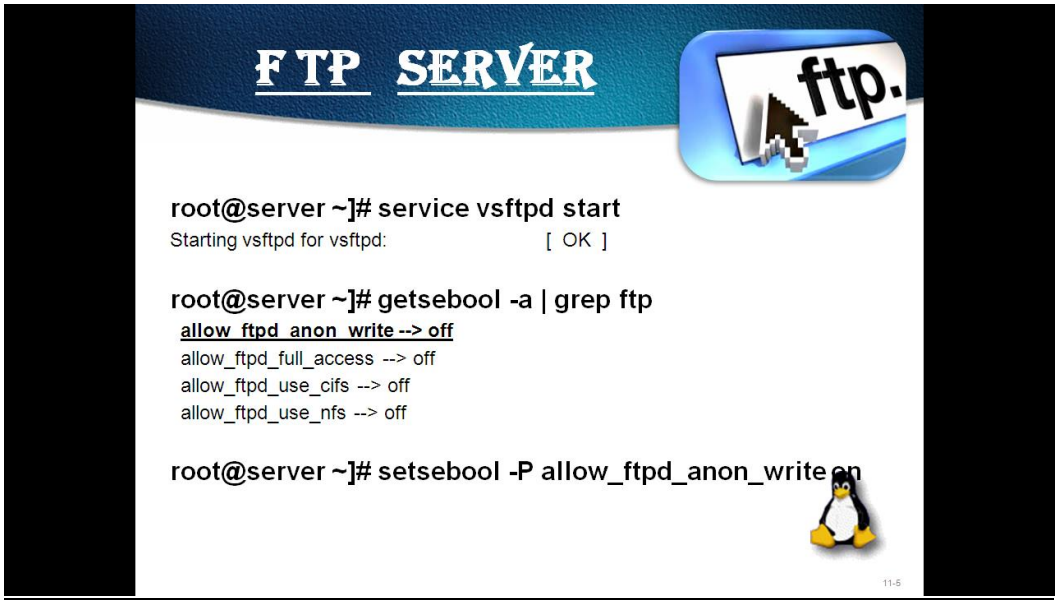

Apache Web server
File Edit View History Bookmarks Tools Help
http://server.rkt.com/
Most Visited
Red Hat
Customer Portal
Documentation
This is first web page for apach...
Welcome to the First web page created on apache web server
to all from R.K.Talreja College

12.	**Practical 12:** **Setting up FTP Server**	






FTP SERVER
ftp.
root@server ~]# yum install vsftpd
Loaded plugins: refresh-packagekit, rhnplugin
This system is not registered with RHN.
RHN support will be disabled.
Setting up Install Process
Nothing to do
root@server ~]# rpm -qa | grep vsftpd
vsftpd-2.2.2-6.el6.x86_64
root@server ~]# chkconfig vsftpd on
root@server ~]# vim /etc/vsftpd/vsftpd.conf
In this File Remove the comment of following line
anon_upload_enable = yes
11-4


FTP SERVER
ftp.
root@server ~]# service vsftpd start
Starting vsftpd for vsftpd: [OK]
root@server ~]# getsebool -a | grep ftp
allow_ftpd_anon_write --> off
allow_ftpd_full_access --> off
allow_ftpd_use_cifs --> off
allow_ftpd_use_nfs --> off
root@server ~]# setsebool -P allow_ftpd_anon_write on
11-5

FTP Server

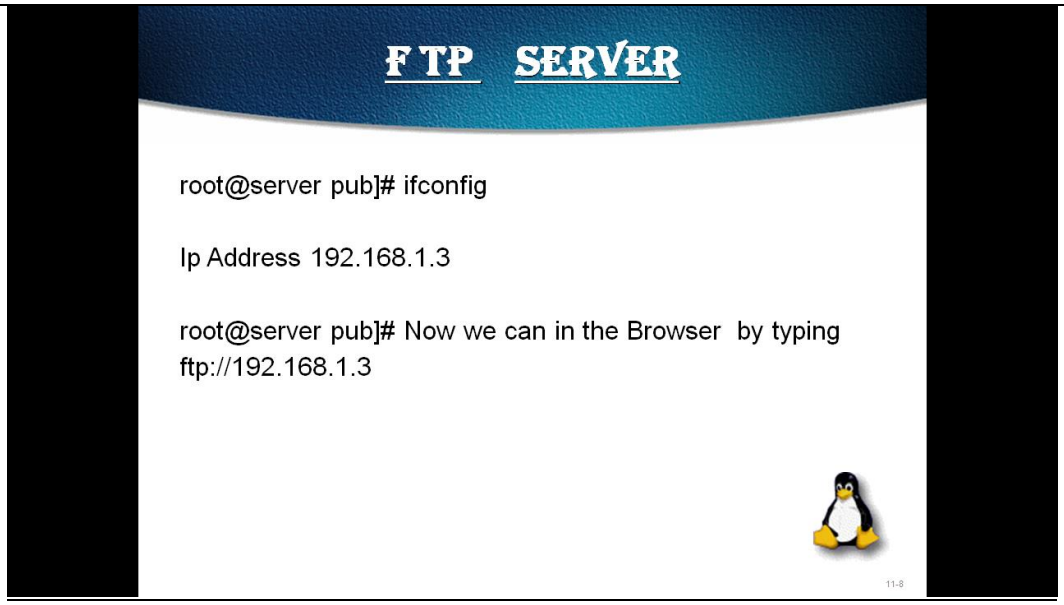
```
root@server ~]# getsebool -a | grep ftp
allow_ftpd_anon_write --> on
allow_ftpd_full_access --> off
allow_ftpd_use_cifs --> off
allow_ftpd_use_nfs --> off

root@server ~]# ls -ldZ /var/ftp/pub
drwxr-xr-x. ftp ftp system_u:object_r:public_content_rw_t:s0 /var/ftp/pub

root@server ~]# chgrp ftp /var/ftp/pub
root@server ~]# chown ftp /var/ftp/pub
root@server ~]# ls -ldZ /var/ftp/pub
drwxr-xr-x. ftp ftp system_u:object_r:public_content_rw_t:s0 /var/ftp/pub
root@server ~]# cd /var/ftp/pub
root@server pub]# ls
```

11-6

FTP SERVER


```
root@server pub]# touch file ( creating empt
root@server pub]#  cat > kiranftp.txt
        Welcome to all
        ^D
root@server pub]# ls
 File   kiranftp.txt
root@server pub]# cd /var/ftp/pub
root@server pub]# chown ftp /var/ftp/pub
root@server pub]# ls -ldZ /var/ftp/pub
drwxr-xr-x. ftp ftp system_u:object_r:public_content_rw_t:s0
/var/ftp/pub
root@server pub]# service vsftpd restart
Shutting down vsftpd:                      [ OK ]
Starting vsftpd for vsftpd:                [ OK ]
```

11-7



You can use get & put commands also on ftp prompt.

13. Practical 13:
Firewall & Security Configuration

Firewalls

Information security is commonly thought of as a process and not a product. However, standard security implementations usually employ some form of dedicated mechanism to control access privileges and restrict network resources to users who are authorized, identifiable, and traceable. Red Hat Enterprise Linux includes several tools to assist administrators and security engineers with network-level access control issues.

Firewalls are one of the core components of a network security implementation.

Netfilter and IPTables

The Linux kernel features a powerful networking subsystem called Netfilter. The Netfilter subsystem provides stateful or stateless packet filtering as well as NAT and IP masquerading services. Netfilter also has the ability to mangle IP header information for advanced routing and connection state management. Netfilter is controlled using the iptables tool.

IPTables Overview

T he power and flexibility of Netfilter is implemented using the iptables administration tool, a command line tool similar in syntax to its predecessor, ipchains, which Netfilter/iptables replaced in the Linux kernel 2.4 and above.
iptables uses the Netfilter subsystem to enhance network connection, inspection, and processing.
iptables features advanced logging, pre- and post-routing actions, network address translation, and port forwarding, all in one command line interface.

Basic Firewall Configuration

Just as a firewall in a building attempts to prevent a fire from spreading, a computer firewall attempts to prevent malicious software from spreading to your computer. It also helps to prevent unauthorized users from accessing your computer.

In a default Red Hat Enterprise Linux installation, a firewall exists between your computer or network and any untrusted networks, for example the Internet. It determines which services on your computer remote users can access. A properly configured firewall can greatly increase the security of your system. It is recommended that you configure a firewall for any Red Hat Enterprise Linux system with an Internet connection.

Basic Firewall Configuration

Just as a firewall in building attempts to prevent a fire from spreading, a computer firewall attempts to prevent malicious software from spreading to your computer. It also helps to prevent unauthorized users from accessing your computer.

In a default Red Hat Enterprise Linux installation, a firewall exists between your computer or network and any untrusted networks, for example the Internet. It determines which services on your computer remote users can access. A properly configured firewall can greatly increase the security of your system. It is recommended that you configure a firewall for any Red Hat Enterprise Linux system with an Internet connection.

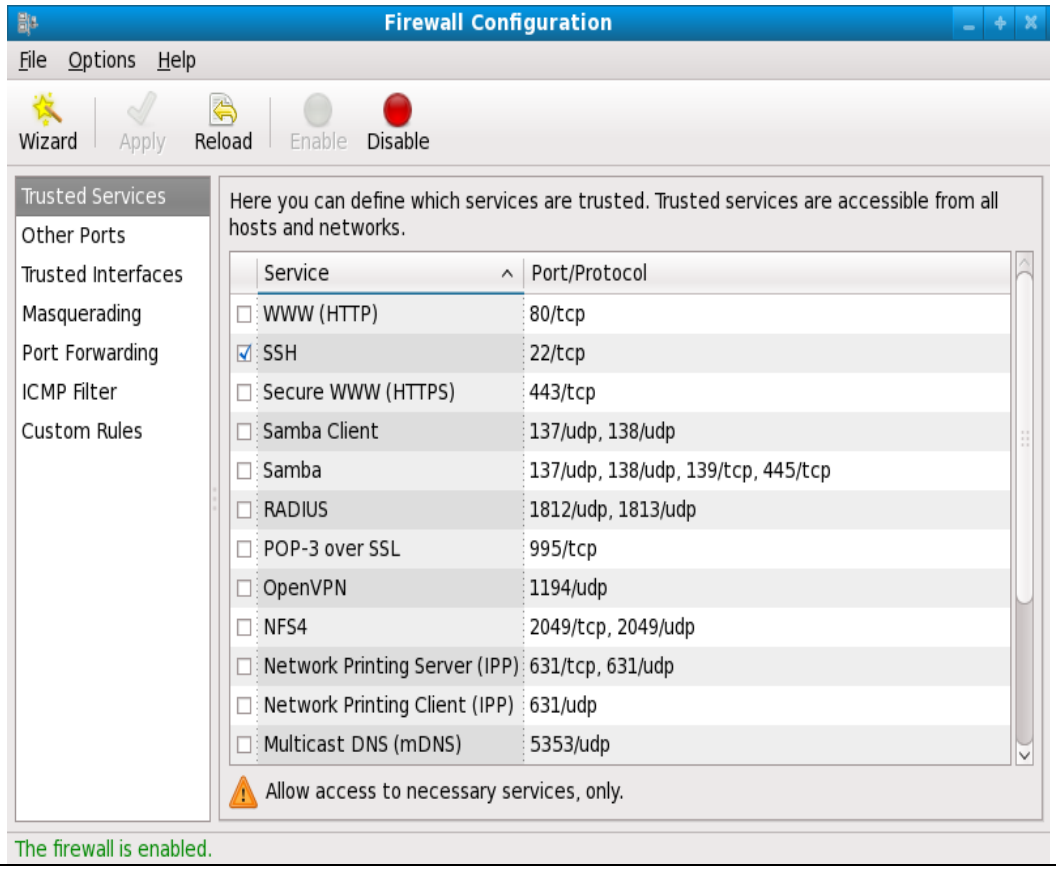
Firewall Configuration Tool

During the Firewall Configuration screen of the Red Hat Enterprise Linux installation, you were given the option to enable a basic firewall as well as to allow specific devices, incoming services, and ports.

After installation, you can change this preference by using the Firewall Configuration Tool.

T o start this application, either select System → Administration → Firewall from the panel, or type

system -config-firewall at a shell prompt.



Enabling and Disabling the Firewall

Select one of the following options for the firewall:
Disabled — Disabling the firewall provides complete access to your system and does no security checking. T his should only be selected if you are running on a trusted network (not the Internet) or need to configure a custom firewall using the iptables command line tool.

Firewall configurations and any customized firewall rules are stored in the
/etc/sysconfig/iptables file. If you choose Disabled and click OK, these configurations and firewall rules will be lost.

Enabled — T his option configures the system to reject incoming connections that are not in response to outbound requests, such as DNS replies or DHCP requests. If access to services running on this machine is needed, you can choose to allow specific services through the firewall.
If you are connecting your system to the Internet, but do not plan to run a server, this is the safest choice.

Trusted Services

Enabling options in the T rusted services list allows the specified service to pass through the firewall.

WWW (HT T P)

The HTT P protocol is used by Apache (and by other Web servers) to serve web pages. If you plan on making your Web server publicly available, select this check box. T his option is not required for viewing pages locally or for developing web pages. T his service requires that the httpd package be installed.
Enabling WWW (HTT P) will not open a port for HTT PS, the SSL version of HTT P. If this service is required, select the Secure WWW (HTT PS) check box.

FT P

The FTP protocol is used to transfer files between machines on a network. If you plan on making your FT P server publicly available, select this check box. T his service requires that the vsftpd package be installed.

SSH

Secure Shell (SSH) is a suite of tools for logging into and executing commands on a remote machine. T o allow remote access to the machine via ssh, select this check box. This service requires that the openssh-server package be installed.

Telnet

Telnet is a protocol for logging into remote machines.Telnet communications are unencrypted and provide no security from network snooping.Allowing incoming Telnet access is not recommended.To allow remote access to the machine via telnet, select this check box. This service requires that the telnet-server package be installed.

Mail (SMTP)

SMTP is a protocol that allows remote hosts to connect directly to your machine to deliver mail.

You do not need to enable this service if you collect your mail from your ISP's server using POP3 or IMAP, or if you use a tool such as fetchm ail. T o allow delivery of mail to your machine, select this check box. Note that an improperly configured SMT P server can allow remote machines to use your server to send spam.

NFS4

T he Network File System (NFS) is a file sharing protocol commonly used on *NIX systems.

Version 4 of this protocol is more secure than its predecessors. If you want to share files or directories on your system with other network users, select this check box.

Samba Samba is an implementation of Microsoft's proprietary SMB networking protocol. If you need to share files, directories, or locally-connected printers with Microsoft Windows machines, select this check box.

Other Ports

T he Firewall Configuration Tool includes an Other ports section for specifying custom IP ports as being trusted by iptables. For example, to allow IRC and Internet printing protocol (IPP) to pass through the firewall, add the following to the Other ports section:

194 :tcp,631:tcp

Saving the Settings

Click OK to save the changes and enable or disable the firewall. If Enable firewall was selected,

the options selected are translated to iptables commands and written to the

/etc/sysconfig/iptables file. T he iptables service is also started so that the firewall is

activated immediately after saving the selected options. If Disable firewall was selected, the /etc/sysconfig/iptables file is removed and the iptables service is stopped immediately.

The selected options are also written to the /etc/sysconfig/system -config-firewall file so

that the settings can be restored the next time the application is started. Do not edit this file by hand.
Even though the firewall is activated immediately, the iptables service is not configured to start automatically at boot time.

Activating the IPTables Service

T he firewall rules are only active if the iptables service is running. T o manually start the service, use the following command as the root user:

~]# service i ptabl es restart
iptables: Applying firewall rules: [OK]

To ensure that iptables starts when the system is booted, use the following command:

~]# chkconfig --l evel 34 5 iptables o n

Using IPTables

T he first step in using iptables is to start the iptables service. Use the following command as the root user to start the iptables service:

~]# service iptables restart
iptables: Applying firewall rules: [OK]

Note
The ip6tables service can be turned off if you intend to use the iptables service only. If you deactivate the ip6tables service, remember to deactivate the IPv6 network also. Never leave a network device active without the matching firewall.
To force iptables to start by default when the system is booted, use the following command as the root user:

~]# chkco nfig --l evel 34 5 iptabl es o n

This forces iptables to start whenever the system is booted into runlevel 3, 4, or 5.

IPTables Command Syntax

The following sample iptables command illustrates the basic command syntax:
iptabl es -A <chain> -j <target>
The -A option specifies that the rule be appended to <chain>. Each chain is comprised of one or more rules, and is therefore also known as a ruleset.
The three built-in chains are INPUT,OUT PUT,and FORWARD. These chains are permanent and cannot be deleted. The chain specifies the

point at which a packet is manipulated.
The -j <target> option specifies the target of the rule; i.e., what to do if the packet matches the rule.
Examples of built-in targets are ACCEPT,DROP, and REJECT .

Basic Firewall Policies

Establishing basic firewall policies creates a foundation for building more detailed, user-defined rules.
Each iptables chain is comprised of a default policy, and zero or more rules which work in concert with the default policy to define the overall ruleset for the firewall.
The default policy for a chain can be either DROP or ACCEPT.Security-minded administrators typically implement a default policy of DROP, and only allow specific packets on a case-by-case basis. For example, the following policies block all incoming and outgoing packets on a network gateway:

~]# iptabl es -P INPUT DROP
~]# iptabl es -P OUT PUTDROP

It is also recommended that any forwarded packets — network traffic that is to be routed from the firewall to its destination node — be denied as well, to restrict internal clients from inadvertent exposure to the Internet. T o do this, use the following rule:

~]# iptables -P FORWARD DROP

When you have established the default policies for each chain, you can create and save further rules for your particular network and security requirements.
T he following sections describe how to save iptables rules and outline some of the rules you might implement in the course of building your iptables firewall.

Saving and Restoring IPTables Rules

Changes to iptables are transitory; if the system is rebooted or if the iptables service is restarted, the rules are automatically flushed and reset. T o save the rules so that they are loaded when the iptables service is started, use the following command as the root user:

~]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[OK]

The rules are stored in the file /etc/sysconfig/iptables and are applied whenever the service is started or the machine is rebooted.

Common IPTables Filtering

Preventing remote attackers from accessing a LAN is one of the most important aspects of network security. T he integrity of a LAN should be protected from malicious remote users through the use of stringent firewall rules.

However, with a default policy set to block all incoming, outgoing, and forwarded packets, it is impossible for the firewall/gateway and internal LAN users to communicate with each other or with external resources. T o allow users to perform network-related functions and to use networking applications, administrators must open certain ports for communication.

For example, to allow access to port 80 on the firewall, append the following rule:

~]# iptables -A INPUT -p tcp -m tcp --dpo rt 80 -j ACCEPT

This allows users to browse websites that communicate using the standard port 80. T o allow access to secure websites (for example, https://www.example.com/), you also need to provide access to port 443, as follows:

~]# iptables -A INPUT -p tcp -m tcp --dpo rt 4 4 3 -j ACCEPT

Important When creating an iptables ruleset, order is important.

If a rule specifies that any packets from the 192.168.100.0/24 subnet be dropped, and this is followed by a rule that allows packets from 192.168.100.13 (which is within the dropped subnet), then the second rule is ignored.

The rule to allow packets from 192.168.100.13 must precede the rule that drops the remainder of the subnet.

To insert a rule in a specific location in an existing chain, use the -I option. For example:

~]# iptables -I INPUT 1 -i lo -p all -j ACCEPT

This rule is inserted as the first rule in the INPUT chain to allow local loopback device traffic.

There may be times when you require remote access to the LAN. Secure services, for example SSH, can be used for encrypted remote connection to LAN services.

Administrators with PPP-based resources (such as modem banks or bulk ISP accounts), dial-up access can be used to securely circumvent firewall barriers. Because they are direct connections, modem connections are typically behind a firewall/gateway.

For remote users with broadband connections, however, special cases can be made. You can configure iptables to accept connections from

remote SSH clients. For example, the following rules allow remote SSH access:

~]# iptables -A INPUT -p tcp --dpo rt 22 -j ACCEPT
~]# iptables -A O UT PUT -p tcp --spo rt 22 -j ACCEPT

These rules allow incoming and outbound access for an individual system, such as a single PC directly connected to the Internet or a firewall/gateway. However, they do not allow nodes behind the firewall/gateway to access these services. T o allow LAN access to these services, you can use Network Address Translation (NAT) with iptables filtering rules.

FORWARD and NAT Rules

Most ISPs provide only a limited number of publicly routable IP addresses to the organizations they serve.
Administrators must, therefore, find alternative ways to share access to Internet services without giving public IP addresses to every node on the LAN. Using private IP addresses is the most common way of allowing all nodes on a LAN to properly access internal and external network services.
Edge routers (such as firewalls) can receive incoming transmissions from the Internet and route the packets to the intended LAN node. At the same time, firewalls/gateways can also route outgoing requests from a LAN node to the remote Internet service.
This forwarding of network traffic can become dangerous at times, especially with the availability of modern cracking tools that can spoof internal IP addresses and make the remote attacker's machine act as a node on your LAN.
To prevent this, iptables provides routing and forwarding policies that can be implemented to prevent abnormal usage of network resources.
The FORWARD chain allows an administrator to control where packets can be routed within a LAN. For example, to allow forwarding for the entire LAN (assuming the firewall/gateway is assigned an internal IP address on eth1), use the following rules:

~]# iptables -A FORWARD -i eth1 -j ACCEPT
~]# iptables -A FORWARD -o eth1 -j ACCEPT

This rule gives systems behind the firewall/gateway access to the internal network. T he gateway routespackets from one LAN node to its intended destination node, passing all packets through its eth1 device.

14.	**Practical 14:** **Using gcc Compiler (Programming in C++) & Using JAVA Compiler (Execution of Simple Java Programs. & Demonstration of Implementing Socket Prog.)**	

Executing shell script with c by using gcc

Step 1 : Open Vi Editor type C program

```
#include<stdio.h>
#include<conio.h>
void main()
{
   printf("Hello! Welcome to gcc Compiler");
   getch();
 }
```

Step 2 : Save the file hello.c

Step 3 : gcc compiler is used to Compile the Program ie. Gnu's Collection Complier.

Step 4 : gcc hello.c

Step 5 : By default output of the program is saved in a.out file

So to Run Default we can write

./a.out

Output :

Hello! Welcome to gcc Compiler

OR

To Compile & Run

gcc hello.c –o hello.out

./hello.out

Output :

Hello! Welcome to gcc Compiler

TCPEchoServer

```
import java.net.*;  // for Socket, ServerSocket, and InetAddress
import java.io.*;   // for IOException and Input/OutputStream

public class TCPEchoServer
 {
  private static final int BUFSIZE = 32;       // Size of receive buffer
  public static void main(String[] args) throws IOException
   {
    if (args.length != 1)                          // Test for correct # of args
     throw new IllegalArgumentException("Parameter(s): <Port>");

    int servPort = Integer.parseInt(args[0]);

    // Create a server socket to accept client connection requests
    ServerSocket servSock = new ServerSocket(servPort);

    int recvMsgSize;   // Size of received message
    byte[] byteBuffer = new byte[BUFSIZE];  // Receive buffer

    for (;;) { // Run forever, accepting and servicing connections
     Socket clntSock = servSock.accept();     // Get client connection

     System.out.println("Handling client at " +
      clntSock.getInetAddress().getHostAddress() + " on port " +
          clntSock.getPort());
     InputStream in = clntSock.getInputStream();
     OutputStream out = clntSock.getOutputStream();

     // Receive until client closes connection, indicated by -1 return
     while ((recvMsgSize = in.read(byteBuffer)) != -1)
      out.write(byteBuffer, 0, recvMsgSize);

     clntSock.close();  // Close the socket.  We are done with this
client!
    }
```

```
   /* NOT REACHED */
  }
}
```

TCPEchoClient

```
import java.net.*;  // for Socket
import java.io.*;   // for IOException and Input/OutputStream

public class TCPEchoClient {

  public static void main(String[] args) throws IOException {

   if ((args.length < 2) || (args.length > 3))  // Test for correct # of
args
     throw new IllegalArgumentException("Parameter(s): <Server>
<Word> [<Port>]");

   String server = args[0];       // Server name or IP address
   // Convert input String to bytes using the default character
encoding
   byte[] byteBuffer = args[1].getBytes();

   int servPort = (args.length == 3) ? Integer.parseInt(args[2]) : 7;

   // Create socket that is connected to server on specified port
   Socket socket = new Socket(server, servPort);
   System.out.println("Connected to server...sending echo string");

   InputStream in = socket.getInputStream();
   OutputStream out = socket.getOutputStream();

   out.write(byteBuffer);  // Send the encoded string to the server

   // Receive the same string back from the server
   int totalBytesRcvd = 0;  // Total bytes received so far
   int bytesRcvd;          // Bytes received in last read
   while (totalBytesRcvd < byteBuffer.length) {
    if ((bytesRcvd = in.read(byteBuffer, totalBytesRcvd,
```

```
                byteBuffer.length - totalBytesRcvd)) == -1)
        throw new SocketException("Connection close prematurely");
      totalBytesRcvd += bytesRcvd;
    }

    System.out.println("Received: " + new String(byteBuffer));
    socket.close();  // Close the socket and its streams
  }
}
```

Compile & Run the JAVA Program with javac & java as usual.

15.	**Practical 15: (Demo Practical)** **Setting up Hardware Devices i.e. Sound card & printer**	

Configure Linux PRINTER Server Step By Step Guide Example and Implementation

Linux uses the **Common UNIX Printing System**, also known as CUPS. **CUPS** uses the Internet Printing Protocol (IPP) to allow local printing and print sharing. The ***/etc/cups/*** directory stores all the configuration files for printing.

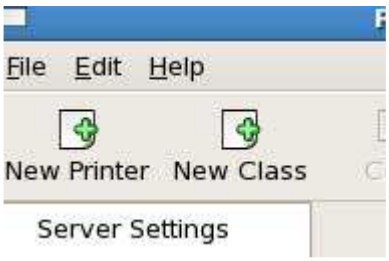
However, these files can be easily managed with the Printer Configuration Tool in Linux.
Exam question Raw (Model) printer named printer1 is installed and shared on 192.168.0.254. You should install the shared printer on your PC to connect shared printer using IPP Protocols.

Exam question Raw printer named printerx where x is your station number is installed and shared on
server1.example.com. Install the shared printer on your PC to connect shared printer using IPP Protocols. Your server is 192.168.0.254.

Before you can use any printer, you first have to install it on a Linux system on your network. To start the Printer Configuration Tool, go to the**System menu** on the top panel and select **Administration**, Printing or execute the command **system-config-printer**.

If no printers are available for the system, only the **Server Settings** view is available for selection. If local printers are configured, a **Local Printers** menu will available.

Install new printer
Click **New Printer** on the toolbar.



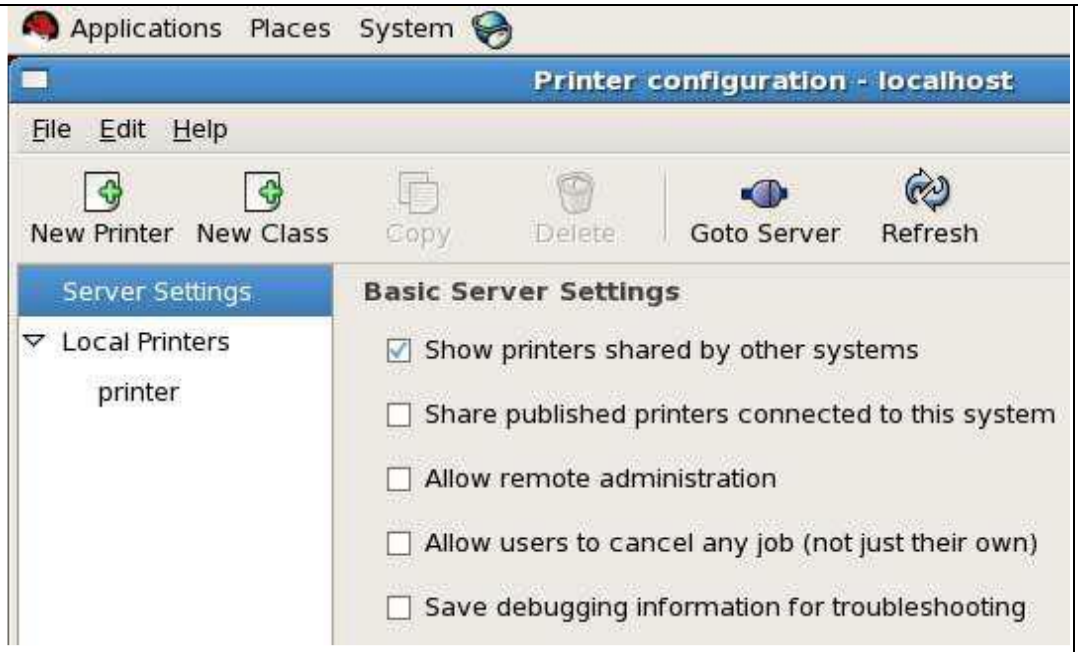
In the dialog window that appears, accept the **default queue name** or change it to a short, descriptive name that begins with a letter and does not contain spaces. Then select **printer** from list and click on **forward** and click on **finish**.

spool directories
When your system prints a file, it makes use of special directories called **spool directories.** The location of the spool directory is obtained from the printer's entry in its configuration file. On Linux, the spool directory is located at **/var/spool/cups** under a directory with the name of the printer.

print job
A print job is a file to be printed. When you send a file to a printer, a copy of it is made and placed in a spool directory set up for that printer.

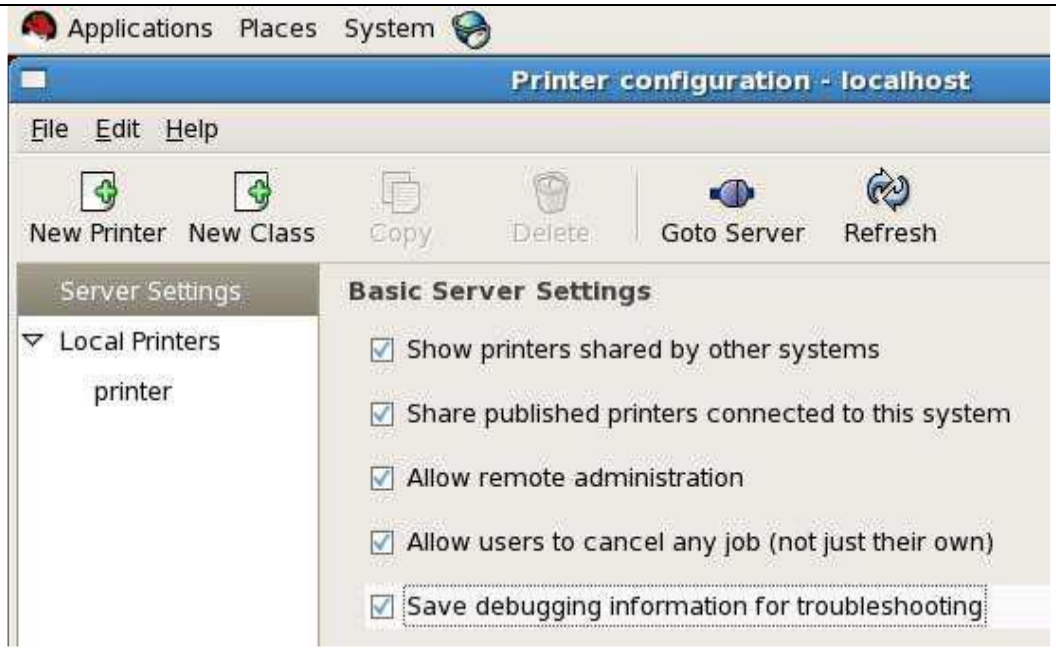
classes
CUPS features a way to let you select a group of printers to print a job instead of selecting just one. That way, if one printer is busy or down, another printer can be automatically selected to perform the job. Such groupings of printers are called **classes**. Once you have installed your printers, you can group them into different classes.

Once you have successfully installed local printer it will show in right pane. and in left pane you can see
all **administrative options.**

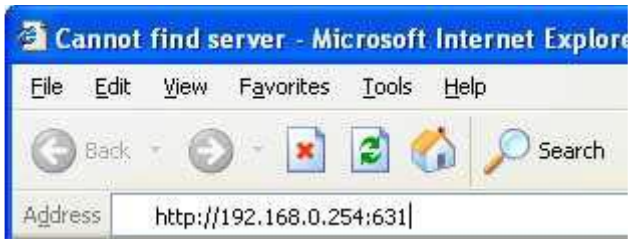


- To **view shared printer** on other system Tick mark on first option
- To **share** locally attached printer tick mark on second option
- To **allow remote administration** of this printer check mark on third option

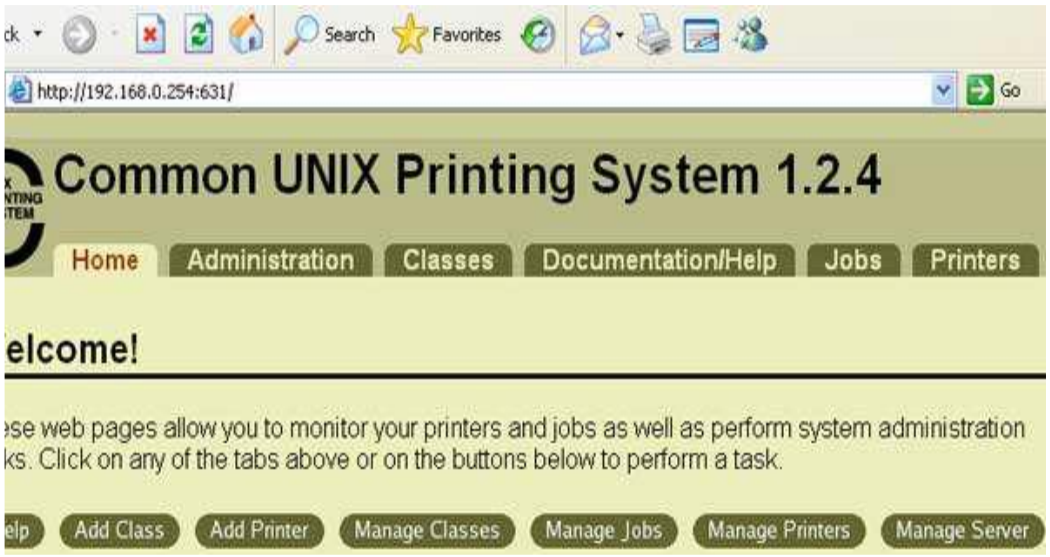
Tick mark on appropriate option and click on **apply**



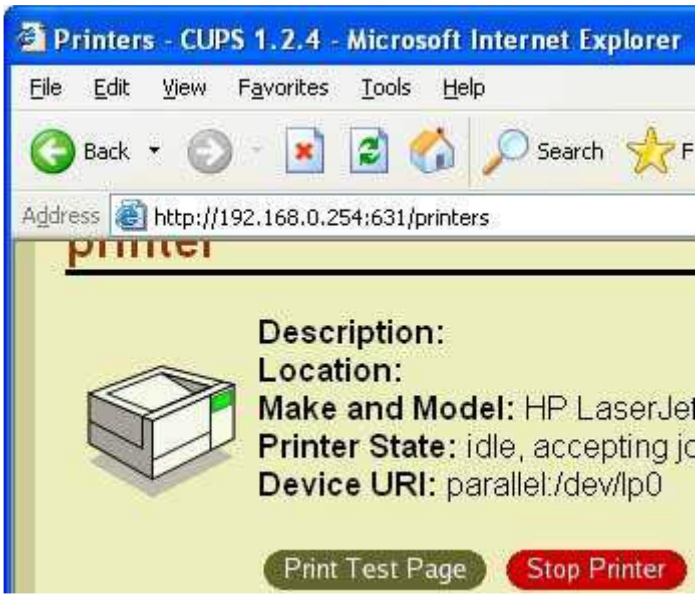
Configure window clients
Go on window system and **ping** from printer server and open **internet explorer** and give the **ip address** of server with printer port **631**



This will launch **CUPS web application** click on **manage printer**



Now you will see the **shared printer** on server click on print **test page**



A test page will be send on printer server copy this **url of printer**

Click on **start** button select **printer and fax** and click on **add new printer**. this will launch add new printer wizard
click **next** on welcome screen and select **network printer**

On this screen select **internet printer** and **paste the url** which you copied

from **internet explorer**

Install appropriate driver from list or use have disk option you have drive cd and click next. On next screen set this printer **defaults** and click on next and **finish**.

Remote administration of print server

Go on linux system and **ping** from server and click on **printing** from **administration menu**

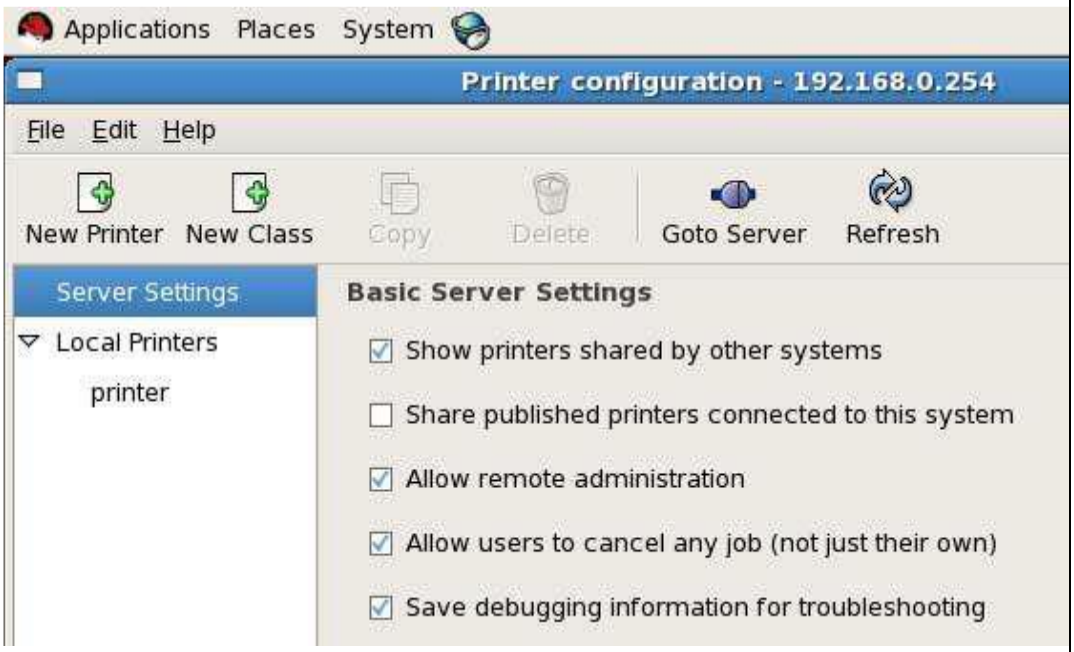
Now click on **go to server**

Now give print server **ip address**

It will take few minute to connect from server depending on network speed

Now give **root password** to connect printer server

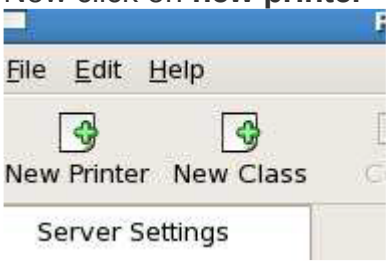
You can see all **print administrative Manu** in right pane Once you have connected with sever



Configure Linux clients

Go on linux system and **ping** from server and click on printing from **administration menu**

Now click on **new printer**



Click on forward In the next New Printer screen, select the type of connection to **internet printing protocols** and in hostname give **server ip**and printer name in **printername.**

select the appropriate model. If multiple drivers are available, select the one most appropriate for your configuration. If you do not want to choose the **default** and click forward and finish. The main Printer Configuration window should now include the name of your printer.

To print test page click on **print test page** and a test page will send to print server

Managing Printers from the Command-Line

The **lpadmin** command enables you to perform most printer administration tasks from the command-line.

```
[root@Client1 ~]# lpadmin
Usage:

    lpadmin [-h server] -d destination
    lpadmin [-h server] -x destination
    lpadmin [-h server] -p printer [-c add-class] [-i interface] [-m model
                        [-r remove-class] [-v device] [-D description]
                        [-P ppd-file] [-o name=value]
                        [-u allow:user,user] [-u deny:user,user]
```

***lpc* To view all known queues**
***lpr* To send print requests to any local print queue**
***lpq* To see the print queue**
***lprm* To delete the jobs of your choice use it with the job number**
***lp* To print any file.**

```
[root@Client1 ~]# lpadmin -p 192.168.0.254
[root@Client1 ~]# lp test
request id is printer-2 (1 file(s))
[root@Client1 ~]#
```

16.	**Practical 16: (Demo Practical)** **Working with X-Windows** **A] Switching TO A Graphical Login** **B] Setup video card, monitor and mouse for the X-server** **C] Change my default desktop to KDE** **D] Accessing X-window remotely.** **E] Installing TrueType fonts from my MS Windows partition?** **F] How do I Display and Control a Remote Desktop using VNC**	
	Working with X-Windows :- **A] Switching to a Graphical Login :-** **There are two methods** **1) By using initab file** **2) By using CTRL + ALT + F1 …… F6** **1) By using initab file** **Important — Access to Software Repositories Might Be Required** To switch to a graphical environment, you might need to install extra software from a *repository*. You can access Red Hat Enterprise Linux repositories with your Red Hat Network subscription through the Internet or use a Red Hat Enterprise Linux installation DVD as a repository. If you installed using a text login and wish to switch to a graphical login, follow this procedure. 1. If you are not already root, switch users to the `root` account: **`su -`** Provide the administrator password when prompted. 2. If you have not already done so, install the **X Window System** and a graphical desktop environment. For example, to install the GNOME desktop environment, use this command: **`yum groupinstall "X Window System" Desktop`** To install the KDE desktop environment, use: **`yum groupinstall "X Window System" "KDE Desktop"`**	

This step may take some time as your Red Hat Enterprise Linux system downloads and installs additional software. You may be asked to provide the installation media depending on your original installation source.

3. Run the following command to edit the `/etc/inittab` file:
   `vi /etc/inittab`
4. Press the **I** key to enter `insert` mode.
5. Find the line that includes the text **`initdefault.`** Change the numeral `3` to `5`.
   Type init 5
6. Type `:wq` and press the **Enter** key to save the file and exit the **vi** text editor.

Reboot the system using the `reboot` command. Your system will restart and present a graphical login.

2) <u>By using CTRL + ALT + F1 …… F6</u>

How to switch between text and graphical consoles?

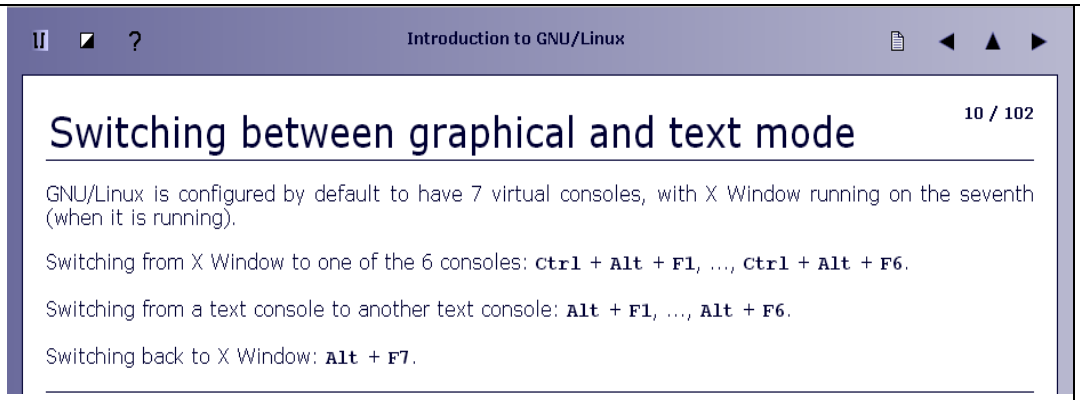
Pressing the key combination <Ctrl><Alt><F1> will switch you to to the first text console at any time.

<Ctrl><Alt><F2> will switch you to the second text console, **<Ctrl><Alt><F3> to the third text console, etc, up to <Ctrl><Alt><F6>, for the total of 6 text consoles.**

<Ctrl><Alt><F7> will switch you to the first graphical user interface (GUI) console if one is running. **<Ctrl><Alt><F8>** to the second GUI console, etc., up to **<Ctrl><Alt><F11>** for the total of 5 GUI consoles.

The 12th console is either used as the 6th GUI (RedHat 6.1) or a place to which kernel messages are continually displayed (Mandrake 7.0, really cool feature). Typically none or only the first GUI console is running.

<Ctrl><Alt><F1> means: "Press the left <Ctrl> and <Alt> keys and hold them. Now press <F1>. Release <F1>. Release <Ctrl> and <Alt> keys."



B) setup video card, monitor and mouse for the X-server?

This should be set-up during your Linux initial installation unless you skipped the step. To set it up now, you may try, as root, one of these text-mode configurators (as root):

Xconfigurator
XF86Setup
xconf

Under RedHat, you can also run the command setup (as root) and access Xconfigurator from there.

To setup X-windows under Linux, you may need to know your hardware. You may want to dust your monitor manual to see what maximum synchronization frequencies (vertical and horizontal) your monitor supports. The message when the computer boots may give you a clue about what type of video card you have and with how much video memory. Also running these commands will likely provide helpful information:

lspci
SuperProbe

Read the label underneath your mouse to find out about the mouse type. Next time you buy a mouse, get a 3-button "Linux-ready" Logitech or similar (Linux makes good use of all three mouse buttons). A standard (clone or not) mouse always makes a good sense--I would never buy an unusual mouse because it may requires a weird driver or otherwise be a installation/functional pain.

During testing of the X-server, if the screen goes blank, displays

funny lines, or otherwise obviously does not function as designed, kill it fast with <Ctrl><Alt><BkSpace> and re-check your monitor sync frequencies. Running too high frequencies can be harmful to your monitor.

If you really have problems, set up a plain vga X server first (resolution 640x480 pixels, 16 or 256 colours). You can fine-tune it later, after you get some understanding of how things work on your system, or perhaps with the help of some nicer setup tools available under X.

After setting up X, you can start it manually using:

startx &

The "&" makes your command run in the background so that your text terminal is not blocked. You could also use:

init 5

which will switch your system to runlevel 5, which means "the graphical user interface run level". To start X automatically (or not, your choice) on the system reboot, read the next few paragraphs.

Can I have a GUI login prompt?

To start your X-server automatically on the system start-up and display a graphical login prompt, you have to change (as root) just one character in the file /etc/inittab. This file specifies something like:

id:3:initdefault:

Change it to

id:5:initdefault:

This sets up the default runlevel to 5, which is X-Windows. The meaning of the different runlevels is explained in the same /etc/inittab file :

0 - halt (Do NOT set initdefault to this)
1 - Single user mode
2 - Multiuser, without NFS (The same as 3, if you do not have networking)
3 - Full multiuser mode
4 - unused
5 - X11
6 - reboot (Do NOT set initdefault to this)

You can change the runlevel from the command line. E.g., this command (has to be executed as root):

init 6

will reboot your computer, while the following command would switch your computer to a single-user mode:

init 1

To find out which runlevel I am currently at, I use the command runlevel.

To fine-tune the appearance of my X login screen, I can use (under X):

kcontrol &

and select "System"-"Login Manager". I like a login screen with an analog clock, big font, the login name of the last user already typed in, and the focus pre-set on the password field in the dialog box.

C] change my default desktop to KDE (or Gnome or yet another)

In my home directory, I create (or edit if it exists) the file .xsession using my favourite pico editor:

pico .xsession

[Pls note the dot at the beginning of the filename, files with names starting with dots are normally "invisible".]

On my RedHat 6.2 system, the file contains just one line:

exec startkde

KDE clearly works best for me, although it feels heavy on older hardware or under a load. Here is my list of windows managers available on the RedHat or Mandrake installation CD:

startkde (to run kde. on some systems, the command may be kde)
gnome-session (to run Gnome)
xfce (to run XFce, my favourite "lightweight" desktop)
afterstep (to run afterstep)
AnotherLevel (to run AnotherLevel)
fvwm2 (to run fvwm2)
fvwm (to run fvwm)

Of course, the alternative windows manager will run only if it is installed on your system. The above windows managers are available on RH/Mandrake CDs for you to decide if you want to install them. I use almost exclusively KDE, although the other managers may be smaller and faster. Gnome is a famous X-windows project which is said to be more advanced and is prettier than KDE, but it is still quite buggy, so perhaps not recommended unless you don't mind occasional trouble. RH6.x contains both major X-windows systems, Gnome and KDE.

D] Acessing X-window remotely.

Start X-server on the local machine, e.g.

xinit

From the x-terminal give the remote machine the permission to display on your local screen:

xhost name_of_the_remote_server

In the really secure environment of my house, I could even give all servers the permission to display on my screen using (don't do it when connnected to the Internet):

xhost +

Telnet the remote server.

Start an X-program on the remote server directing the display on your local screen, for example, you may start a window manager:

startkde -display local_machine_name:0.0 &

The symbol "&" puts the command in the background, so that your telnet window is still available to you.

The 0.0 means "display zero, screen 0", which is your first screen on the first display and makes sense since you can have many concurrent sessions of X running on your computer with Linux.

You don't have to specify the "-display" option if your environment variable DISPLAY specifies the correct location on your current terminal, which is the case on my systems by default, but not on everybody else's as I am told. You can check your DISPLAY setting using:

echo $DISPLAY

After I finish my remote X session, I restore the access control to my X-server using:

xhost -name_of_the_remote_server

or

xhost -

Example. This sequence of commands will run Netscape on the remote machine called marie, directing the display to the X-server with X-windows manager which runs on the local machine hacker:

startx
xhost marie
telnet marie
[login]
netscape -display hacker:0.0 &
[do my stuff]

[logout]
xhost –marie

In principle, you can run a program on any computer on the network, and display the output on any other (not necessarily the one you are sitting at).
I use remote X-windowing a lot to run fat programs (kde, Word Perfect 8, and Netscape) on a slim machine (486-33, 8 MB mem) which would not be able to run those by itself. It is also a convenient and fast way to work with files on a remote system for which the nfs mount is not set up.
X-windows was designed to run remotely over the network. Remote X-windowing is a very powerful tool, on top of being quite a pleasant experience. Try it out.
You can even run a program on a remote Linux (or any Unix) computer and redirect the display to a local MS Windows machine if you install an X-windowing program for MS Windows. For a good overview of choices,

E] Installing TrueType fonts from my MS Windows partition?

Some distributions come with a TrueType font server but no (or a limited choice of) TrueType fonts. You can install your own TrueType fonts though. Here is how I did it manually. Mandrake includes a GUI utility to transfer you MS Windows fonts to Linux, so you don't have to bother with the procedure below.
From under K-menu (KDE), select "System"-"Font Manager" (or equivalent) and note what fonts you have installed.
On the command line, check if the "free type" font server is installed:

rpm -q freetype

This queries (option "q") the rpm package manager for the package "freetype". If the package is installed, go to next step. If "freetype" is not installed, install it now from your distribution CD. "freetype" was installed on my system after a "full" RH installation.
As root, make a directory that is to hold your TrueType fonts:

cd /usr/X11R6/lib/X11/fonts
mkdir TrueType

This directory is referred to in the configuration file /etc/X11/XF86Config so make sure that the name of the directory is

exactly as shown. If you would like to name the directory differently, you have to edit /etc/X11/XF86Config and make appropriate adjustments. My "default installation" RedHat contained such a line:

FontPath "/usr/X11R6/lib/X11/fonts/TrueType"

As root, copy your *.ttf files from the original location to the TrueType font directory that you just created. I took some TrueType from my MS Windows partition, you may need to use a different source location:

```
cd /usr/X11R6/lib/X11/fonts/TrueType
cp /mnt/dos_hda1/windows/fonts/my_private_fonts/*.ttf .
```

Before copying any fonts, make sure that it does not violate your licence agreement.

As root, run the following commands:

```
cd /usr/X11R6/lib/X11/fonts/TrueType
ttmkfdir > fonts.dir
cp fonts.dir fonts.scale
```

Close all X-windows applications and log out from X-windows.

As root, restart your X-font server (or reboot your computer):

```
/etc/rc.d/init.d/xfs stop
/etc/rc.d/init.d/xfs start
```

Log back onto your KDE, and from under K-menu, select "System"-"Font Manager" to see if the fonts installed correctly.

F] How do I Display and Control a Remote Desktop using VNC

(VNC = Virtual Network Computing). A very useful application--don't miss it.

VNC is a cross-platform utility that allows me to display a remote graphical desktop over a standard network connection. For example, I can use VNC on an MS Windows PC to display an X-window environment of my mighty Linux server downstairs, or the other way around. VNC will even run over a 56k modem networking, but probably only for fun or in emergency (too slow a connection for normal work).

Recent Mandrake or RH will have vnc on their distributions CD. The MS Windows version you have to download yourself. See http://www.uk.research.att.com/vnc/ for download information and

more details.

On Linux, VNC consists of four commands: vncserver, vncviewer, vncpasswd, and vncconnect. I typically need just two of them: vncserver and vncviewer. A brief description of the commands follows.

vncserver

The server that has to be running on the host (remote) computer. You start the server as the user whose desktop will be displayed (don't run the server as root or somebody else somebody may kidnap your computer!).

vncviewer

The local application which connects to the vncserver and displays the remote environment. You need to know the password and ip address of the server to connect.

vncpasswd

Password selection utility for vncserver. The server won't run without password (good behaviour). Therefore, if you don't select one, it will prompt you. Hence, I don't need to explicitly run vncpasswd.

vncconnect

Tells vncserver to connect to a listening VNC viewer on the given computer and port. This way I can avoid giving anybody a password.

Xvnc

A "master" program that I don't really need to run directly (vncserver and vncviewer are scripts which call Xvnc).

For a list of all available options I run:

Xvnc -help

It is not recommenced to run the VNC server as root due to potential security issues. If you need root privileges, login as a user and then execute su

Two examples of "typical" sessions follow.

Example 1. Sitting at an MS Window computer, I can display an X environment from my Linux server, using the following sequence:

[start a DOS terminal and type in the following command]

telnet my_linux_server_name

[log in to your user account on Linux and type in it the following command]

vncserver

[provide a really good password of your choice when prompted; mine was "357+Simon&Garfunkel"]

[re-enter the same password for confirmation]

[watch the messages and note the screen number on which the server is started; mine was ":4"]

[From the "Start" menu on the MS Windows computer, select

	"Programs" - "Vnc" - "Run VncViewer" [in the input box that appears, type the server ip address and screen number as shown on the next line] **my_linux_server_ip_address:4** [in the input box that appears type the password as follows] **357+Simon&Garfunkel** [an X-windows desktop should now appear on top of your MS Windows desktop] [do your work as you normally would in Xwindows] [when done, switch to the telnet session window and type in it the following two commands] **vncserver - kill :4** logout Example 2. Sitting at my Linux X desktop, I can display and remotely control an MS Windows computer screen. Hopefully, nobody else is using this MS Windows computer at the same time, because I move its mouse pointer. [Walk to the MS Windows computer because you probably cannot telnet it] [From the "Start" menu, select "Programs" - "Vnc" - "Run WinVnc (app mode)" [From the "System Tray", click the mouse right button on the "Vnc" icon, and select "Properties"] [In the dialog box that appears, fill in the password. Leave the screen number on "auto".] [Walk back to your Linux desktop] [Start an X terminal and type in it] **vncviewer ms_windows_server_name_or_ip** [When prompted, type in the password] [a MS Windows desktop should now appear on top of your X] [do your work as you normally would on MS Windows] [When done, right click on the Vnc icon in the system tray and select "Close VNC".]	

17. Practical 17: Configuring Mail Services Using Sendmail

Mail Services

- Sendmail is the most widely used Mail Transport Agent (MTA) on the internet
- MTAs send mail from one machine to another.
- Sendmail is not a client program, which you use to read your email.
- Sendmail is one of the behind-the-scenes programs which move email over the Internet.
  - Normally it runs as a background daemon
  - Can even be run out of the super daemon (xinetd)

Sendmail

How to configure linux sendmail server step by step guide Example and Implementation

In OUR DDAY TO DAY LIFE, email is an essential component to the work day.
Email is used to communicate with many peoples..
By default **sendmail** server allows to connect to **localhost** only. So we should edit the **/etc/mail/sendmail.mc** file to allow connect to other hosts.
The **sendmail** daemon is configured from a directory of files in /etc/mail and a directory of configuration files in**/usr/share/sendmail-cf**. There are two basic configuration files:

* **sendmail.cf** The main **sendmail configuration file**.
* **sendmail.mc** A macro that's easier to edit, which can be used to generate a new **sendmail.cf** file.

For this example we are using two systems one linux server one linux clients.
These are the pre quest for a sendmailserver

* A linux server with *ip address* 192.168.1.3 and *hostname* server.rkt.com
* A linux client with ip address 192.168.1.12 and hostname server.rkt1.com
* A Configured DNS server on Linux server
* *Updated /etc/hosts file* on both linux system
* Running portmap and xinetd services (service xinetd stop , service portmap stop)
* Firewall should be off on server (service iptables stop)

We have configured all these steps in our pervious article.

Check DNS server

We suggest you to review that article and we had already configured DNS server before start configuration of **sendmail server** we have to check whether our DNS is properly configured or not .
use dig command (`dig server.rkt.com & dig -x 192.168.1.111`).
Once you have completed the necessary steps follow this guide.

==========

Configure sendmail server

Sendmail and m4 rpm are required to configure sendmail .

```
]#rpm –qa | grep sendmail
```

Mail server program reads the **/etc/mail/sendmail.cf**. To change the configuration on mail server, we should edit the**/etc/mail/sendmail.mc** file. When Sendmail is started or restarted with the **service sendmail restart** command a new **sendmail.cf** file is automatically generated.

open /etc/mail/sendmail.mc for editing

```
vi /etc/mail/sendmail.mc
```

show hidden line with : set nu option on vi editor command mode.
Go to [line no 116]

```
sendmail.mc
DAEMON_OPTIONS('Port = smtp , Addr = 192.168.1.3,Name = MTA ')
```

*You can allow other computers to use your **sendmail server** by commenting*
In the **sendmail.mc** file , lines that begin with **dnl**, which stands for delete to new line, are cons
Some lines end with dnl, but lines ending in dnl are not comments.

comment this line with dnl keyword followed by **# sign**

```
sendmail.mc
dnl # DAEMON_OPTIONS('Port = smtp , Addr = 192.168.1.3,Name = MTA ')
```

save this file with :wq and exit.
Now generate new sendmail.cf file by using *m4 command* as shown here

```
sudo bash –c "cd /etc/mail && m4 sendmail.mc>sendmail.cf"
```

]# vi /var/named/forward.zone
Add MX Entry in forward.zone

```
rk1.com  IN MX 10 server.rkt1.com.
```

Now restart sendmail service and also set it on with chkconfig

```
service sendmail restart
```

if sendmail service restart without any error means you have configured sendmail successfully.

==========

Configure sendmail client side

We are using another linux system to test **sendmail server**. All configuration are same as you have done on server system.

1. Check **sendmail** and **m4** rpm for install.
2. Open **/etc/mail/sendmail.mc** file and locate **line no 116** and put a **dnl**with **#** sing

and save file. All step are same which you have done on server.

3**. Now generate new sendmail.cf file by using m4 command as shown here**

```
sudo bash –c "cd /etc/mail && m4 sendmail.mc>sendmail.cf"
```

4. **Now restart sendmail service and also set it on with chkconfig**

```
service sendmail restart
```

==========

Testing of sendmail server

We will test **sendmail server** by sending and receiving mail in lab environment.
for this we use two user one on each system.

Now create one user on each system **kiranmail** on **server**

```
useradd kiranmail
```

and **clientmail** on **client system**

```
useradd clientmail
```

Now send mail from user kiranmail to clientmail and from clientmail to user kiranmail and also check eac
mail command
Use full user name to send mail. For example to send mail to clientmail use clientmail@server.rkt1.com a
kiranmail use kiranmail@server.rkt.com

Command is

]# mail clientmail@server.rkt1.com

Type following lines :
Subject :
Cc :
Data :
.
EOT
OR other way is
]# telnet localhost 25
Type following lines :
mail from :
rcpt to :
data
.
quit
]#