

# Linux Essentials

Lesson 00:

IGATE is now a part of Capgemini

People matter, results count.

Capgemini Public



# Document History

Date	Course Version No.	Software Version No.	Developer / SME	Reviewer(s)	Approver	Change Record Remarks
11-August-2016	1.0	5.5	Amal Thambi			Created as per the Infra BU requirement

# Course Goals

## Course Goals

- This course is designed for entry level Infra resources to enable them with the skills required for provisioning and maintaining a Linux Server in production environment.



## Pre-requisites

This course requires that you meet the following prerequisites:

- The audience should have completed IS Linux Fundamentals

# Intended Audience

- For Entry level Infrastructure Services(IS) resources



IGATE Sensitive  
Proprietary and Confidential

# Day Wise Schedule

## ➤ Day 1

**Lesson 1: Basic of RHEL & Installation**

**Lesson 2: System Initialization**

**Lesson 3: Device Management**

**Lesson 4: Process Management**

## ➤ Day 2

**Lesson 5: System Services**

**Lesson 6: Bash Editing**

**Lesson 7: File System / Disk Management**

## ➤ Day 3

**Lesson 8: File Management**

**Lesson 9: Package Management**

**Lesson 10: User & Group Management**

**Lesson 11: File Permission**

## ➤ Day 4

**Lesson 12: Network Services**

**Lesson 13: Backup & Restore**

**Lesson 14: Troubleshooting**

# Table of Contents

## **Lesson 1: Basic of RHEL & Installation**

- 1.1 History
- 1.2 Versions
- 1.3 Pre-Req for Installation of RHEL 5.5
- 1.4 Installation of RHEL 5.5 on VM Machine

## **Lesson 2: System Initialization**

- 2.1 BIOS Initialization
- 2.2 Boot Loader
- 2.3 Kernel initialization

## **Lesson 3: Device Management**

- 3.1 Device Nodes
- 3.2 Char Device
- 3.3 Block Device

## **Lesson 4: Process Management**

- 4.1 Process and Job Control

# Table of Contents

## **Lesson 5: System Services**

- 5.1. NTP - Network Time Protocol
- 5.2. System Logging
- 5.3. SSH Service
- 5.4. SCP Service
- 5.5. CUPS Service
- 5.6. xinetd Service – Telnet
- 5.7. Service Management
- 5.8. Scheduling Service - crontab

## **Lesson 6:Bash Editing**

- 6.1. Bash profile & its components
- 6.2. Set Environment Variables
- 6.3. Create aliases

## **Lesson 7: File System / Disk Management**

- 7.1. File system Management
- 7.2. Disk Management

# Table of Contents

## **Lesson 8: File Management**

- 8.1. File Operation
- 8.2. Disk Utilities
- 8.3. Links
- 8.4. Compression / Decompression of files

## **Lesson 9: Package Management**

- 9.1. Package management
- 9.2. RPM
- 9.3. YUM - Yellow Update Modifier

## **Lesson 10: User & Group Management**

- 10.1 User & Group Management
- 10.2 Sudo

## **Lesson 11: File Permission**

- 11.1 File permission
- 11.2 ACL - Access Control List for Files & Directory

# Table of Contents

## **Lesson 12: Network Services**

- 12.1 Network configuration
- 12.2 NW Devices
- 12.3 Virtual NW Adapters
- 12.4 NW Diagnostics - ping , traceroute

## **Lesson 13: Troubleshooting**

- 13.1 Root Password Recovery

# References

- Books:

**RHCSA/RHCE Red Hat Linux Certification Study Guide (Exams EX300), 6th Edition - Certification Press**



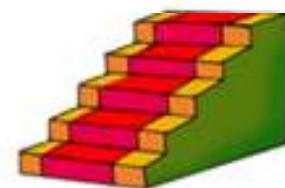
**Red Hat Enterprise Linux 6 Administration: Real World Skills for Red Hat Administrators – Sybex Publications**

- Websites

[https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/5/html/Deployment\\_Guide/index.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/index.html)

# Next Step Courses

- RHEL 7 Administration



## Other Parallel Technology Areas

- Red Hat Linux Server

# Linux Essentials

## Lesson 1 Basic of RHEL & Installation

# Module Overview

**1.1 History**

**1.2 Versions**

**1.3 Pre-Req for Installation of RHEL 5.5**

**1.4 Installation of RHEL 5.5 on VM Machine**

## 1.1. History

- **1984: The GNU Project and the Free Software Foundation.**
  - Creates open source version of UNIX utilities.
  - Creates the General Public License (GPL).
- **Software license enforcing open source principles**
  - 1991: Linus Torvalds Creates open source, UNIX-like kernel, released under the GPL  
Ports some GNU utilities, solicits assistance online.
  - Today: Linux kernel + GNU utilities = complete, open source, UNIX-like operating system.
- **Packaged for targeted audiences as distributions.**

## 1.2 Versions

### ➤ Versions

- RHEL 2.1
- RHEL 3
- RHEL 4
- RHEL 5
- RHEL 6
- RHEL 7

### ➤ Editions

- Server
- Workstation

## 1.3 Pre-Req for Installation of RHEL 5.5

- CPU
  - 2GHz or higher
- Memory/RAM
  - 1 GB minimum, upto the system limit
- Hard Disk
  - 4 GB minimum

## 1.4 Installation of RHEL 5.5 on VM Machine

1. Boot from DVD
2. Select Install or upgrade existing system options.
3. Select language as English
4. Select Keyboard type as US English
5. Optionally test media if required
6. Select the storage device
7. Enter the hostname
8. Select the timezone
9. Enter the root password
10. Select the partition option
11. Select the type of installation(default:basic server)
12. Select additional software to be installed
13. Start the installation.
14. Once the installation is completed, perform the post installation tasks

# Summary

- In this Module you have learnt:
  - What is the history of Linux?
  - What are the different version of Linux?
  - What are the Pre-Req for Installing of RHEL 5.5
  - Installation of RHEL 5.5 on VM Machine

# Lab Exercise



## ➤ INSTALLATION OF Linux

# Review Questions

1. Who created Linux?
2. What is the latest version of RHEL?

# Linux Essentials

## Lesson 2 System Initialization

# Module Overview

- 2.1 BIOS Initialization**
- 2.2 Boot Loader**
- 2.3 Kernel initialization**

## 2.1 BIOS Initialization

- BIOS will primarily check all the Peripherals devices.
- It selects boot device which can either be floppy disk, cd-rom, hard drive etc.
- BIOS then reads first sector of boot device & executes it. This includes information about GRUB.
- GRUB loads and executes kernel.
- Kernel executes init program which is the first process to get executed on the machine. Hence it's Pid is 1.

## 2.2 Boot Loader

- **1984: The GNU Project and the Free Software Foundation.**
  - Creates open source version of UNIX utilities.
  - Creates the General Public License (GPL).
- **Software license enforcing open source principles**
  - 1991: Linus Torvalds Creates open source, UNIX-like kernel, released under the GPL  
Ports some GNU utilities, solicits assistance online.
  - Today: Linux kernel + GNU utilities = complete, open source, UNIX-like operating system.
- **Packaged for targeted audiences as distributions.**

# GRUB & grub.conf

- **GRUB- the Grand Unified Bootloader**
  - Command-line interface available at boot prompt.
  - Mostly Boots from ext2/ext3
  - Can be password protected
- **Grub configuration file is /boot/grub/grub.conf.**
- **Changes to grub.conf take effect immediately.**

## 2.3 Kernel initialization

### ➤ Kernel boot time functions

- Device detection
- Device driver initialization
- Mounts root file system read only
- Loads initial process (init)

# init initialization

- **init reads its config: /etc/config**
  - Initial run level
  - System initialization scripts
  - Run level specific scripts directories
  - Initialize X in run level 5

# Run Levels

- init defines run levels 0-6, S.
- The run level is selected by either,
  - The default in /etc/inittab at boot.
  - Passing an argument from the boot loader.
  - Running init x after boot (where x is the desired run level).
- Show current & previous run levels
  - /sbin/runlevel

# /etc/rc.d/rc.sysinit

➤ **Important tasks include:**

- Activate udev & selinux
- Sets kernel parameters in /etc/sysctl.conf
- Sets the system clock
- Loads keymaps
- Enables swap partitions
- Sets hostname
- Root filesystem check & remount
- Activate RAID & LVM devices
- Enable disk quotas
- Check & mount other file systems
- Cleans up stale locks and PID files

## /etc/rc.d/rc

- Initializes the default run level as per the /etc/inittab initdefault line such as id:3:initdefault:
  - /etc/rc.d/rc 0
  - /etc/rc.d/rc 1
  - /etc/rc.d/rc 2
  - /etc/rc.d/rc 3 ← (Run level 3)
  - /etc/rc.d/rc 4

# System Shutdown

## ➤ Shutting down the system

- Shutdown -h now
- Halt
- Poweroff
- init 0

# System Reboot

- **Rebooting rarely fixes problem in linux.**
  - If you feel a reboot is necessary try bringing the system down to run-level 1 and then back up to runlevel 3 or 5. This is much faster than a reboot.
- **Rebooting the system**
  - Shutdown -r now
  - Reboot
  - init 6

# Summary

- In this Module you have learnt:
  - BIOS Initialization
  - Boot Loader
  - Kernel initialization

## Lab Exercise

---

- **Changing the runlevel**
- **Viewing the runlevel directories, default runlevel**
- **changing the default runlevel**

# Review Questions

- 1. What is a runlevel?**
- 2. What is the default runlevel of your linux server?**
- 3. What is initrd?**



# Linux Essentials

## Lesson 3 Device Management

# Module Overview

3.1 Device Nodes

3.2 Char Device

3.3 Block Device

## 3.1 Device Nodes

- In order for the operating system to recognize the hardware device, the device must have a software name, usually referred to as a device special file or device node.
- Device nodes correspond to resources that an operating system's kernel has already allocated.
- computer system accesses device nodes using standard system calls and treats them like regular computer files

## 3.2 Char Device

- Character special files or character devices provide unbuffered, direct access to the hardware device.
- They do not necessarily allow programs to read or write single characters at a time; that is up to the device in question.
- Character devices are sometimes known as raw devices

E,g) Keyboard

### 3.3 Block Device

- **Block special files or block devices provide buffered access to hardware devices, and provide some abstraction from their specifics.**

e.g) hard disk

# Summary

- In this Module you have learnt:
  - Device Nodes
  - Char Device
  - Block Device

# Lab Exercise

- Partition and mount the hard disk to the system temporarily and permanently

# Review Questions

- 1. How is harddisk accessed**
- 2. How a 3rd IDE hdd will be detected in Linux**

# Linux Essentials

## Lesson 4 Process Management

# Module Overview

## 4.1 Process and Job Control

# What is a Process?

- A process is a set of instructions loaded into memory. It is an instance of program in execution.
  - Numeric Process ID (PID) used for identification. It is allocated by the kernel.
  - UID, GID and SELinux context determines filesystem access.
- Normally inherited from the executing user.
- Many processes can run at the same time.

# Process Types

- Foreground processes
  - Foreground processes are the process which run on terminal and user cannot do anything till the process has finished its execution.  
Example : \$find . -name core -print > /tmp/results
- Background processes
  - Background processes are the process which run on background. i.e, process gets detached from the terminal, freeing up the terminal while it is running.  
Example : \$find . -name core -print > /tmp/results &
- Only one process can run in foreground and multiple processes can run on background.

# Process Management Command/Utilities

Command /Utility	Description
Nice/renice	Changes the priority of a new/running process
kill	Terminates or sends a signal to a process
ps	Reports the status of active processes, including process ID, terminal name, execution time, and command name.

## ps command

- ps command is used to display the process attributes of all active processes.
- Syntax:
  - ps [ option [ arguments ] ... ]
- Options:
  - -f - full form
  - -e - system processes

## Altering Scheduling Priority

- Nice values may be altered.
  - When starting a process:  
`$ nice -n 5 command`
  - After starting:  
`$ renice 5 PID`
- Only root may decrease nice value.

# Signals

Most fundamental inter-process communication.

- Sent directly to processes, no user-interface required.
- Programs associate actions with each signal.
- Signals are specified by name or number when sent. Following are few examples,
  - Signal 15, TERM (default) - Terminate cleanly
  - Signal 9, KILL - Terminate immediately
  - Signal 1, HUP - Re-read configuration files
- Ex:
  - By PID: kill [signal] pid

# Terminating Processes

- Linux uses signals to communicate with processes
- Signals are sent with the kill and pkill commands

Number	Signal	Description
9	SIGKILL	Forces the process to terminate unconditionally. This is the "sure kill" signal.
15	SIGTERM	Termination signal. Shuts down the process but gives the process a chance to terminate properly by cleaning up.

Example : kill -9 1814

kill -15 1814

# Interactive Process Management Tools

- top
- Capabilities
  - Display real-time process information.
  - Allow sorting, killing and re-nicing.

# Monitoring and Killing Background Running Processes

- In Linux we can get the information of jobs which are running at the background. This can be done using jobs command.
  - Ex: \$jobs
- A job which is running at the background can be to foreground using fg command.
  - Ex: fg %2

# Summary

➤ In this lesson you have learnt

- What is a Process
- Process command and its options
- Process types
- Kill command
- Foreground and background process

## Lab Exercise

- List processes
- Change process priority
- Kill foreground and background process

## Review Questions

- Question 1: \_\_\_\_\_ is appended to the command to send it at the background.
  
- Question 2: \_\_\_\_\_ command is used to know which all command are running at the background.



# Linux Essentials

## Lesson 5 System Services

# Module Overview

- 5.1. NTP - Network Time Protocol**
- 5.2. System Logging**
- 5.3. SSH Service**
- 5.4. SCP Service**
- 5.5. CUPS Service**
- 5.6. xinetd Service – Telnet**
- 5.7. Service Management**
- 5.8. Scheduling Service - crontab**

## 5.1. NTP - Network Time Protocol

- Network Time Protocol daemon.
- Network Time Protocol is the most common method to synchronize the software clock of a GNU/Linux system with internet time servers.
- It is designed to mitigate the effects of variable network latency and can usually maintain time to within tens of milliseconds over the public Internet.

## 5.2. System Logging

- Linux system administrators often need to look at log files for troubleshooting purposes. In fact, this is the first thing any sysadmin would do.
- The default location for log files in Linux is /var/log

## 5.3. SSH Service

- sshd
- used for secure connectivity to server
- It uses port 22

## 5.4. SCP Service

- scp allows files to be copied to, from, or between different hosts.
- It uses ssh for data transfer and provides the same authentication and same level of security as ssh

## 5.5. CUPS Service

- CUPS (an acronym for Common Unix Printing System)
- Is a modular printing system for Unix-like computer operating systems which allows a computer to act as a print server.
- A computer running CUPS is a host that can accept print jobs from client computers, process them, and send them to the appropriate printer.
- One can access the browser interface by <http://localhost:631>

## 5.6. xinetd Service – Telnet

### ➤ Xinetd

- starts programs that provide Internet services.
- Instead of having such servers started at system initialization time, and be dormant until a connection request arrives, xinetd is the only daemon process started and it listens on all service ports for the services listed in its configuration file. When a request comes in, xinetd starts the appropriate server.
- Because of the way it operates, xinetd (as well as inetd) is also referred to as a super-server.

### ➤ Telnet

- Is a program that allows users to log into your server and get a command prompt
- One of the disadvantages of Telnet is that the data is sent as clear text. This means that it is possible for someone to use a network analyzer to peek into your data packets and see your username and password. A more secure method for remote logins would be via Secure Shell (SSH) which uses varying degrees of encryption.

## 5.7. Service Management

- chkconfig can be used to activate/deactivate/list the services in your server
  - chkconfig httpd on
- service command can be used to start/stop/status services in your system
  - service httpd start

## 5.8. Scheduling Service - crontab

- A system daemon which performs a specific task at regular intervals.
- where the file contains the commands to execute

MIN	HOUR	DOM	MOY	DOW	COMMAND
(0-59)	(0-23)	(1-31)	(1-12)	(0-6)	---
\$ 0	18	*	*	*	/home/gather

# Summary

➤ **In this lesson you have learnt**

- What is NTP?
- How to see System Logging?
- What is SSH Service?
- What is SCP?
- What is CUPS?
- What is xinetd. How it helps other services?
- How to manager Services?
- How to schedule tasks using crontab?



# Linux Essentials

## Lesson 6 Bash Profile

# Module Overview

- 6.1. **Bash profile & its components**
- 6.2. **Set Environment Variables**
- 6.3. **Create aliases**

## 6.1. Bash profile & its components

- The default shell is the Bash shell.
- Bash is a command language interpreter that executes commands read from any input(file/keyboard).
- The Bash Shell starts by reading the /etc/profile file , which usually contains the system variables, user environment and aliases.
- The login process continues with the files
  - .bash\_profile
  - .bash\_login
  - .profile

# User Environment initialization

execute /etc/profile

IF ~/.bash\_profile exists THEN

  execute ~/.bash\_profile

ELSE

  IF ~/.bash\_login exist THEN

    execute ~/.bash\_login

  ELSE

    IF ~/.profile exist THEN

      execute ~/.profile

  END IF

END IF

END IF

## 6.2. Set Environment Variables

- Variables which are available in the users total environment are called as environment variables.
- Few common environment variables are,
  - HOME
    - describes the path to user's home directory.
  - PATH
    - specifies the path, in which, invoked commands needs to be searched.
  - SHELL
    - Sets the default shell that will be used by Tools.

```
export <envvariablename>=value
```

**For persistent environment variables modify the appropriate profile files(system wide or user specific)**

## 6.3. Create aliases

- The alias command allows you to make new shortcuts and synonyms for commonly used commands. The basic usage is:

```
alias newcommand='yourcommand -arguments'
```

```
alias ls='ll'
```

# Summary

➤ In this lesson you have learnt

- What is bash profile?
- How to set environment variable?
- How to create alias?

# Linux Essentials

## Lesson 7 File system and Disk Management

# Module Overview

- 7.1. **File system Management**
- 7.2. **Disk Management**

## 7.1. File system Management

- Linux supports numerous file systems, but common choices for the system disk on a block device include the ext\* family (ext2, ext3 and ext4), XFS, JFS, ReiserFS and btrfs.
- You need to format a partition with appropriate file system before storing data.
- mkfs is used to build a Linux file system on a device, usually a hard disk partition.
  
- df
- du

## 7.2. Disk Management

- Disks can be managed using GUI using Disk Utility tool
- fdisk can be used to create/delete/list partitions
- configuration file /etc/fstab contains the necessary information to automate the process of mounting partitions.

## Summary

➤ **In this lesson you have learnt**

- What are the different file systems used in linux?
- How to create a file system on a partition?
- How to partition a hard disk?



# Linux Essentials

## Lesson 8 File Management

# Module Overview

- 8.1. **File Operation**
- 8.2. **Disk Utilities**
- 8.3. **Links**
- 8.4. **Compression / Decompression of files**

## 8.1. File Operation

### Creating a File

- touch file.txt
- cat > file.txt
- vi file.txt

### Copying a File

- cp example1.txt barney.txt

### Renaming/Moving a File

- mv foo2.txt backups/foo3.txt

### Deleting a File

- rm bar.txt

# Directory Manipulation

## Creating a Directory

- mkdir amal
- mkdir -p linuxtutorialwork/foo/bar
- mkdir -pv linuxtutorialwork/foo/bar

## Copying a Directory

- cp example1 barney

## Renaming/Moving a Directory

- mv barney backups

## Deleting an Empty Directory

- rmdir linuxtutorialwork/foo/bar

## Deleting a Non-Empty Directory

- rm -r backups

## 8.2. Disk Utilities

### ➤ df

- df finds the disk free space or disk usage.
- Ex: \$df
- Outputs a table consisting of six columns. Column names explains each column. Columns, size, used and avail use kilobyte as unit.

### ➤ du

- du command displays the list of directories that exist in the current directory along with their sizes.
- The last line of the output gives the total size of the current directory including its subdirectories.
- Note that by default the sizes given are in kilobytes.

## 8.3. Links

### ➤ HardLinks

- Hard links cannot link directories.
- Cannot cross file system boundaries.

In /full/path/of/original/file /full/path/of/hard/link/file

### ➤ SoftLinks

- To create links between directories.
- Can cross file system boundaries.
- Removing the original file of a softlink will break the link

In -s /full/path/of/original/file /full/path/of/soft/link/file

## 8.4 Compression / Decompression of files

- Red Hat Enterprise Linux provides the bzip2, gzip, and zip tools for compression from a shell prompt.
- The bzip2 compression tool is recommended because it provides the most compression and is found on most UNIX-like operating systems.
- The gzip compression tool can also be found on most UNIX-like operating systems.
- To transfer files between Linux and other operating system such as MS Windows, use zip because it is more compatible with the compression utilities available for Windows..

Compression Tool	File Extension	Decompression Tool
bzip2	.bz2	bunzip2
gzip	.gz	gunzip
zip	.zip	unzip

## Summary

➤ **In this lesson you have learnt**

- How to perform different file operations?
- How to use different disk utilities?
- What is the difference between hard and soft links?
- How to compress/decompress in linux?

# Linux Essentials

## Lesson 9 Package Management

# Module Overview

- 9.1. Package management
- 9.2. RPM
- 9.3. YUM - Yellow Update Modifier

## 9.1. Package management

- RPM started as packaging format for Red Hat Linux.
- It is a program for installing, uninstalling and managing software packages in Linux.
- Advantages:
  - Straight forward program installation/uninstallation.
  - Ease of updating programs.
  - Availability of versions.
  - Software information stored in a local database.
- Packages are provided by Red Hat Network.
  - Centralized management of multiple systems.
  - Easy retrieval of errata packages.
  - Systems must be registered first.
  - Custom package repositories may also be used.

## 9.2. RPM

- Package installation is never interactive.
  - Applies to all software(Core OS & Add-ons).
- 
- Primary RPM options:
    - Install : rpm -i
    - Erase: rpm -e
    - Verbose: -v
    - Query all installed packages : -qa
    - Query a installed package : -qp <rpmname>.rpm

## 9.3. YUM - Yellow Update Modifier

- Yum allows automatic updates, package and dependency management.
- Configuration in /etc/yum.conf and /etc/yum.repos.d/
- Used to install, remove and list software
  - `yum install packagename`
  - `yum remove packagename`
  - `yum update packagename`
  - `yum list available`
  - `yum list installed`
- Graphical Package Management
  - List and install software updates
  - View, install and un-install other packages

# Summary

➤ **In this lesson you have learnt**

- What is rpm format?
- How to install/remove packages using rpm?
- How to install/remove packages using yum?

# Linux Essentials

## Lesson 10 User Management Essentials

# Module Overview

- 10.1 User and Group Management
- 10.2 Sudo

## 10.1 User and Group Management

Command	Description
useradd	Adds a new user account.
userdel	Deletes an user account.
usermod	Modifies an user account.
groupadd	Adds a new group.
groupmod	Modifies an existing group (for example, changes the GID or name).
groupdel	Deletes a group

## User Information

- All user account information except encrypted password is stored in /etc/passwd.
- The fields in /etc/passwd are as follows:
  - username:password:UID:GID:comment:home\_directory:login\_shell.
- The /etc/shadow file contains the users' encrypted passwords and is very important to protect.

# sudo

- **sudo**
  - execute a command as another user.
- **Users listed in /etc/sudoers execute commands with :**
  - an effective user id of 0
  - Group id of root's group
- **An administrator will be contacted if a user not listed in /etc/sudoers attempts to use sudo.**

# Summary

## In this lesson you have learnt

- How to create/modify/delete users?
- How to create/modify/delete groups?
- Where user-password information is stored?
- What is sudo?

# Linux Essentials

## Lesson 11 File Permissions

# Module Overview

11.1 File permission

11.2 ACL - Access Control List for Files & Directory

## 11.1 File permission

- File/Directory Permissions can be modified with chmod by using 2 methods:
  - Symbolic
  - Numeric

## Changing File Permissions - chmod

- chmod command is used to change the file permissions.
- Syntax: chmod <category> <operation> <permission> <filenames>

# Changing Permissions – Symbolic Method

\$ chmod u+x note

- \$ ls -l note
  - -rwx r--r-- 1 ..... note

\$ chmod ugo+x note

- \$ ls -l note
  - -rwxr-xr-x 1 ..... note

- When we use + symbol, the previous permissions will be retained and new permissions will be added.
- When we use = symbol, previous permissions will be overwritten.

Category	Operations	Attribute
u-user	+assigns permission	r-read
g-group	-remove permission	w-write
o-others	=assigns absolute permission	x-execute
a-all		

# Changing Permissions – Numeric Method

## ➤ Octal notation:

- It describes both category and permission.
- It is similar to = operator (absolute assignment).
  - read permission: assigned value is 4
  - write permission: assigned value is 2
  - execute permission: assigned value is 1
- Example 1:

```
$ chmod 666      note
  ▪ It will assign read and write permission to all.
```

## 11.2 ACL - Access Control List for Files & Directory

- ACL provides an additional flexible permission mechanism for file system on a Linux system.
- It enhances the traditional UNIX file permissions for files & folder. With ACL, you can give permissions for any user or any group with fine-grained access rights.
  
- Set Permissions
  - `setfacl -R -m u:rajesh:rwx /test/demoacl.txt`
- Get(list ACL permissions) Permissions
  - `getfacl /test/demoacl.txt`

# Summary

## In this lesson you have learnt

- How to configure file/directory permissions?
- How to configure fine-grained permissions using ACL?

# Linux Essentials

## Lesson 12 Network Services

# Module Overview

- 12.1 Network configuration
- 12.2 NW Devices
- 12.3 Virtual NW Adapters
- 12.4 NW Diagnostics - ping , traceroute

## 12.1 Network configuration

- Network configuration can be modified using:
  - Using GUI
  - CLI using ifconfig
  - Using system-config-network
  - Modifying configuration files

## 12.2 NW Devices

- Network interfaces in Linux are enumerated as eth[0123... ], but these names do not necessarily correspond to actual labels on the chassis.

e.g) eth0, eth1, eth2

## 12.3 Virtual NW Adapters

Multiple number of virtual network adapters can be added in a Virtual machine.

Virtual network adapters can be configured to directly communicate with the physical network or as private or as NAT or host only connection.

## 12.4 NW Diagnostics - ping , traceroute

- **ping**

- Can be used to test the network connectivity to a remote host

- **traceroute**

- Can be used to find the list of all intermediate hops a packet is travelling to reach the destination

- **host/dig command**

- Can be used to test DNS name resolution

# Summary

## In this lesson you have learnt

- How to configure network using commandline?
- How to add/remove/configure Virtual Network Adapters?
- What are the network diagnostic tools and how to use them?

# Linux Essentials

## Lesson 13 Troubleshooting

# Module Overview

## 13.1 Root Password Recovery

## 13.1 Root Password Recovery

- Root password might get forgotten or no more available
- In these type of cases, one can delete/reset the root password

- 1.Login as single user mode
- 2.Delete/Reset the root password

# Summary

## In this lesson you have learnt

- How to recover from the loss of the root password?