UBUNTU & CENTOS SERVER ADMINISTRATION

This is the keyword-rich, attention-grabbing subtitle

MD. TANVIR RAHMAN

To my niece Safiyah Nawar and Nujaira Zaynab

UBUNTU & CENTOS SERVER ADMINISTRATION

Copyright © 2019 by MD. Tanvir Rahman.

For information contact :
(Address, website)
http://www.tanvirrahman.ml

First Edition: December 2019

CONTENTS

UBUNTU & CENTOS SERVER ADMINISTRATION.....I

INTRODUCTION.....1

VMWARE INSTALLATION.....11

UBUNTU SERVER INSTALLATION.....14

CENTOS SERVER INSTALLATION.....20

COMMAND LINE IN LINUX.....26

BASIC LINUX COMMANDS.....30

IP ADDRESSING.....59

SETTING STATIC IP IN CENTOS7.....71

SETTING STATIC IP IN UBUNTU.....86

PACKAGE MANAGEMENT IN LINUX.....100

COMPARISON BETWEEN TWO PACKAGE MANAGEMENT SYSTEM.....134

YUM SERVER.....137

APT SERVER.....145

KERNEL MANAGEMENT.....157

SSH: THE SECURE SHELL.....158

TELNET.....171

DISK MANAGEMENT.....177

PARTITION ID.....181

CREATE FILE SYSTEM.....183

RAID.....185

CREATING RAID 1 IN CENTOS 7.....203

CREATING RAID 5 IN CENTOS 7.....218

CREATING RAID 10 IN CENTOS 7.....228

COULD BE A SUBTITLE.....242

ABOUT THE AUTHOR.....249

ACKNOWLEDGMENTS.....250

INTRODUCTION

subtitle

An operating system is a software that runs on our computer. Handling all the instructions between a user and the Computer hardware .But the operating system is not just one software it also consists of a lot of other smaller program that runs on this operating system to that helps users to do their work. we run this smaller program on top of this operating system to do everything.

Liunx is just another operating system. Its a rock solid operating system. Linux work both as a server and Desktop operating system.

Linux operating system is great for a lot of reason . thease are the following

* ***Multi user OS:*** Linux is a multi user operating system. That means more than one user can work on a system at the same name

* ***Multi tasking OS:*** Linux is a multi taking operating system you can run multiple program at once .this allows the operating system to run several process all at once.

* ***Multi Platform OS:*** Linux can run currently more that 24 types of platform and 64-bit Intel based personal computer .All variants of Apple mac,Sun Spark and ipod ,even the Microsoft xbox.

* ***Interoprable OS***: Linux can operate with most network protocols and also most Language it can easily interact with Windows OS,NOVEL,UNIX and other operating system that has a smaller market

* ***Scalabel OS***: Linux operating system has support for even Raspberry pi which is a credit card size computer to Very powerful Server. Most of the server of the world is running Linux OS. They have also run in low power computer

* ***Portable OS***: Linux is portable operating system. Linux is mostly written in C programming language .C is a language that is specially for writing operating system level software. And it can be ported to run on a new computer.

* ***Flexible OS***: Linux operating system can be used to make a router,graphical workstation,home entertainment computer,file server,web server,mail server,cluster, just any computing purpose.

* ***Stable***: Linux kernel is very mature. For being stable it is used for most of the server in the world.

* ***Efficient***: The design of the Linux enables you to include on only the thing you needed thats why it can run on both raspberry pi to a big server.

* ***Free*** : Linux is a Free operating system.

GNU PROJECT

GNU Stands for (GNU is NOT UNIX). To make a free clone of the UNIX OS GNU project started 1984.To maintain the free software FSF(Free Software Foundation) is created. It Creates the GNU C compiler ,EMACS Text editor and many other software.

The GNU General Public License (GPL) is a very creative license that used to copyright to protected the freedom of the software user. When a software is licensed under the GPL recipients are bound by the copyright to respect freedom of anyone to use and share the software and also change the source code if necessary.

HISTORY OF LINUX

Linux is a clone of the UNIX based operating system. Unix is created at BEL LABS for AT&T corporation. To make a free clone of the unix Linus Trovalds created a minix .he wrote the kernel which is the heart of the linux .After that a lot of developer helped him to add more feature and functionality .and at that time the GNU Project was making free software for the computer and to make an OS they need a functional kernel which can communicate with the hardware .They took the linux kernel and add the GNU software on top of the kernel and made the GNU/LINUX Operating system.

Linux Trovalds is still considered as the dictator of the Linux kernel. He ultimately determines which feature will be added in the linux kernel and what features are not.

Packaging Linux:Distribution

A complete linux system is called distribution. A linux distribution contains the Linux kernel and the GNU project Tools and any number of software that can make the OS diverse functionality.

There are a lot of distribution on linux .Some of them specifically for servers and some of them are Desktop. Every customized distribution includes software packages for different users.

A single linux distribution often appears in different version .For example CENTOS distribution comes with a full core distribution and a LIVE CD version.

Ubuntu is based on Debian Distribution And Centos is community version of the Commercial RED HAT linux distribution.

Core Linux Distribution

Core linux distribution contains the Linux Kernel and GNU operating system one or more DE(Desktop Environment) and application that is available ready to install and run. The core linux Distribution are the compete linux distribution. These are the popular distribution

* Red Hat linux
* Fedora Core
* Centos Linux
* SUSE linux
* Ubuntu Linux
* Gentoo Linux
* Debian Linux
* Slackware Linux
* Mandriva Linux
* Turbo Linux
* Puppy Linux

we use the ***UBUNTU linux*** and the ***CENTOS linux*** to illustrate how the servers work.

PRINCIPLE OF LINUX

* Everything works as a file , even the system hardware

* Small work is done by the individual program

* Any complected work will be divided into smaller part and then process this by different different module.

* All the configuration will be stored in a text file

* linux OS use a standard hierarchical file structure in which the files/user files are arranged

* * *

UBUNTU SERVER

Ubuntu is built on the Debian architecture and comprised linux server and Dsktop. Ubuntu release updates every six months Ubuntu packages are based on packages from Debians unstable branch. Ubuntu is currently funded by the Canonical LTD.And GENOME 3 is the default GUI interface for the ubuntu from 17.10 version. We are goinf to use Ubuntu Server 18.04.3 LTS for our work.you can download the latest long term varsion of ubuntu server in this URL

https://ubuntu.com/download/server

CENTOS SERVER

Ubuntu is built on the Debian architecture and comprised linux server and Dsktop. Ubuntu release updates every six months Ubuntu packages are based on packages from Debians unstable branch. Ubuntu is currently funded by the Canonical LTD.And GENOME 3 is the default GUI interface for the ubuntu from 17.10 version. We are going to use Ubuntu Server 18.04.3 LTS for our work.you can download the latest long term varsion of ubuntu server in this URL

https://ubuntu.com/download/server

VMware Installation

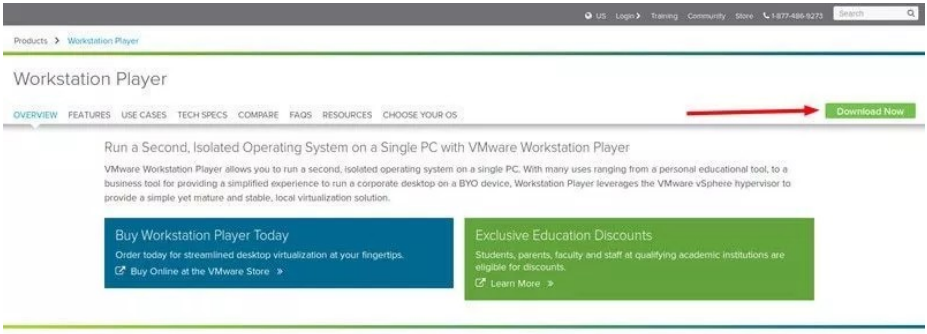
Step 1:

install the required build packages

=> sudo apt install build-essential

Step 2:

Download VMware workstation player from the website.



Step 3:

go to the installed directory make the file executable

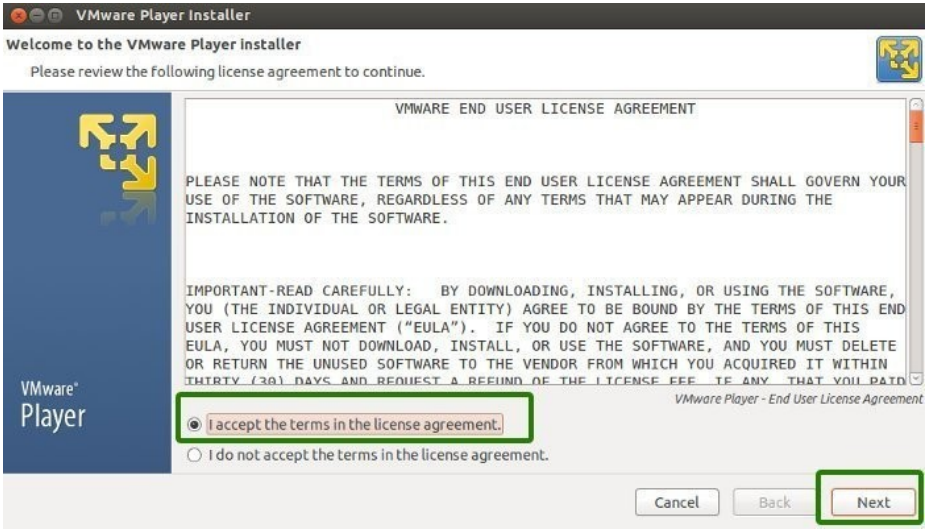
=> chmod 777 Vmware-Player*

[we will talk about the chmod 777 later for now just use it]

Step 4:

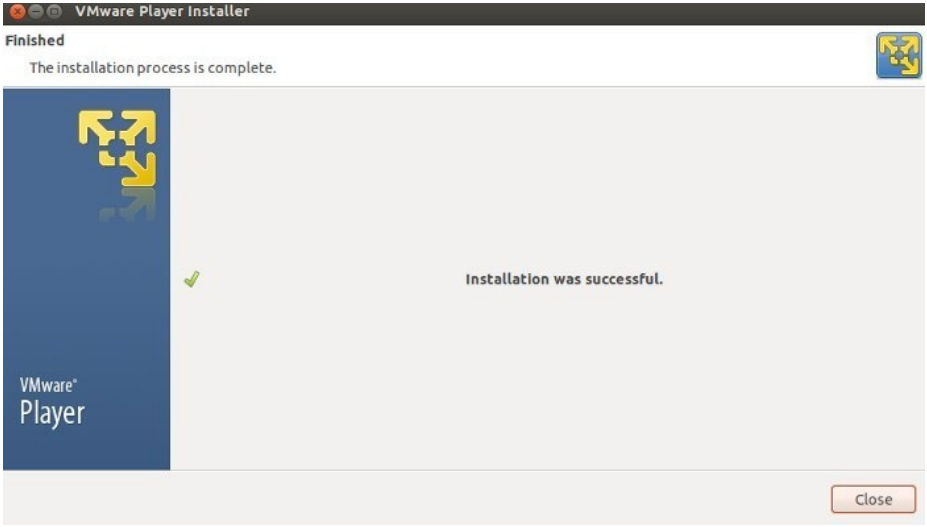
execute the program with **sudo**

=>sudo ./Vmware-Player*



[no license key is required .If you want to install vmware workstation instead of vmware player you need to have the license key]

after a successful installation screen will show to you


VMware Player Installer
Finished
The installation process is complete.
Installation was successful.
VMware
Player
Close

* * *

Ubuntu Server Installation

Ubuntu

Requirements:

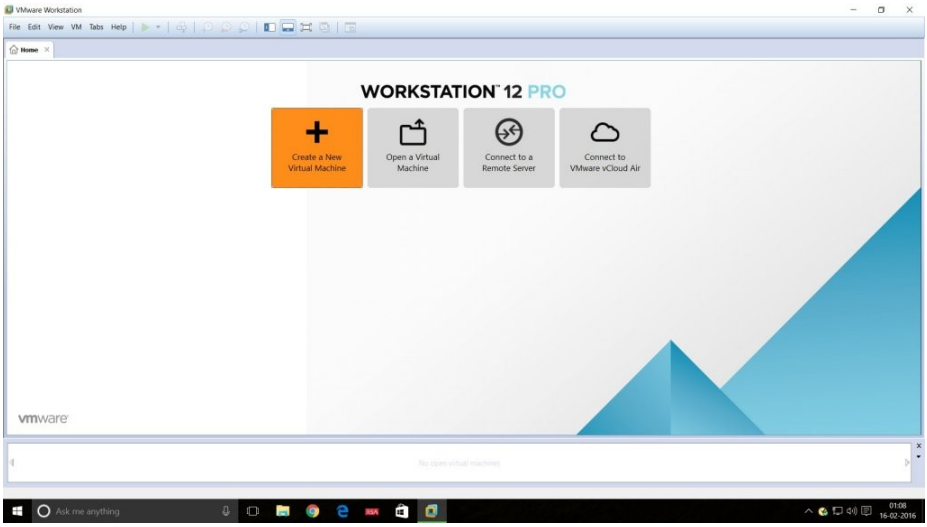
1)Host PC with at least 4GB of ram

2)VMware Workstation/Virtualbox

3)Ubuntu server ISO image

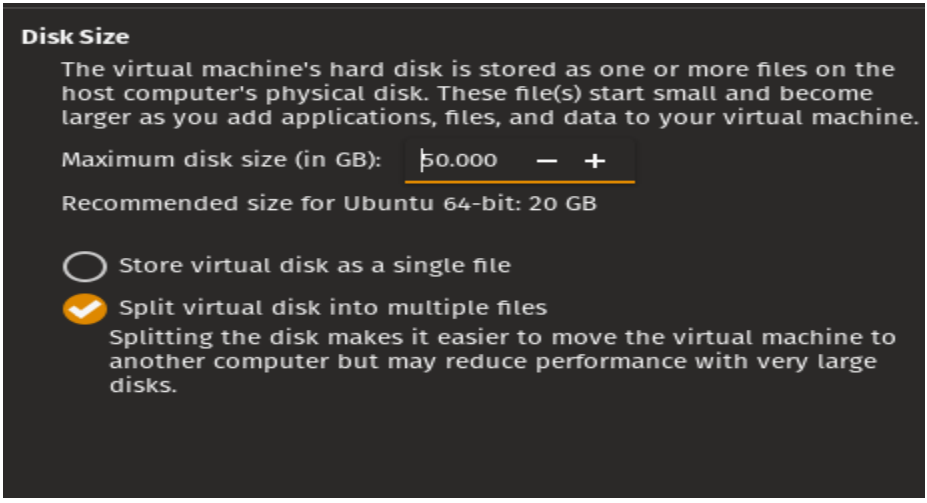
Step 1:

Lunch VMware Workstation New Virtual Machine Wizard



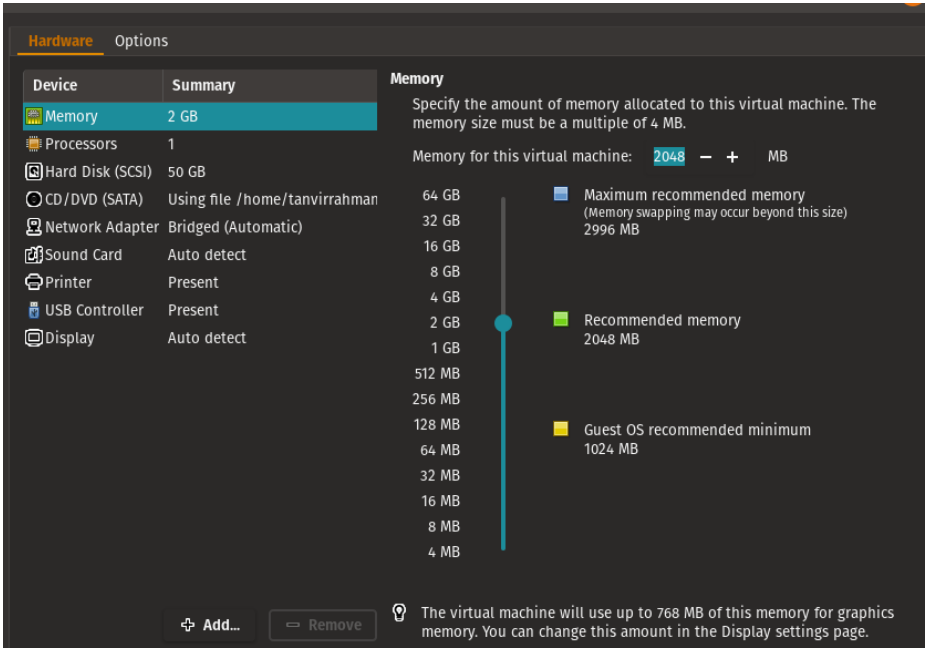
Step 2:

Select the installation media or source and choose the disk size.



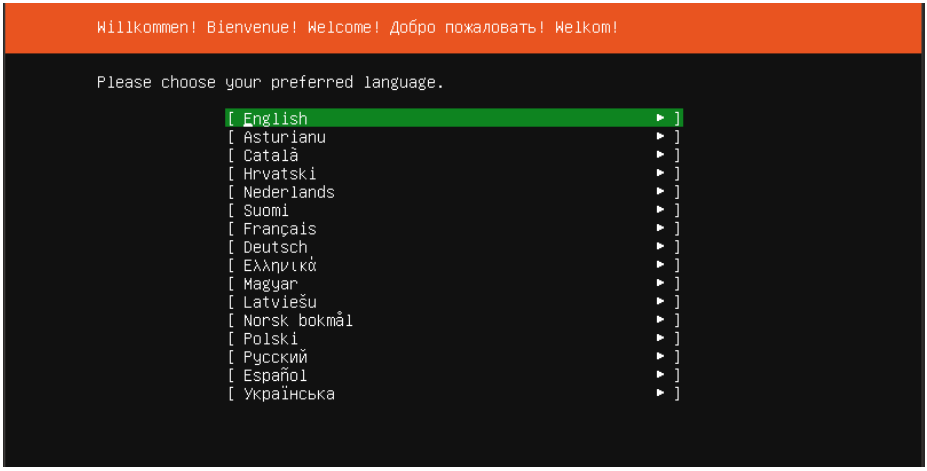
Step 3:

final configuration of the Vmware will be like this



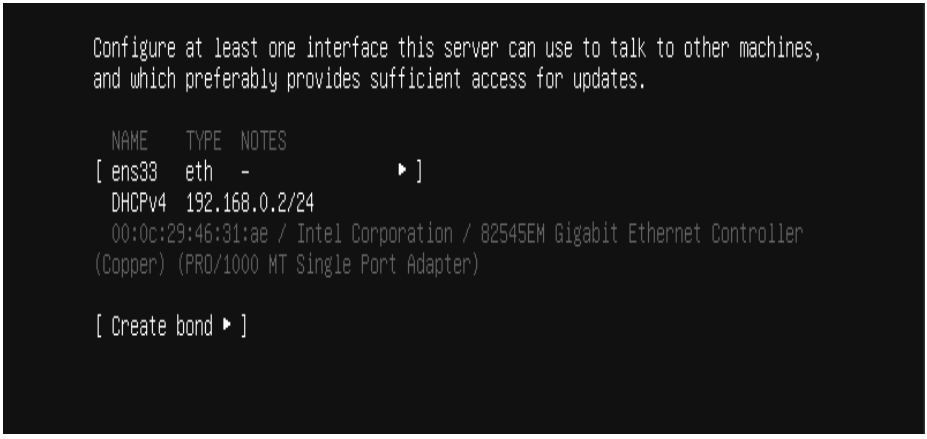
Step 4:

start the installation ,first set the language



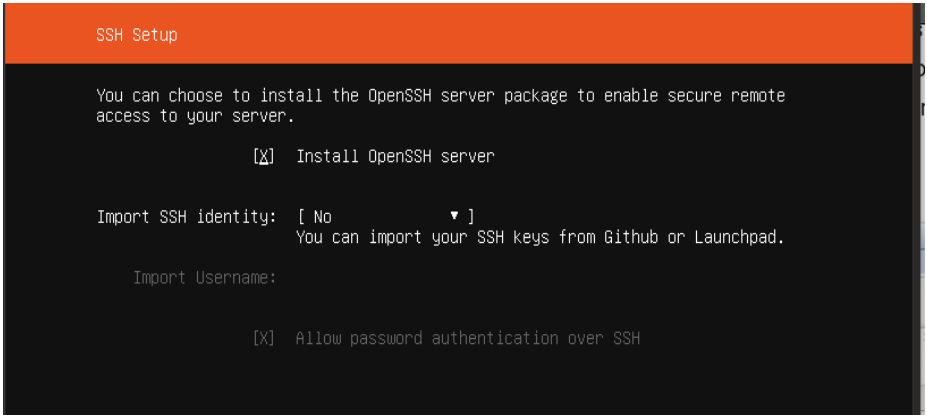
Step 5:

select DHCP for network for now .we change the ip address of the server later



Step 5:

Select “install openSSH server” so we can connect to the computer with our hosts



Step 6:

There are three types of partition

* **Guided :** use entire disk : it use the entire disk with guided partition system

* **Manual :** In manual partition user have to allocate the space manually.

For minimal settings three partition is a mandatory

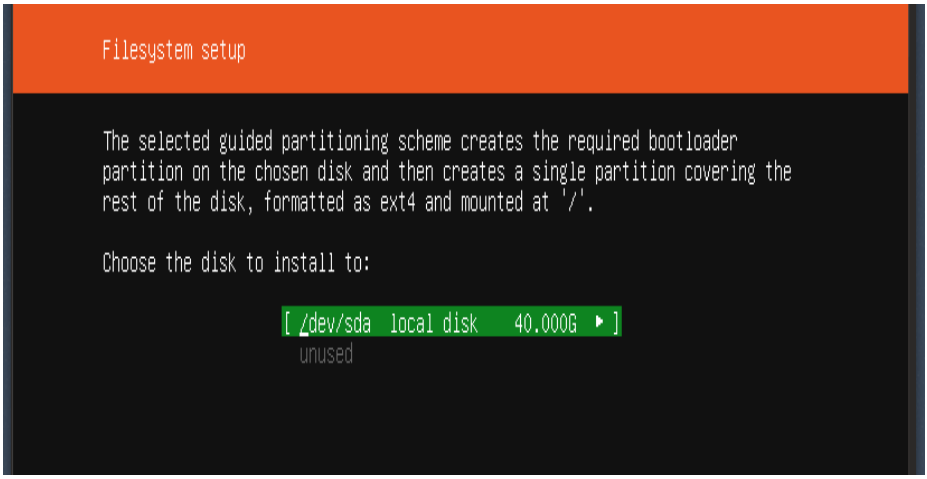
1) /boot

2) /swap

3) /root

* **Guided :** on LVM :this option allow user to set a LVM based partition

Select the entire disk for installation .we select the Entire disk guided partition will talk about the other boot system later.



Step 7:

login to the system with your credential .

SUMMARY:

we learn how to set up a ubuntu server on Vmware Virtual machine with a dhcp network. Using guided partition.

⁂

CENTOS Server Installation

Centos

Requirements:

1)Host PC with at least 4GB of ram

2)VMwareWorkstation/Virtualbox

3)CENTOS7 server ISO image

Step 1:

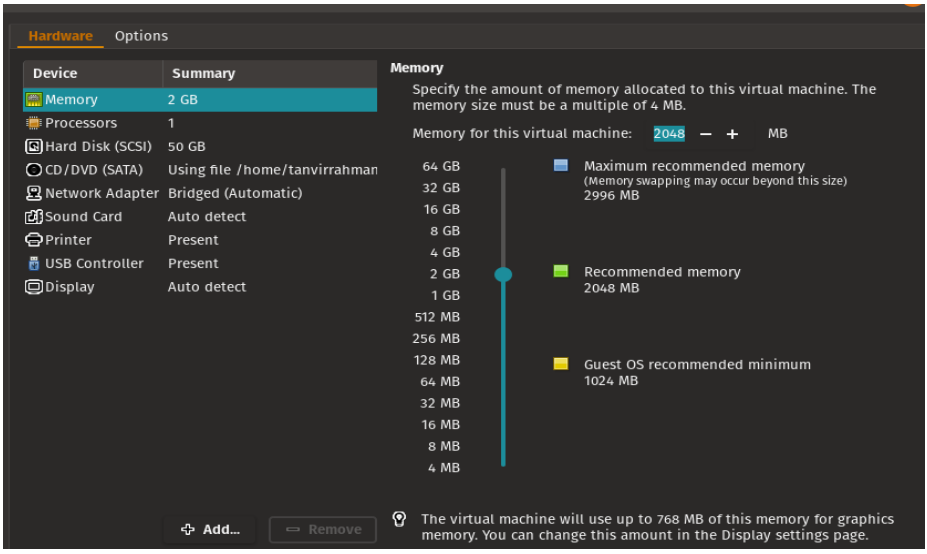
Lunch VMware Workstation New Virtual Machine Wizard

Step 2:

Select the installation media or source and choose the disk size.

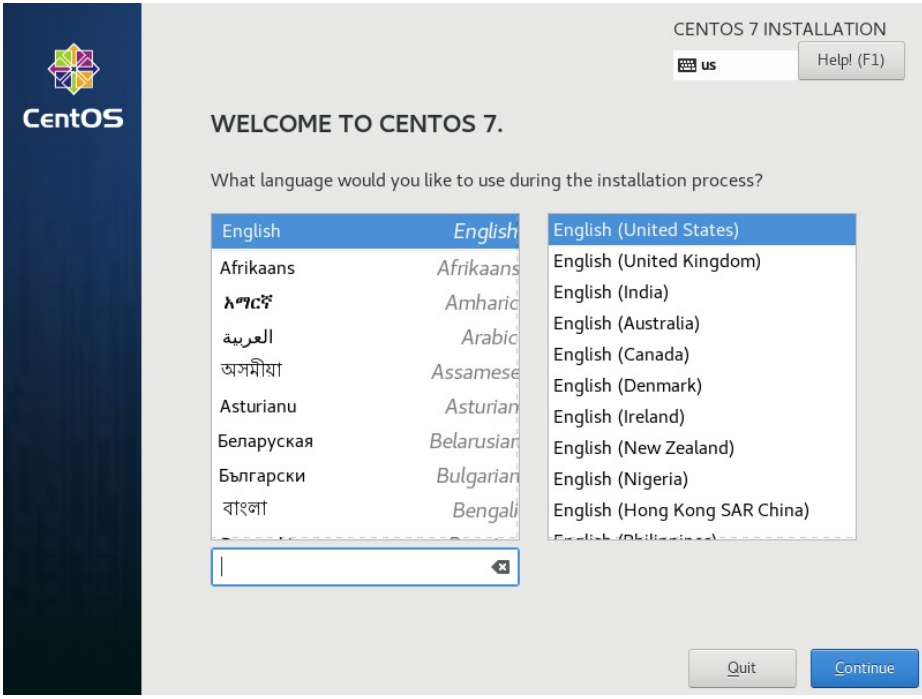
Step 3:

final configuration of the Vmware will be like this



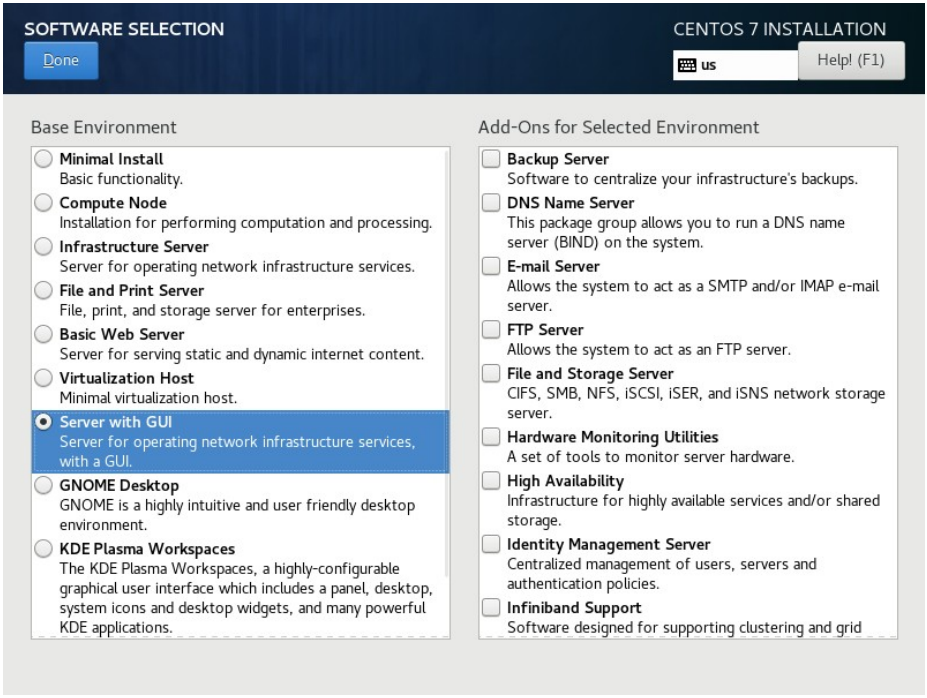
Step 4:

start the installation , First set the language



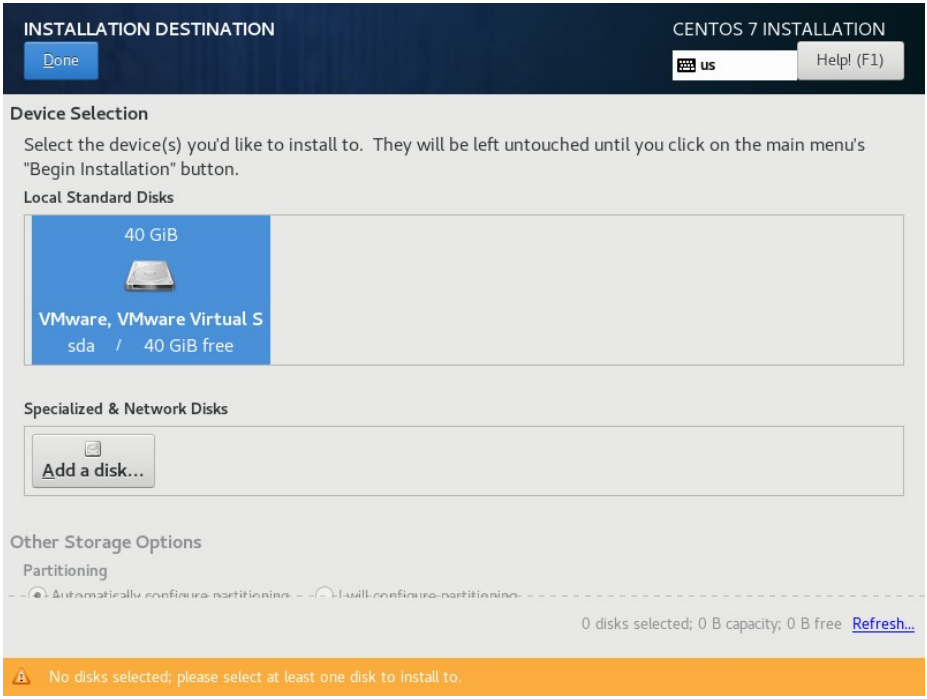
Step 5:

From the **software selection** select server with a GUI



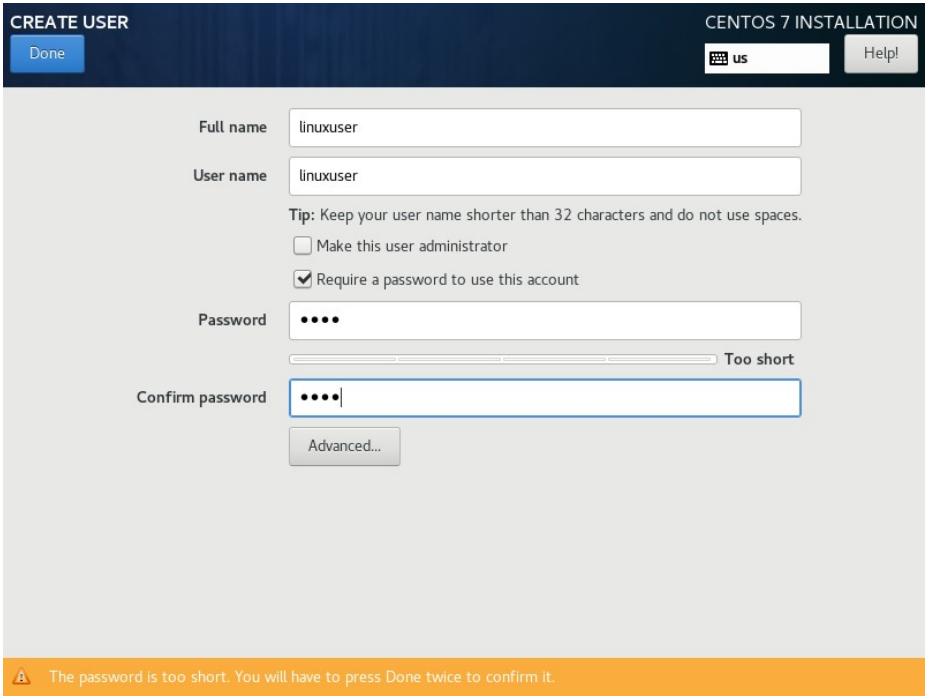
Step 6:

Select the volume for installing .in this installation we go for the entire disk guided partition



Step 7 :

Enter the root password and confirm it. and create a user and set password for the user



Step 7 :

wait for the installation to finished. After that reboot the system

Step 8 :

login with root credentials

SUMMARY:

we learn how to set up a Centos server on Vmware Virtual machine with Using guided partition.

***

COMMAND LINE IN LINUX

Every workstation version of linux has a Beautiful GUI(Graphical User Interface) but most of the server Ubuntu ot CENTOS run on command line mode.you can add graphical user interface to that but without the command line you cant manage the server properly. Once you learn Command line you will find that it is more powerful and flexible for user to manage your server than a graphical mode.

WORKING AS ROOT

By default every linux OS creates a user root.Many operating system like CENTOS ask for a root password .But Ubuntu server don't do that .There is a very good reason behind that .in Linux OS root has a limitless power .root can do anything ,change anything even can delete anything from the server .so it is very important to be careful when you work as

root .Thats why in ubuntu server every time you do anything that need superuser privileges you use the command '***sudo***' .This command allows the normal user so they can peform task that needs superuser privileges.you type sudo then your command it may ask for password after providing the password it will perform the action with the superuser privileges . But if you want the root shell then type this command

=> sudo su

But it is not recommended to work with the root shell .Do not use the root shell unless it is absolutely necessary . work with sudo if you need superuser privilege.

TERMINAL & SHELL

What is Terminal:

Terminal is a program that opens a window and lets you play with the shell. There are a bunch of different terminal emulators that you can see in the linux Distribution. such as Gnome terminal, konsole ,xterm, rxvt .nxterm ,eterm, Tilix etc. Terminal lets you interact with the shell

What is shell :

Shell itself is a program that takes command from the keyboard and gives them to the operating system to perform. You can work with the graphical user interface but if the server you are using has no graphical user interface this will be the only interface you got and you have to do all of your work in the CLI interface

there are different types of shell

* ***tsh*** → ***tsh*** is a shell with a scripting language similar to the C programming language

* ***sash*** → stand alone shell .its a very minimal shell runs almost

every system .it is basically popular for troubleshooting the system

* ***zsh →*** ***zsh*** is a shell which is compatible with bash but has a lot of extra functionality

* ***fish →*** ***fish*** stands for friendly interactive shell .mostly popuar in desktop. It has a very good auto completion feature

* ***Bash →*** ***bash*** stands for Bourne Again Shell that is the enhanced version of the original UNIX program ***sh*** .it is written by the Steve Bourne. It is the most populer shell and the default shell of the most linux operating system.

We will use the Bash all over the example.

***

BASIC LINUX COMMANDS

These are the basic command to operate a linux operating system

Command name: ***ls***

description:

ls command used to see the files and folder inside a directory . it is the most used command in linux.

syntax:

ls -[option] <directory>

1) ***ls -m*** will show the files and folder with comma

2) ***ls -a*** show the hidden files also

3) ***ls -l*** will show the files and folder in a listing format

4) ***ls -lh*** will show the file with listing and size

5) ***ls -i*** will show the list of files and folders with Inode

6) ***ls -t*** will show the modification time with directory listing

example:

[pic]

Command name: ***more***

description:

It works like the more command .it also give scrolling options

syntax:

More <options> <file_name>

1) ***less -E*** : automatically exit the first time it reaches end

of file.

2) ***less -f*** : forces non-regular file to open.

3) ***less -F*** : exit if entire file can be displayed on first screen

4) ***less -g*** : highlight the string which was found by last search command

5) ***less -G*** : suppresses all highlighting of strings found by search commands

6) ***less -i*** : cause sears line numbers

7) ***less -p <pattern>*** : it tells less to start at the first occurrence of pattern in the file

8) ***less -s*** : causes consecutive blank lines to be squeezed into a single blank line to ignore case

9) ***less -n*** : suppresses line numbers

10) ***less -p <pattern>*** : it tells less to start at the first occurrence of pattern in the file

11) ***less -s*** : causes consecutive blank lines to be squeezed

12) ***less -N*** : shows line number

example:

[pic]

Command name: ***strings***

description:

To display the content of the file

syntax:

strings <filename>

example:

[pic]

Command name: ***tree***

description:

To display the Directory stricture in a tree format

syntax:

Tree <directory>

[you may have to tool with package manager]

example:

[pic]

Command name: ***dir***

description:

To display the files and folder inside the directory

syntax:

dir <directory_name>

[you have to install 'tree' tools before using this command]

example:

[pic]

Command name: ***cal***

description:

To display the calendar

syntax:

Cal

cal <year>

cal <month> <year> command]

example:

[pic]

Command name: ***clear***

description:

clear the screen

syntax:

clear

example:

[pic]

Command name: bc

description:

basic calculator

syntax:

bc

example:

[pic]

Command name: mkdir

description:

making directory

syntax:

mkdir <directory> : for making single directory

mkdir -p <directory/directory>:

for making recursive directory

example:

[pic]

Command name: rmdir

description:

Remove empty directory

[you cant remove any directory which has file in it with

this command]

syntax:

rmdir <empty_directory>

example:

[pic]

Command name: file

description:

display the file type

syntax:

file <filename>

example:

[pic]

Command name: ln

description:

Create a link of the source filename.In case in hard link if you delete the main file link wont remove but in case of the soft link if you delete the main file the the linked file will be removed

syntax:

ln <option> <source_file> <shortcut_file>

ln -s : for creating soft link

ln -P : for creating hard link

example:

[pic]

Command name: ***history***

description:

Shows users command history it will show the last 1000 command of the user you can set the limit if you like

syntax:

history

example:

[pic]

Command name: ***locate***

description:

It will search the entire system for that file [you need to apply the command '***updatedb***' for getting latest entry]

syntax:

Locate <file_name>

example:

[pic]

Command name: ***uname***

description:

Show all the information about the kernel , OS and hardware-platform

syntax:

uname -a : all information, in the following order
uname -s :print the kernel name
uname -n : print host name
uname -r : print the kernel release
uname -v : print the kernel version
uname -m : print the machine hardware name
uname -p : print the processor type
uname -i : print the hardware platform
uname -o : print the operating system

example:

[pic]

Command name: ***tar***

description:

For creating archive and extracting archive hardware-platform

syntax:

tar -cvf <archive_name> <source> : for creating archive

tar -xvf : for extracting archive

example:

[pic]

Command name: ***gzip***

description:

For compressing normal file or archive file

syntax:

gzip <file_name>

example:

[pic]

Command name: ***gunzip***

description:

It is used for uncompromising a compressed file

syntax:

gunzip <compressed_file>

example:

[pic]

Command name: ***lsmod***

description:

Show a list of the modules used by the kernel

syntax:

lsmod

example:

[pic]

Command name: ***rmmod***

description:

Delete any module used by the kernel
[not Recommended . don't do it unless you are absolutely sure what you are doing]

syntax:

mmod <module_name>

[you need to be a root user to perform this action]

rmmod-f , forces a module unload and may crash your machine. This requires Forced Module Removal option in your kernel. DANGEROUS

rmmod -v, enables more messages

rmmod -V, show version

example:

[pic]

Command name: ***modprobe***

description:

Adding new module to the system

syntax:

modprobe <module_name>

example:

[pic]

Command name: ***ps***

description:

See the current running process of the system

syntax:

ps

example:

[pic]

Command name: ***top***

description:

Top command is used for process monitoring. [more information about top in Process management]

syntax:

top

example:

[pic]

Command name: ***renice***

description:

Used for changing the priority of a process running on a system. [more info in process management chapter]

syntax:

renice -n <priority> -p <pid

example:

[pic]

Command name: ***kill***

description:

Used for terminating process for this purpose

syntax:

Kill -<sigterm> -p pid

example:

[pic]

Command name: ***uptime***

description:

Shows the system's running time. and load averages of previous 1 minute ,5 minute and 15 minute.

[this information can be found in top and htop command also]

syntax:

uptime

example:

[pic]

Command name: ***iostat***

description:

Shows the Cpu and I/O information

[more information in process management Devices]

syntax:

1) ***Iostat -c*** : generate cpu status only

2) ***iostat -d*** : generate I/O statistics for all the devices

3) ***iostat -x*** : generate detail I/O statistics

4) ***iostat -x*** : generate detail I/O statistics and CPU information

5) ***iostat -p <devices>*** : generate details for that specific devices

6) ***iostat -m*** : generate statistics in Megabyte

7) ***iostat -k*** : generate statistics in Kilobyte

8) ***iostat -N*** : generate LVM options

9) ***iostat -t*** : generate statistics with timestamp

10)***nfsiostat*** : Shows information of NFS devices

example:

[pic]

Command name: ***hostnamectl***

description:

Display hostname and its related settings also change hostname and its related settings

syntax:

Hostnamectl : provide information about current host and its properties

hostnamectl set-hostname <hostname> :It will change the hostname

example:

[pic]

Command name: ***pwd***

description:

Print the current directory path

syntax:

pwd

example:

[pic]

Command name: ***dmesg***

description:

Display the detected hardware status during boot time [the file location is ***'var/log/dmesg'***]

syntax:

dmesg

example:

[pic]

Command name: ***init***

description:

Display the detected hardware status during boot time

[the file location is ***'var/log/dmesg'***]

syntax:

Init <run_lavel>

0 :Power-off the machine

6 :Reboot the machine

2, 3, 4,5 :start runlevel X.

1, s, S :Enter rescue mode

q, Q : Reload init daemon configuration

u, U :Reexecute init daemon

example:

[pic]

Command name: mkswap

description:

Used to format the partition used for swap space

syntax:

mkswap <file_system>

example:

[pic]

Command name: swapon

description:

To activate the swap space

syntax:

swapon -a <file_system>:

[enable all swaps from ***/etc/fstab***]

example:

[pic]

Command name: ***swapoff***

description:

To deactivate the swap partition

syntax:

swapoff <file_system>

example:

[pic]

Command name: ***mkfs***

description:

To format the partition this tools is used

[more information about file system]

syntax:

mkfs -t <fs_type> <file_system>

To format the partition this tools is used

[more information about file system]

1) ***mkfs.ext2 /dev/sdx***: for ext2 file system
2) ***mkfs.ext3 /dev/sdx***:for ext3 file system
3) ***mkfs.ext4 /dev/sdx***: for ext3 file system
4) ***mkfs.minix /dev/sdx*** :for minix file system
5) ***mkfs.xfs /dev/sdx*** :for xfs file system

example:

[pic]

Command name: ***poweroff***

description:

power off the machine

syntax:

poweroff

example:

[pic]

Command name: ***whoami***

description:

Display the username which is currently logged in

syntax:

whoami

example:

[pic]

Command name: wc

description:

Used to find out number of lines, word count, byte and characters count in the files specified in the file arguments

syntax:

1) ***wc <file_names>***
2) ***wc -m <file>*** : print the character in in the file
3) ***wc -w <file>*** : print the word in in the file
4) ***wc -l <file>*** : print the line in in the file

example:

[pic]

Command name: w

description:

Used to show who is logged in to the computer and what they are doing

syntax:

w

example:

[pic]

Command name: ***arch***

description:

Display the computer architecture

syntax:

arch

example:

[pic]

Command name: ***alias***

description:

Instructs the shell to replace one string with another string while executing the commands

syntax:

Alias <string>='<target straing>'

example:

[pic]

Command name: ***bg***

description:

Used to send any foreground job to background

syntax:

bg

example:

[pic]

Command name: ***cp***

description:

Used to copy a file or a group file from one destination to other

syntax:

cp <source_file> <target_destination>

example:

[pic]

Command name: ***echo***

description:

Used to display line of text/string that are passed as an argument

syntax:

echo <arguments>

example:

[pic]

Command name: ***fdisk***

description:

Format disk as well as creating and manipulating disk partition table
[more information in disk management chapter]

syntax:

fdisk <file_system>

example:

[pic]

Command name: ***cfdisk***

description:

Format disk as well as creating and manipulating disk partition table using a text based GUI interface
[more information in disk management chapter]

syntax:

sudo cfdisk

example:

[pic]

Command name: ***lsblk***

description:

Displays the total amount of free space available along with the amount of memory used and swap memory in the system

syntax:

lsblk

example:

[pic]

Command name: ***lsmod***

description:

List the current kernel modules that are currently loaded

[it actually print the content of the '/proc/modules' with a nice format]

syntax:

lsmod

example:

[pic]

Command name: ***lspci***

description:

Display the information about the currently connected PCI Buses .

[list of devices that are connected to the computer]

syntax:

lspci

example:

[pic]

Command name: ***lshw***

description:

List all the Details information of the hardware of the computer

syntax:

lshw

example:

[pic]

Command name: ***lshcpu***

description:

Display the detailed information about the CPU

syntax:

lscpu

example:

[pic]

Command name: man

description:

Display the reference of the tools or command that are you using

syntax:

man <command>

example:

[pic]

Command name: sudo

description:

give you the superuser privileges

syntax:

sudo <command>

example:

[pic]

Command name: ip

description:

Used for performing several network administration

tasks

syntax:

Ip <option> <command>

example:

[pic]

Command name: ***touch***

description:

Create an empty file

syntax:

touch <file_name>

example:

[pic]

Command name: ***ifconfig***

description:

shows the ip address related information

syntax:

ifconfig

example:

[pic]

Command name: ***gerp***

description:

global regular expression used for searching keyword

syntax:

ls | grep initrd

example:

[pic]

Command name: ***wget***

description:

interactive cli based downloader

syntax

wget <download_url>

example:

[pic]

Command name: ***reboot***

description:

reboot the system

syntax

reboot

example:

[pic]

Command name: ***ping***

description:

test any host or network which is alive physically and logically

syntax

ping <pi_address/domain_name>

example:

[pic]

These are the basic commands to run a linux system .There are a lot of command more to maintain the server.

***

IP ADDRESSING

COMPUTER NETWORK

A computer network is a group of computer and other computing peripherals that linked together through some kind of communication channels to communicate with each other and share their resources among a width range of users.

Their jobs are

1) Facilitate communication via email,file server,web server,instant messaging etc
2) Share resources of the hardware like printer or scanner
3) Enable File sharing
4) create a centralized control among the total network

TYPES OF COMPUTER NETWORK

Network Basically divided into three groups:

1) Local Area Network (LAN)

2) Metropolitan Area Network (MAN)

3) Wide Area Network (WAN)

LAN

A local area network (LAN) within a small area like home,school,office or group of buildings. They can share their resources and device like printer and scanner and data storage. Most of them are centrally organized. And because of the type of the communication the data transfer rate is very high. And local area network does not need any leased communication line

MAN

A metropolitan area network (MAN) spans an entire campus by connecting multiple LAN. MAN is larger than the LAN, because it consists of a number of LAN .MAN works like more of a ISP but it does not owned by a single organization. instead MAN provides a shared network connection to all its users

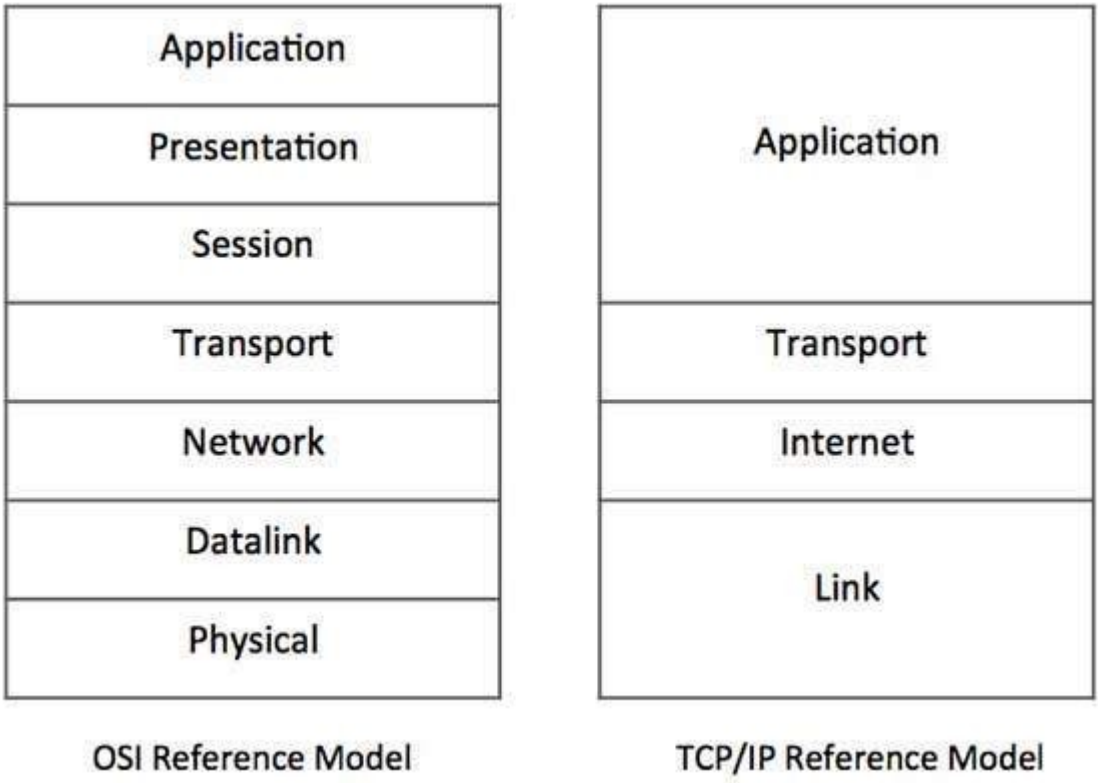
WAN

A wide area network (WAN) within a large scale of geographical area is called WAN. It is created by connecting different LAN from a long distance. And the transmission speed generally is slower than the LAN or MAN but the data transfer rate is increasing .

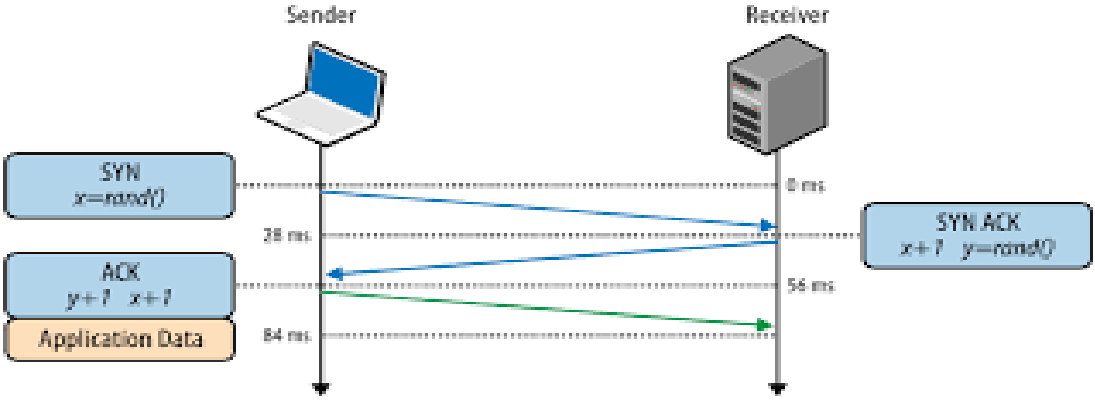
TCP/IP PROTOCOL SUITE

A majority of the internet users use a protocol suite called Internet protocol suite which is also known as the TCP/IP protocol suite.The two protocols are ***TCP*** (Transmission

control protocol) & IP (internet protocol).In here TCP is a connection oriented protocol means it transmit data in a sequence and it has a acknowledgment process. If the acknowledgment are not recived



then the data will be re transmitted it can guarantee the delivery of the data to the host and IP is used to maintain the address of the specific host.



IP Addressing

IP addressing is the most important topic in the networking.ip address is basically a numeric identifier that used to identify a Machine .Ip address is a software address not a hardware address that means it can change depending on the network you are connected.The hardware address is the NIC address thats called the Physical address that cant be changed.

Important Clement of a IP address

Bit: Bit is one digit either 0 or 1

Byte: made up with 8 bits its just a ordinary 8 bit binary number.

Network Address : Network address is used send packets to the network .for example 10.0.0.0,192.168.0.0 etc

Broadcast Address : It is used by the host to send information to all the nodes on a network. The address are like 192.168.0.255,172.166.255.255

Every ip address there are two different parts

1) Network part
2) Host Part

Every ip address gives the information about the network and the hosts

Subnet Mask

A subnet mask is a 32 bit umber that masks an ip address and devides the ip address to a network address and hosts address.
Is is done by setting all the network bits to '1' and setting hosts bit to '0'

[Two host ip address are reserved for special purpose The '0'

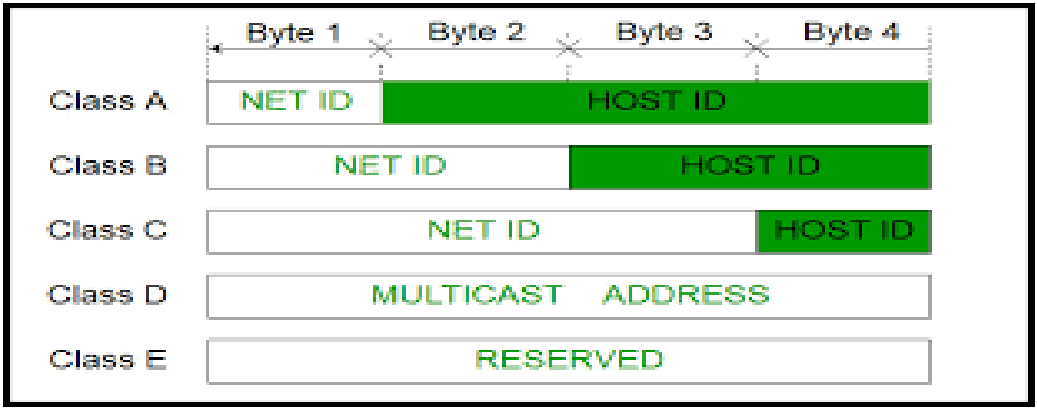
address and the '255' address.the '0' address is reserved for the Network .so if any ip address have a '0' on its last its a network address. and '255' is the broadcast address they cant be assign to a host]

5 types of IP address:

1) class A ip address
2) class B ip address
3) class C ip address
4) class D ip address
5) class E ip address

Class A ip address:

In class A ip address the first byte is reserved for the network address and three remaining bytes are for the hosts.
[it starts with 0.0.0.0 and ends with 127.255.255.255]
subnet mask: 255.0.0.0



It has a small network with huge number hosts.

Class B ip address:

In class A ip address the first two bytes is reserved for the network address and two remaining bytes are for the hosts. More network less hosts
[it starts with 128.0.0.0 and ends with 191.255.255.255]
subnet mask : 255.255.0.0

Class C ip address:

class C ip address the first three bytes is reserved for the network address and remaining one bytes are for the hosts. If you need a lot of network and small number of hosts in every networks class C ip address is used.
[it starts with 192.0.0.0 and ends with 223.255.255.255]

subnet mask : 255.255.255.0

Class D ip address:

class D ip address is a special address. Its called a multicast address. It is basically used for finding router [it starts with 224.0.0.0 and ends with 239.255.255.255]

Class E ip address:

Reserved for the Scientific Experiment

Private IP address:

Not all the address of these class is used for public network .some are not routable through the internet.private ip address is used in the Localy and a local ip address can connect to the internet through a public ip address with NAT (Network address translation).NAT allows a public address to the internet

Class	Address Range	Default Subnet Mask
A	10.0.0.0 - 10.255.255.255	255.0.0.0
B	172.16.0.0 - 172.31.255.255	255.255.0.0
C	192.168.0.0 – 192.168.255.255	255.255.255.0

Loopback address

Loopback address is used to test the communication on a local NIC (Network Interface Card) .Data packets are sent by the node in the loopback address are re-routed back into the same node.It is used for testing the connected physical network. it also enables the user to test an application with an instance of server and client on the same machine.we call it ***localhost***

it starts with 127.0.0.0 and ends 127.255.255.255

Ping

ping stands for ***Packet Internet Gopher*** is a ICMP echo request and reply message that used to check the physical and logical connectivity of the machine on a internet network.

Traceroute

Traceroue is used to find the path of the packet traverses through the internet.

***

SETTING STATIC IP IN CENTOS7

EASY WAY

Every Server needs to have a network connection. without a static ip address you cant run a server .Giving a server a static ip address is the most important thing to do.

When you install a server the most of the time your installer automatically configure your server network and gets the ip address from a DHCP server. But to run a server you need a static ip address. So we need to change its network from DHCP to static and give the server a static ip address .Here we talk about how to give static ip address to a centos7/Redhat7 server.

There are multiple way to give server static address ,Here we talk about easy method

first step

you need to select a static ip address , subnet mask and the gateway that you give your machine .according to your network specifications.

In his example we used a virtual centos7 box . And we give the following ip address subnet mask ,gate way and DNS

IP ADDRESS : 192.168.0.10

SUBNET MASK: 255.255.255.0

GATEWAY:192.168.0.1

DNS: 8.8.8.8

second step

you need to find the network interface that you give the static ip address
A Server can have multiple network interface.

In our virtual machine there are two network interface. We can see the interface from this command

=>***ifconfig***

or

=> ***ip address show***

result:

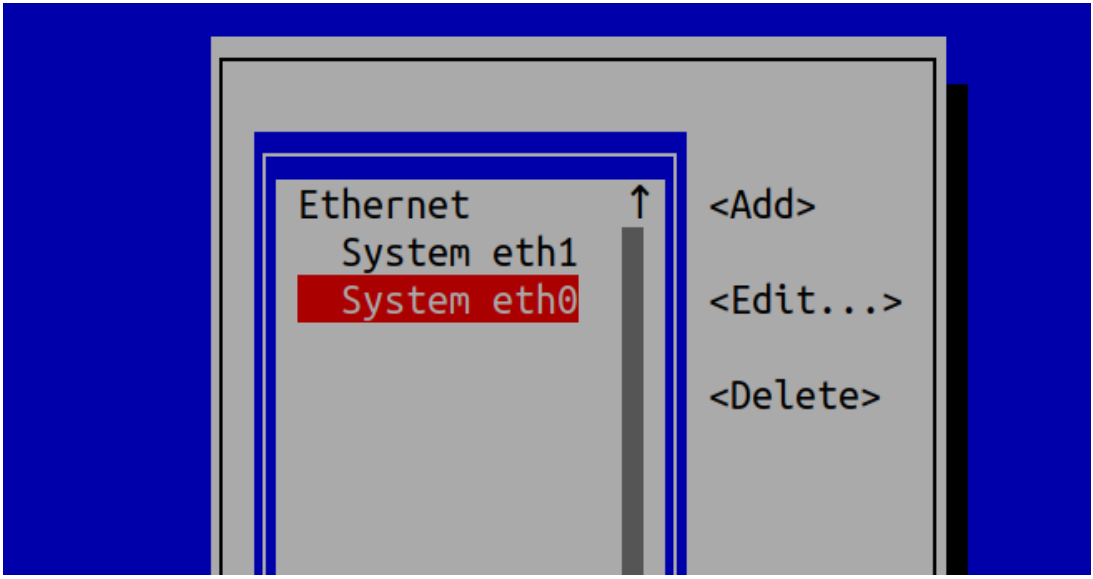
we are currently connected to the server with a ssh connection through eth0.So we cant change the ip address to eth0.this will disconnect the ssh connectivity .we are going to give the static ip address to the eth1 interface

Third step

use the nmtui command and you have to be root to give this command

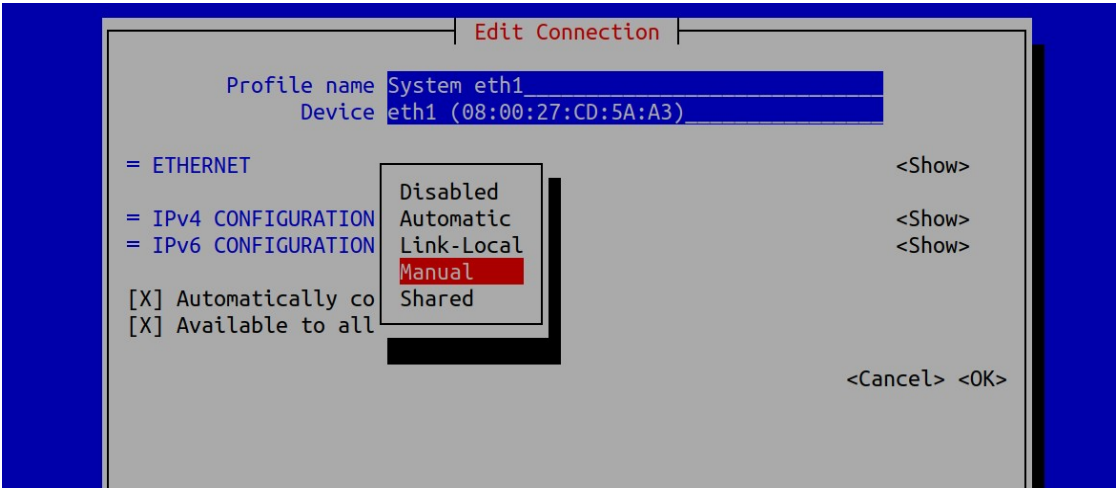
=>***sudo nmtui***

After giving this command this screen appear. From there Select The ***"Edit a connection"***



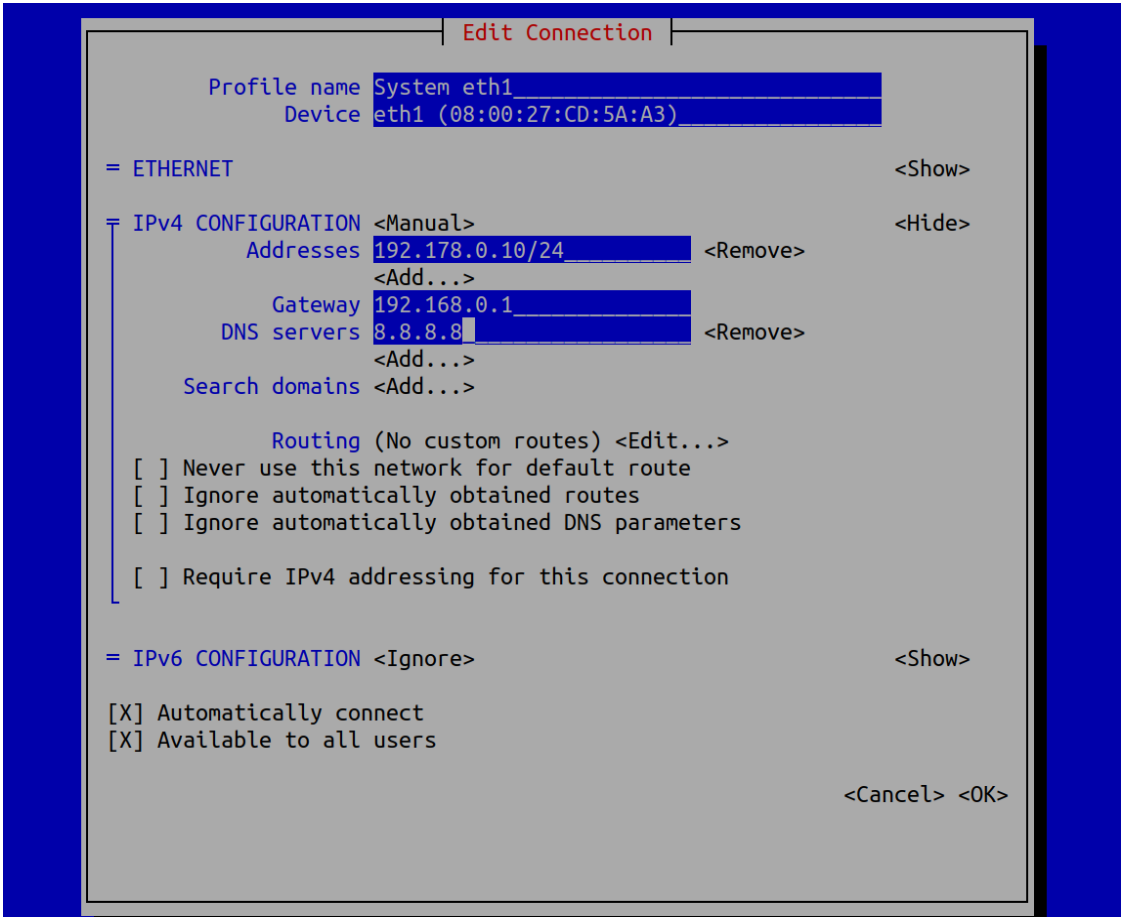
Fourth step

it will show you all the interface .choose your interface in this case we will choose eth1.



Fifth step

we choose the ipv4 and from the option we choose ‘manual’ and Edit the menu



Sixth step

we give the ip address.we have to give the subnet mask with CIDR notation.

Gateway and the The DNS address and click ok. Then quit the program.

Seventh step

if we see our ip address we can see the the ip address still dont change.to make the change we need to restart the interface.

We shutdown the interface with this command

=>***sudo ifdown eth1***

Then we start the interface again

=>***sudo ifup eth1***

Eighth step

Then if we check ip address using

=>***ifconfig eth1***

```
[vagrant@tanvir ~]$ ifconfig eth1
eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.178.0.10  netmask 255.255.255.0  broadcast 192.178.0.255
        inet6 fe80::a00:27ff:fecd:5aa3  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:cd:5a:a3  txqueuelen 1000  (Ethernet)
        RX packets 62  bytes 5854 (5.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 24  bytes 2452 (2.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[vagrant@tanvir ~]$
```

we can see the ip address changed .

Ninth step

We have to test the connection via pinging a network.

=>***ping 8.8.8.8***

```
[vagrant@tanvir ~]$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=63 time=80.2 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=63 time=102 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=63 time=123 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 80.248/101.916/123.156/17.519 ms
[vagrant@tanvir ~]$
```

So the connection is up and running. Thats is the easy way of giving an ip address to a cenos7/Redhat7 server a static address.

TRADITIONAL WAY

first step

you need to select a static ip address , subnet mask and the gateway that you give your machine .according to your network specifications.

we give the following ip address subnet mask ,gate way and Dns

Ip address : 192.168.0.10

subnet mask: 255.255.255.0

Gateway:192.168.0.1

DNS: 8.8.8.8

second step

you need to find the network interface that you give the static ip address
A Server can have multiple network interface.

In our virtual machine there are two network interface. We can see the interface from this command

=>***ifconfig***
or
=> ***ip address show***

```
[vagrant@tanvir ~]$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::5054:ff:fe8a:fee6  prefixlen 64  scopeid 0x20<link>
        ether 52:54:00:8a:fe:e6  txqueuelen 1000  (Ethernet)
        RX packets 1110  bytes 135804 (132.6 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 940  bytes 149277 (145.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.5  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::a00:27ff:fecd:5aa3  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:cd:5a:a3  txqueuelen 1000  (Ethernet)
        RX packets 13  bytes 1362 (1.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 16  bytes 1826 (1.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 32  bytes 2592 (2.5 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 32  bytes 2592 (2.5 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

we are currently connected to the server with a ssh connection through eth0.So we cant change the ip address to eth0.this will disconnect the ssh connectivity .we are going to give the static ip address to the eth1 interface

Third step

we have to do to the *'/etc/sysconfig/network-scripts'*

directory

=> ***cd /etc/sysconfig/network-scripts***

In this directory There are a lot of files .From there we have to select the '***ifcfg-eth1***' [yours can be different .select the file based on your interface it will be like ifcfg-<interface>]

```
[vagrant@tanvir ~]$ cd /etc/sysconfig/network-scripts/
[vagrant@tanvir network-scripts]$ ls
ifcfg-eth0   ifdown-ippp    ifdown-sit      ifup-bnep  ifup-plusb   ifup-TeamPort
ifcfg-eth1   ifdown-ipv6    ifdown-Team     ifup-eth   ifup-post    ifup-tunnel
ifcfg-lo     ifdown-isdn    ifdown-TeamPort ifup-ippp  ifup-ppp     ifup-wireless
ifdown       ifdown-post    ifdown-tunnel   ifup-ipv6  ifup-routes  init.ipv6-global
ifdown-bnep  ifdown-ppp     ifup            ifup-isdn  ifup-sit     network-functions
ifdown-eth   ifdown-routes  ifup-aliases     ifup-plip  ifup-Team    network-functions-ipv6
[vagrant@tanvir network-scripts]$
```

Fourth step

we have to edit the file with a text editor with root privileges. We have to edit the file ifcfg-eth1

=>***vim ifcfg-eth1***

BOOTPROTO=static

ONBOOT=yes

IPADDR=192.168.0.10

PREFIX=24

GATEWAY=192.168.0.1

DNS1=8.8.8.8

```
#VAGRANT-BEGIN
# The contents below are automatically generated by Vagrant. Do not modify.
BOOTPROTO=static
ONBOOT=yes
DEVICE=eth1
NM_CONTROLLED=yes
#VAGRANT-END
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
IPADDR=192.178.0.10
PREFIX=24
GATEWAY=192.168.0.1
DNS1=8.8.8.8
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=no
NAME="System eth1"
UUID=9c92fad9-6ecb-3e6c-eb4d-8a47c6f50c04
~
~
~
```

fifth step

if we see our ip address we can see the the ip address still dont change.to make the change we need to restart the interface.

We shutdown the interface with this command

=>***sudo ifdown eth1***

Then we start the interface again

=>***sudo ifup eth1***

Sixth step

Then if we check ip address using

=>***ifconfig eth1***

```
[vagrant@tanvir ~]$ ifconfig eth1
eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.178.0.10  netmask 255.255.255.0  broadcast 192.178.0.255
        inet6 fe80::a00:27ff:fecd:5aa3  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:cd:5a:a3  txqueuelen 1000  (Ethernet)
        RX packets 62  bytes 5854 (5.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 24  bytes 2452 (2.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[vagrant@tanvir ~]$
```

we can see the ip address changed .

Seventh step

We have to test the connection via pinging a network.

=>***ping 8.8.8.8***

So the connection is up and running. Thats is the another way of giving an ip address to a cenos7/Redhat7 server a static address.

* * *

SETTING STATIC IP IN UBUNTU

EASY WAY

Setting the ip address in a debian machine with a easy method
You have to follow these steps

first step

you need to select a static ip address , subnet mask and the gateway that you give your machine . according to your network specifications.

In his example we used a virtual debian box . And we give the following ip address subnet mask , gateway and Dns

IP ADDRESS : 192.168.0.10

SUBNET MASK: 255.255.255.0

GATEWAY:192.168.0.1

DNS: 8.8.8.8

second step

you need to find the network interface that you give the static ip address A Server can have multiple network interface.
In our virtual machine there are two network interface. We can see the interface from this command

=>***ifconfig***
or
=> ***ip address show***

result:

```
[vagrant@tanvir ~]$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::5054:ff:fe8a:fee6  prefixlen 64  scopeid 0x20<link>
        ether 52:54:00:8a:fe:e6  txqueuelen 1000  (Ethernet)
        RX packets 1110  bytes 135804 (132.6 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 940  bytes 149277 (145.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.5  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::a00:27ff:fecd:5aa3  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:cd:5a:a3  txqueuelen 1000  (Ethernet)
        RX packets 13  bytes 1362 (1.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 16  bytes 1826 (1.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 32  bytes 2592 (2.5 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 32  bytes 2592 (2.5 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

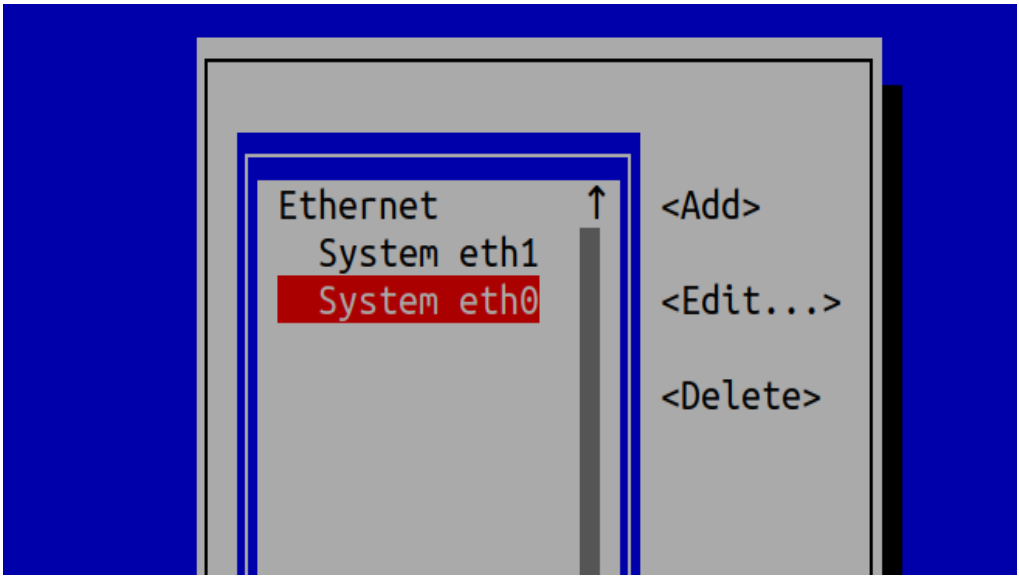
we are currently connected to the server with a ssh connection through eth0.So we cant change the ip address to eth0.this will disconnect the ssh connectivity .we are going to give the static ip address to the eth1 interface

Third step

use the ***nmtui*** command and you have to be root to give this command

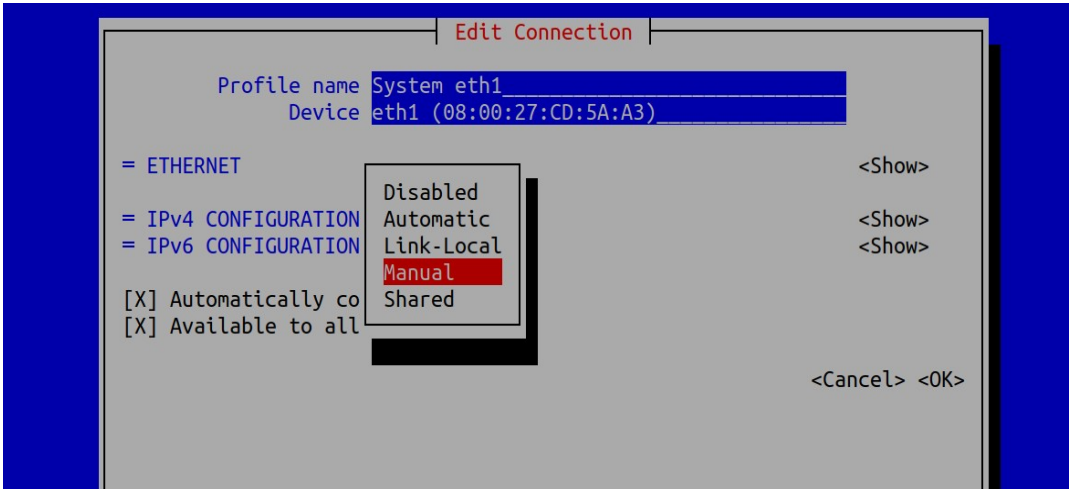
=>***sudo nmtui***

After giving this command this screen appear. From there Select The “Edit a connection”



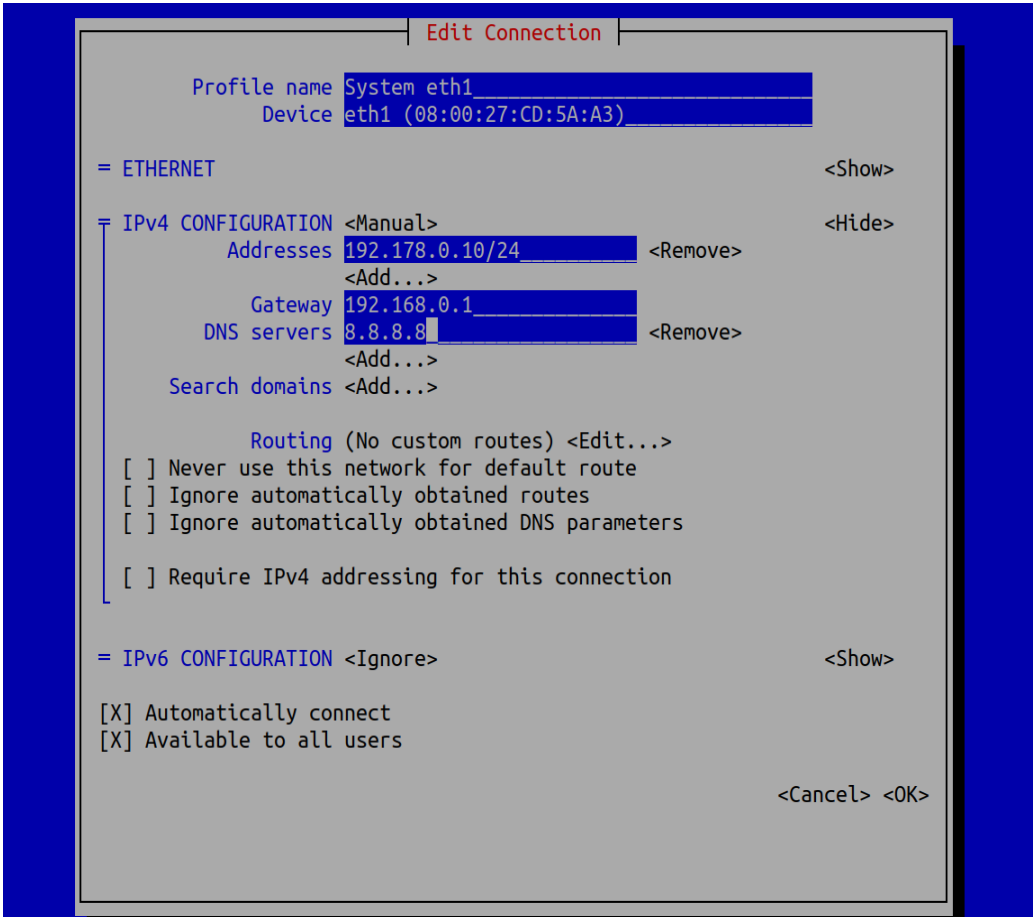
Fourth step

it will show you all the interface .choose your interface in this case we will choose eth1.



Fifth step

we choose the ipv4 and from the option we choose ‘manual’ and Edit the menu



Sixth step

we give the ip address.we have to give the subnet mask with CIDR notation.

Gateway and the The DNS address and click ok. Then quit the program.

Seventh step

if we see our ip address we can see the the ip address still dont change. to make the change we need to restart the interface.

We shutdown the interface with this command

=>***sudo ifdown eth1***
or
=>***nmcli connection down eth1***

Then we start the interface again

=>***sudo ifup eth1***
or
=>***nmcli connection up eth1***

Eighth step

Then if we check ip address using
=>***ifconfig eth1***

```
[vagrant@tanvir ~]$ ifconfig eth1
eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.178.0.10  netmask 255.255.255.0  broadcast 192.178.0.255
        inet6 fe80::a00:27ff:fecd:5aa3  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:cd:5a:a3  txqueuelen 1000  (Ethernet)
        RX packets 62  bytes 5854 (5.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 24  bytes 2452 (2.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[vagrant@tanvir ~]$
```

we can see the ip address changed .

Ninth step

We have to test the connection via pinging a network.

=>***ping 8.8.8.8***

So the connection is up and running. Thats is the easy way of giving a static ip address to a Ubuntu/Debian server

NETPLAN

New version of ubuntu linux has a new tools for setting ip address .This is called netplan. now its a little bit hard because you have to maintain indentation and certain rules to give it.and the main challenge is you have to do it by editing a file

First step

The network configuration stored in ***'/etc/netplan'*** directory

Second step

There are different '***yaml***' configuration file for different interface .in my VM there are two different interfaces. so there are two different interfaces. you have to configure the configuration file based on what interface you want to configure

Third step [very very important !!!]

you mast take backup before you edit the file

=>***cp 99-vagrant.yaml 99-vagrant.yaml.bak***

[this is very very important cause if you make mistake in the indentation you have to]

Fourth step

Edit the file

=>***sudo vim 99-vagrant.yaml***

the file format will be like this

```
# the datasource.  Changes to it will not persist across
an instance.
# To disable cloud-init's network configuration
capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg
with the following:
# network: {config: disabled}
network:
    ethernets:
        eth1:
            addresses: []
            dhcp4: true
    version: 2
```

the ip address we assign will be

IP ADDRESS : 192.168.0.102

SUBNETMASK : 255.255.255.0

GATEWAY : 192.168.0.1

DNS : 8.8.8.8,8.8.8.4

Fifth step

Fill like this

```
# the datasource.  Changes to it will not persist across
an instance.
# To disable cloud-init's network configuration
capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg
with the following:
# network: {config: disabled}
network:
    ethernets:
        eth1:
            addresses: [192.168.0.102/24]
            gateway4: 192.168.0.1
            nameservers:
                        addresses: [8.8.8.8,8.8.4.4]
            dhcp4: no
    version: 2
```

Sixth step

find error

*=>**sudo netplan -debug apply***

Seventh step

apply the changes

=>***sudo netplan apply***

that is the new way of giving static ip address

***

Package management in Linux

As a server administrator you will need to install different software on your server on different occasion .Most of the Linux operating system(Ubuntu Server/Centos server/Open SUSE server) has two different ways of installing software. First are the software packages that contain the programs that are ready to install and that integrate with the server easily. The server keeps the list of installed packages in the database that makes maintaining very easy. The second option to install software in via tarball. Which basically just an archive of the software. Archive can be anything (can be any record of the data) but it can b also used to deliver software. The first method is proffered most of the time Because server can keep track of the software that are installed via packages .Software installed via tarball are not tracked. There is a second difference between packages and tarballs that some software need other packages for working properly (this is called dependency).both tarball and packages have program installed that check if the

dependencies are met but only the software packages interact with the package manager. And in that way it can install the missing dependencies which other installation system cant do. So now a days software packages are preferred. Software packages mostly made in two different formats .On Red Hat and openSUSE and similar distribution rpm packages is used .And debian based operating system like ubuntu server deb package is used.But this packages can be converted. And the other advantage is software can be install by compiling the source code too.

High level and Low level Pckage management Tools

in order to interact with the software packages there are two types of available tools. low level package management also known as local package management system. and the high level tools are known as online package management tools.

Distribution	Low-Level Tools	High Level Tools
Debian based distribution	dpkg	apt/aptitude
Centos/Red Hat	rpm	yum
Open SUSE	rpm	zypper

[do not use red hat rpm file in openSUSE system]

If you already download or create your own .deb package you can manage it with ***dpkg*** command.

UBUNTU PACKAGE MANAGEMENT

Installing package with dpkg

For installing packages with dpkg . command is

=>***dpkg -i <package_name>***

List of current package:

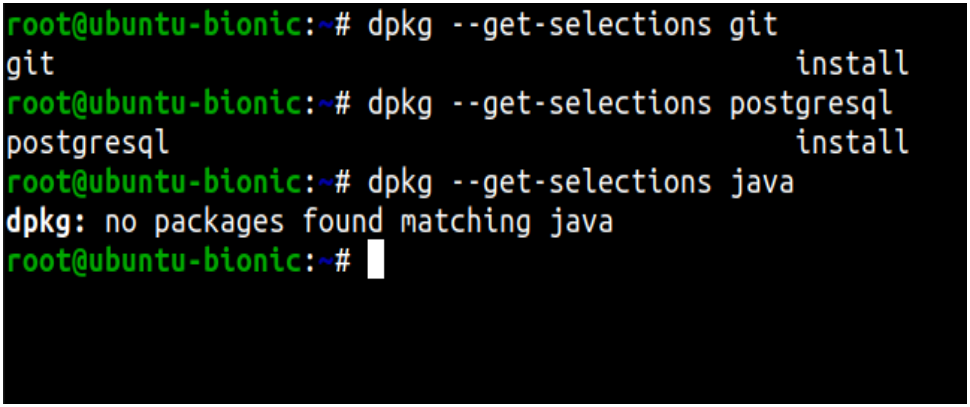
To list all the current packages that are currently installed in Ubuntu server the command is
=>dpkg -L

it will show the name,version,architecture and a small description

Check packages installation status

if you need to know any packages installed or not then following command can show if the package installed or not

=>***dpkg –get-selections <package_name>***



Check Details information about packages:

To check details about a installed packages use this command

=>***sudo dpkg -s <package_name>***

Disadvantage of dpkg:

suppose we want to install a downloaded packages ***webmin.deb.*** We will show some dependency problem like this and it install the program without the dependency and the program wont

run you have to install dependency manually the other dependencies that's a big complexity .If you remove the program it still create the problem if you try to install other program.

```
root@ubuntu-bionic:~# ls
webmin_1.920_all.deb
root@ubuntu-bionic:~# dpkg -i webmin_1.920_all.deb
Selecting previously unselected package webmin.
(Reading database ... 65008 files and directories currently installed.)
Preparing to unpack webmin_1.920_all.deb ...
Unpacking webmin (1.920) ...
dpkg: dependency problems prevent configuration of webmin:
 webmin depends on libnet-ssleay-perl; however:
  Package libnet-ssleay-perl is not installed.
 webmin depends on libauthen-pam-perl; however:
  Package libauthen-pam-perl is not installed.
 webmin depends on libio-pty-perl; however:
  Package libio-pty-perl is not installed.
 webmin depends on apt-show-versions; however:
  Package apt-show-versions is not installed.
 webmin depends on python; however:
  Package python is not installed.

dpkg: error processing package webmin (--install):
 dependency problems - leaving unconfigured
Processing triggers for ureadahead (0.100.0-21) ...
Processing triggers for systemd (237-3ubuntu10.23) ...
Errors were encountered while processing:
 webmin
root@ubuntu-bionic:~#
```

[To fix this problem we can use the online package management system

=>sudo apt-get install -f

it will search the dependencies and install them

Remove packages:

to remove packages from the system this command is used

=>***dpkg -r <package_name>***

Completely remove package and configuration file:

to completely remove package and the related configuration file this command is used

=>***dpkg -P <package_name>***

If you find a file and want to know which package it belongs to use this command

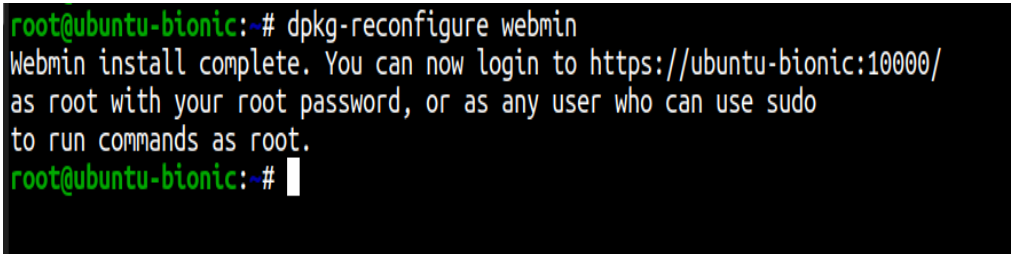
=>***dpkg -S <file_path>***

```
root@ubuntu-bionic:~# dpkg -S /bin/cp
coreutils: /bin/cp
root@ubuntu-bionic:~# dpkg -S /bin/cat
coreutils: /bin/cat
root@ubuntu-bionic:~# dpkg -S /bin/ping
iputils-ping: /bin/ping
root@ubuntu-bionic:~#
```

Reconfigure packages:

if you face any problem in your package configuration. You can reconfigure the package with this command
=>***dpkg-reconfigure <package_name>***

But to do this you need to know the exact name of the package. It will automatically rewind the installation process and give you chance to reconfigure.



Installing packages with apt

The ***apt*** utility is a powerful and free package management command line program, that is used to work with Ubuntu's APT (Advanced Packaging Tool) library to perform installation of new software packages, removing existing software packages, upgrading of existing software packages and even used to upgrading the entire operating system

On ubuntu server or any debian based OS there is a list repository url which is populated during the installation in ***'/etc/apt/sources.list'*** but you can add repository.

Update repository:

Before installing any package you need to update the software repository.
Command
=>***sudo apt update***

[you need to be root to perform the action]

Upgrade existing Software:

To upgrade every package in the latest version use this command

=>***sudo apt upgrade***

Update OS distribution

to upgrade the distribution for example upgrading ubuntu 16.0 to ubuntu latest version this command is used

=>***sudo apt dist-upgrade***

Install Packages

for installing packages this command is used

=>***sudo apt install <package_name>***

for example
to install vim editor we use this command
=>***sudo apt install vim***

Remove Packages

for removing packages this command is used
=>***sudo apt remove <package_name>***

for example
to remove vim editor we use this command
=>***sudo apt remove vim***

[this command will remove the packages but not the dependencies .To remove this command is used
=>***sudo apt autoremove***

apt-cache command

The apt-cache command line tool is used for searching apt software package cache. In simple words, this tool is used to search software packages, collects information of packages and

also used to search for what available packages are ready for installation on Ubuntu based systems.

Apt-cache search command

=>sudo apt-cache search <package_name>

This command show all the program will show all the program that depends on the packages. suppose you install gmail packages this command

=>sudo apt-cache search gmail
will show all the packages that are depends on this packages like 'thunderbird'

Package Details

You can also see the details of any packages with apt just like the ***dpkg -s.***
command

=>sudo apt-cache show vim

Find Unmet Dependencies:

This command will find all the unmet dependencies of the system

=>sudo apt-cache unmet

Find Specific Dependency of Packages:

=>sudo apt-cache depends <package_name>

This command will give all the dependencies of the Packages.

Find Reverse Dependencies:

=>sudo apt-cache rdepends <package_name>

This command will find the reverse dependencies of the program .That means it will show all the packages that depends on that packages.

For example:
=>sudo apt-cache rdepends git

this command will show all the other program that depends on the git program.

Aptitude package management tool:

There is a new package management tools called aptitude. to use that first you have to install it with this command

=>***sudo apt install aptitude***

Install package via aptitude:

installing command with aptitude is

=>***sudo aptitude install <package_name>***
example:
=>***sudo aptitude install emacs***

Search package via aptitude

For searching any packages this command is used

=>***sudo apttitude search <package_name>***

The main advantage of the aptitude is when you run the aptitude program without any flag

=>***aptitude***

this will open a menu based installer inside the terminal. That means you will get almost a gui based installer inside a terminal.

Graphical Package management System:

If you want to use a graphical Package management system you can use synaptic package management software. its very easy to install,remove,and
upgrade packages with synaptic package management.

Apt Repository:

when we install or search a package with apt command it will search some online repository for that packages. The list of that url is stored in a file
'***/etc/apt/sources.list***' and the file contained in
'***/etc/apt/sources.list.d***'

if we see the '***sources.list***' file with this command

=>***cat /etc/apt/sources.list***
we will see something like this

the information available from the configured sources is acquired by 'apt update' or equivalent command from another apt fronted.
Users can manually add repository url in that file. after adding repository you have to issue 'apt update' command to make it available for using.

Or you can just create a file in ***/etc/apt/sources.list.d*** directory.The file must be end with .list extension. The apt package manager also read repository configuration from there

for example:
first open a file with vim editor inside the ***sources.list.d*** repo

=>vim /etc/apt/sources.list.d/games.list

add the repository path in that file

deb http://archive.getdeb.net/ubuntu wily-getdeb games

Or user can add repository by interactive command.

Use the add-apt-repository (or symlink apt-add-repository) command to add repository. You just need to provide reference address as the following command.

=>add-apt-repository 'deb http://archive.getdeb.net/ubuntu wily-getdeb games'

to remove any repository from by using this following command

=>add-apt-repository -r 'deb http://archive.getdeb.net/ubuntu wily-getdeb games'

*[*every time you make a change to repository you must apply 'apt update' command to make the change on effect *]*

CENTOS PACKAGE MANAGEMENT

Rpm (Red Hat Package Manager)and Yum(Yellowdog Updater Modified) package management tools are basically Centos/Redhat,fedora like Operating system.

Like dpkg in debian based OS. Rpm is the local package management tool(low level package management tool).and Yum is the online package management tool(high level package management tool).Yum is like apt in ubuntu OS.

[just like the dpkg the rpm command may face dependency problem while installing software .and yum search the dependency automatically and install them.]

rpm package management

Install package:

For install package with rpm this command is used

=>rpm -i <package_name>

remove package

For remove this package with rpm this command is used
=>***rpm -e <package_name>***

[if one package depends on the other package you cant remove it with rpm command unless you remove the other packages that depends on it. For example if you want to remove the 'openssh' package because the 'open-ssh client' packages depends on it .First you have to remove this. But if you use the yum command to remove the any packages this will happen automatically.]

Force Install package:

if you want to install a packages with or without the dependency (force install) you can do it with this command

=>***rpm -i –nodeps <package_man>***

[its not recommended because it leaves you a broken dependency problem]

Verbosity:

if we want to see whats happening when installing or removing we can use the verbosity flag.

Install package with verbosity flag

=>***rpm -i -v <package_name>***

Remove package with verbosity flag

=>***rpm -e -v <package_name>***

Check Package install Status:

if you want to check is a package is installed or not .you can do with this command

=>***rpm -Vv <package_name>***
for example

=>***rpm -Vv nano-2.3.1-10.el7.x86_64.rpm***

[if you want to find out that your package is intact you can find it by checking the output flag. Because if you change any configuration and run the command again it will show you different result. That proves that file is changed]

Check Package Checksum:

To check the file checksum this command is used

=>rpm -vK <package_name>

Find Package Description:

To find the description of any installed package this command is used

=>rpm -qi <installed_package>
for example

=>rpm -qi nano

Query All Packages:

To query all the packages this command is used

=>rpm -q -a

you can find any installed packages with this command
=>rpm -q -a | grep <packages_name>
example
=>rpm -q -a | grep dhcp

Yum package management:

yum(Yellowdog Updater Modified) is more advance package management tools you can do everything with yum that can be done with rpm.yum uses a lot of third party repository to install packages automatically by resolving their dependency issue

Find Package information:

To find detail information about any packages this command is used .it will search the repository and give detail information about the packages.

=>***yum info <package_name>***

Search package:

To search the packages in the repository this command is used

=>***yum search <package_name>***

Install package:

To install packages this command is used.it will install the packages with the dependency

=>yum install <package_name>

This command will ask for confirmation. to install automatically. Just add a -y option .

=>yum install -y <package_name>

Remove package:

To remove package with all its dependencies this command is used.

=>yum remove <package_name>

This command will ask for confirmation. to install automatically. Just add a -y option .

=>yum remove -y <package_name>
or
=>yum erase -y <package_name>

Update package:

If you have any outdated version of any packages and you need to update it. you can use the update command to update to its latest stable version. If it needs any additional dependency it will automatically resolve them

=>***yum update <package_name>***

List packages:

To list all the available packages in the Yum repository this command is used

=>***yum list | more***

To list all the installed packages this command is used

=>***yum list installed***

you can use the list function as a searching purpose .for

searching packages this command is used

=>***yum list <package_name>***

Yum provides function:

if you find any program or any files and want to find out which packages it belongs to. You can find it with this command

=>***yum provides <file_name/program_names>***

Check update packages:

If you want to check weather any update available for your installed packages you can check using this command
=>***yum check-update***

Update system:

If you want to update all your packages and system and install all the latest patches and security updates in your system this

command is used

=>***yum update***

[one of the main advantage of the yum over the apt command is before installing any packages yum will automatically update the repository]

List all the group packages:

Number of packages are bundled up to make a particular group. Instead of installing individual packages you can install the whole particular group. To list all the group this command is used

=>***yum grouplist***

Install group packages:

To install a particular package group we use the groupinstall.

=>***yum groupinstall '<group package name>'***

for example

=>***yum groupinstall 'Basic Web Server'***

Update group packages:

To update a particular package group we use the groupupdate.

=>***yum grouupdate '<group package name>'***

for example

=>***yum grouupdate 'Basic Web Server'***

Remove group packages:

To remove a particular package group we use the groupremove.

=>***yum groupremove '<group package name>'***

for example

=>***yum groupremove 'Basic Web Server'***

List Enabled yum repository:

To list all the enabled yum repository this command is used

=>***yum repolist***

List All yum repository

To list all the enabled and disabled yum repository this command is used

=>***yum repolist all***

List packages from a particular repository:

To install a packages from a particular repository this command is used

=>***yum –enablerepo=epel install java***

[This command wont enable the repository permanently .its only for the current command]

Permanently Enable/Disable a particular repository:

To enable a repository permanently this command is used

=>yum-config-manager –enable <repo_name>

[This command will enable the repository permanently]

To disable a repository permanently this command is used

=>yum-config-manager –disable <repo_name>

[This command will disabled the repository permanently]

Clean yum Cache

To clean all the cached files from enabled repository this following command is used.

=>yum clean all

View History

To view all the past transactions of the yum command this following command is used

=>***yum history***

Yumdownlaoder

there is another tools called 'yumdownloader' in the redhat/centos based system. The job of this tools is to download the rpm file. Means it just download the rpm file but doesn't install it. The following command is used to download rpm file

=>***yumdownloader <pakage_name>***

for example

=>***yumdownloader git***

it will install the ***git.rpm*** file but it wont download the dependency. To download any package with the dependencies this command is used

=>***yumdownloader --resolve <pakage_name>***

for example

=>***yumdownloader --resolve git***

Yum Repository

just like the ***'sources.list'*** file in the ubuntu package management there is also a place where the repository files stored.its in the '***/etc/yum.repos.d***'
we can list all the files with the 'ls -s' command.you will see something like this

there can be more than one .repo file if you look inside the file with this command

=>***cat repofile.repo***

example
=>***cat CentOS-Base.repo***

if you look inside the file it will like the ***'sources.list'.*** Just a little bit different

There are different different mirror list for ***'base','updates','extras'*** and additional '***packages***' and every section has a

1) name for the mirror list
2) baseurl for that mirror
3) gpgcheck option
4) enable option
5) gpgkey

if you want you can disable the gpgcheck cause the the repository may not be encrypted.

there is a configuration file in ***/etc/yum.conf*** .By changing the configuration you can customize the operation of the yum tools.

=>***keepcache=0*** will not keep the cache file
=>***logfile=***'/var/log/yum/log' will store the log file in that file
=>***obsolute=1*** delete the obsolete packages
=>***gpgcheck=1*** will check gpg every time it install packages
=>***plugins=1*** will allow yum to install plugins

[yum uses different plugins. one of them is fastest mirror.it founds the fastest mirror so the user find the packages as fast as possible]

Comparison Between two Package management system

Operation	Debian package management	Centos package Management
Show package information	sudo apt show <pkg> sudo dpkg -s <pkg>	sudo yum info <pkg> sudo rpm -qi <pkg>
List all the packages	Sudo apt list sudo dpkg -L	Sudo yum list sudo rpm -q -a
Download Packages	sudo apt download <pkg>	Yumdownloader <pkg> Yumdownloader –resolve <pkg>
Search packages	sudo apt search <pkg>	Sudo Yum search <pkg>

	sudo aptitude search <pkg>	
Install packages	*Sudo apt install <pkg>* *sudo aptitude install <pkg>* *sudo dpkg -i <pkg>*	*Sudo yum install <pkg>* *sudo rpm -i <pkg>*
Remove Packages	*Sudo apt remove <pkg>* *sudo dpkg -r <pkg>* *sudo aptitude remove <pkg>*	*Sudo yum remove <pkg>* *sudo yum erase <pkg>* *sudo rpm -e <pkg>*
Check integrity	*Sudo dpkg -V <pkg>*	*Sudo rpm -V <pkg>*
Update packages/system	*Sudo apt update*	*Sudo yum update*
Upgrade System	*Sudo apt upgrade*	*Sudo yum upgrade*

YUM SERVER

WITHOUT CONFIGURING FTP SERVER

every centos or red hat installation DVD is shipped with a lot of necessary packages for all kinds of basic server setup. We can use those packages to make a local yum server so we can install the packages with their dependencies. We can achive this goal by creating a ftp server and configure it . Or we can create yum server without creating any ftp server.if you create a FTP server multiple host on the network can access your yum server and pul the necessary packages but if you configure without the FTP server only you can use your local yum sevrer

First Step

we mount the ***cdrom*** in the ***/media*** folder

=> ***mount /dev/cdrom /media***

Second Step

create a directory in the / directory name ***"/myrepo"***

Third Step

copy the whole file in the cdrom in the ***"/myrepo"***
=> ***cp -r /media/* /myrepo***

Fourth Step

go to ***/etc/yum.repos.d***

[root@localhost ~]# cd /etc/yum.repos.d/

[if you want to keep only local you can delete rest of the file in the folder
]
create a file name ***"myrepo.repo"***

Fifth Step

=>vim myrepo.repo

[myrepo]
baseurl=file:///myrepo
enabled=1
gpgcheck=0

Sixth Step

update with this command

[root@localhost ~]# yum update --disablerepo="*" --enablerepo='myrepo'

Seventh Step

install packages

yum install --disablerepo="*" --enablerepo='myrepo' <package name>

WITH A FTP SERVER

First we have to install a file server. To install it we have to install some dependencies first because rpm do not install dependencies. ***Vsftpd*** is a file server packages . These packages are in the ***sr0*** drive we first mount it

First Step

=>mount /dev/sr0 /mnt
=>cd /mnt/Packages

Second Step

for working properly we have to install these packages which are the dependencies of the vsftpd

1) python-deltarpm
2) createrepo

Third Step

installing command:
=> ***rpm ivh –force –nodeps python-deltarpm****
for creating repo we have to install another packages
=> ***rpm –ivh –force –nodeps createrepo****
after that we install vsftpd and set the file server

Fourth Step

=> ***rpm -ivh –force –nodeps vsftpd****

after installing the ***vsftpd*** automatically the ***/var/ftp/pub*** directory will be created. in the pub directory all the files in the file server stay publicly. Inside the pub directory we create another directory called ***rhel7***(you can name it anything) .Create the folder (if not created)

=>***mkdir -p /var/ftp/pub/rhel7***

now copy all the thing in the ***sr0*** in this folder

Fifth Step

=>cp -rv /mnt/* /var/ftp/pub/rhel7/

Sixth Step

now we will create configuration file .before that we have to remove all the configuration file
from the ***/etc/yum.repos.d*** folder
=>cd /etc/yum.repos.d=>rm -rf *
create a file with vim editor

Seventh Step

=>vim rhel7.repo
in the file add the line for setting the path:
in the editor

[base]
name="red har local packages"
baseurl="file:///var/ftp/pub/rhel7/Packages"

enabled=1
gpgcheck=0

Eights Step

now we create the repo with the packages
command is
=>***createrepo -v /var/ftp/pub/rhel7/Packages***

Ninth Step

=> ***yum clean all***
=> ***yum list all***
=> ***yum repolist***

APT SERVER

WITH A APACHE WEB SERVER

Just like the centos, the debian /Ubuntu server also gives opportunity to make a local server for package management. And in the Debian server or debian based other server we use the local APT repository.It is necessary because setting up a local repository saves a lot of bandwidth and make possible for local clients to install necessary pacakges .so the client dont have to pull the packages from the public server

First Step

log in to the server with root user and update the system

=> ***apt update && apt upgrade***

Second Step

install the packages to make a local repository

=> ***apt install build-essential***

Third Step

we need a web server to serve all the packages to the clients.we will use the apache web server

=> ***apt install apache2***

Fourth Step

we go to the web browser and see if the web server is up and running

Fifth Step

Create a Directory inside the web server public directory to save packages depending on the system architecture .For example if you use a 32 bit system create a “i386” directory or for 64 bit system use “amd64” directory. You can keep both directory and serve packages to different architecture system at the same time.In this example we only make repo for 64 bit

system only.

=> ***mkdir /var/www/html/packages/amd64***

Sixth Step

copy all the DEB packages from the Debian installation media I) debian server comes with three DVD all of them have different packages .you have to copy from all the dvd one by one to the destination

1) Mount the first DVD and search and copy all the ".deb" files to the ***/var/www/html/packages/amd64***

=> ***mount /dev/cdrom /media/cdrom***

2) Search and copy all the .deb file to the destination with this command

=> ***find /media/cdrom/pool -name "*.deb" -exec cp {} /var/www/html/packages/amd64 \;***

*[*it will find and search all the deb packages to the destination*]*

3) unmount the dvd and insert the next DVD and repeat the last two process and copy all the packages to the destination.

Seventh Step

To varify this go to the web browser and go to the '*http://localhost/packages/amd64*' url .you will find all the packages there.

Eight Step

Navigate to the "/var/www/html/packages/amd64" directory.

=>***/var/www/html/packages/amd64***

Ninth Step

Now we have to scan The packages to make a catalog file for using by the APT command.

=>***dpkg-scanpackages . /dev/null | gzip -9c >Packages.gz***

[Packages.gz – the 'P' have to be capital letter]

[depending on the number of packages this will take time]

sample output:

dpkg-scanpackages: info: Wrote 1151 entries to output Packages

file.

[we have created the catalog file.but we have to do that process everytime we add new packages]

Tenth Step

Edit ***/etc/apt/sources.list***

1) "/etc/apt/sources.list" contain all the repository localtion.we have to delete[or comment out all the online repo and add this line in the file].and we have to add a flag to force the server to install packages fro untrusted/insecure repo.

=>vim /etc/apt/sources.list

---

deb [allow-insecure=yes] file:/var/www/html/packages/amd64 /

[note there have to be a space after the amd64 : amd64/<space>/]

Eleventh Step

Update Repository

=> ***apt update***

Twelveth Step

11) Install packages
=>***apt install <package_name>***

[example]
=>***apt install vsftpd***

WITHOUT A APACHE SERVER

First Step

log in to the server with root user and update the system

=> ***apt update && apt upgrade***

Second Step

install the packages to make a local repository

=> ***apt install build-essential***

Third Step

we go to the web browser and see if the web server is up and running

Fourth Step

Create a Directory

=> ***mkdir -p /packages/amd64***

Fifth Step

copy all the DEB packages from the Debian installation media I) debian server comes with three DVD all of them have different packages .you have to copy from all the dvd one by one to the destination

1) Mount the first DVD and search and copy all the ".deb" files to the ***/packages/amd64***

=> ***mount /dev/cdrom /media/cdrom***

2) Search and copy all the .deb file to the destination with this command

=> ***find /media/cdrom/pool -name "*.deb" -exec cp {}***

/packages/amd64 \;

*[*it will find and search all the deb packages to the destination*]*

3) unmount the dvd and insert the next DVD and repeat the last two process and copy all the packages to the destination.

Sixth Step

Navigate to the "***/packages/amd64***" directory.

=>cd /packages/amd64

Seventh Step

Now we have to scan The packages to make a catalog file for using by the APT command.

=>dpkg-scanpackages . /dev/null | gzip -9c >Packages.gz

[Packages.gz – the 'P' have to be capital letter]

[depending on the number of packages this will take time]

sample output:

dpkg-scanpackages: info: Wrote 1151 entries to output Packages

file.

[we have created the catalog file.but we have to do that process everytime we add new packages]

Eighth Step

Edit ***/etc/apt/sources.list***

1) "/etc/apt/sources.list" contain all the repository localtion.we have to delete[or comment out all the online repo and add this line in the file].and we have to add a flag to force the server to install packages fro untrusted/insecure repo.

=>vim /etc/apt/sources.list

---

deb [allow-insecure=yes] file:/packages/amd64 /

[note there have to be a space after the amd64 : amd64/<space>/]

Ninth Step

Update Repository

=> ***apt update***

Tenth Step

11) Install packages

=>***apt install <package_name>***

[example]

=>***apt install vsftpd***

***

KERNEL MANAGEMENT

SSH: THE SECURE SHELL

WHAT IS SSH?

SSH is a cryptographic network protocol for secure network services

USES

→ ***It is used for the remote login***
→ ***Secure File Transfer (SFTP/SCP)***
→ ***Port Forwarding***
→ ***SOCKS protocols for web browsing through encrypted proxy***
→ ***Secure remote file mounting via SSHFS***

Login With SSH Using Password

requirements:

→ we have two server

1) server1, ip:192.168.0.10/24

2) server2, ip :192.168.0.11/24

First Step

we need to install the ***openssh-server*** in server2 [in centos server its actually pre-installed]

=>yum update -y

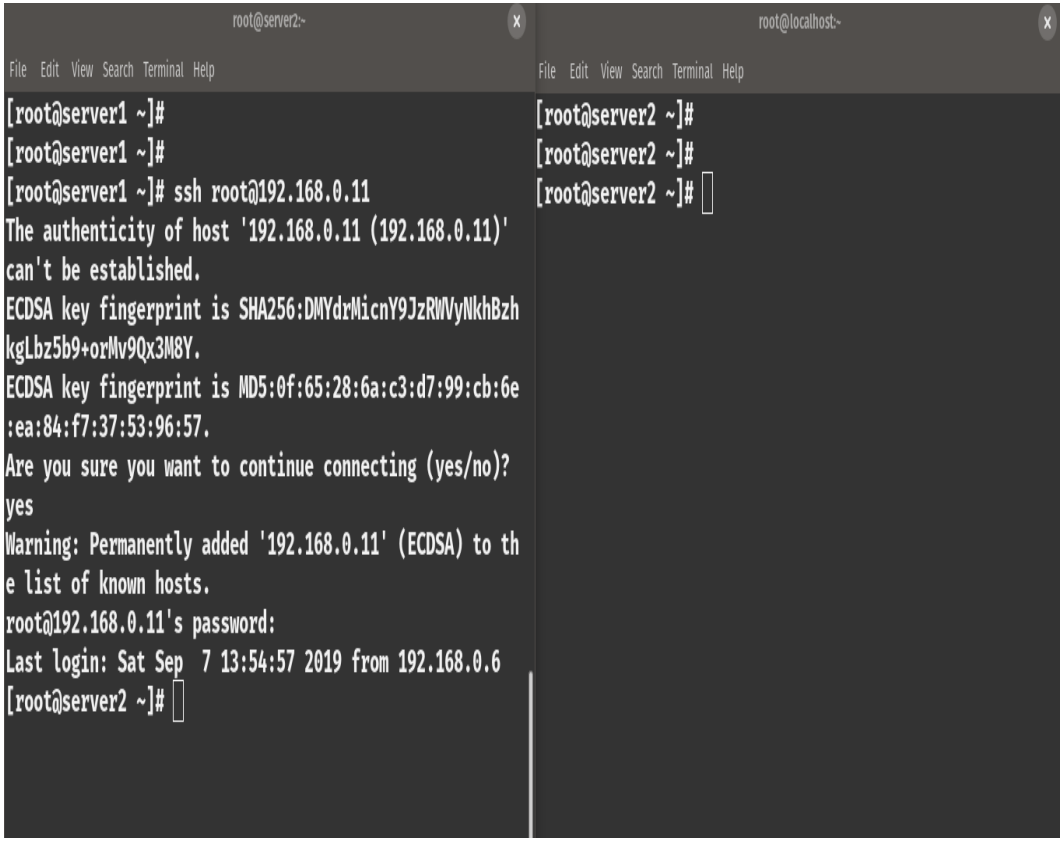
=>yum install sshd -y

Second Step

2) from server1 use the command and give the password

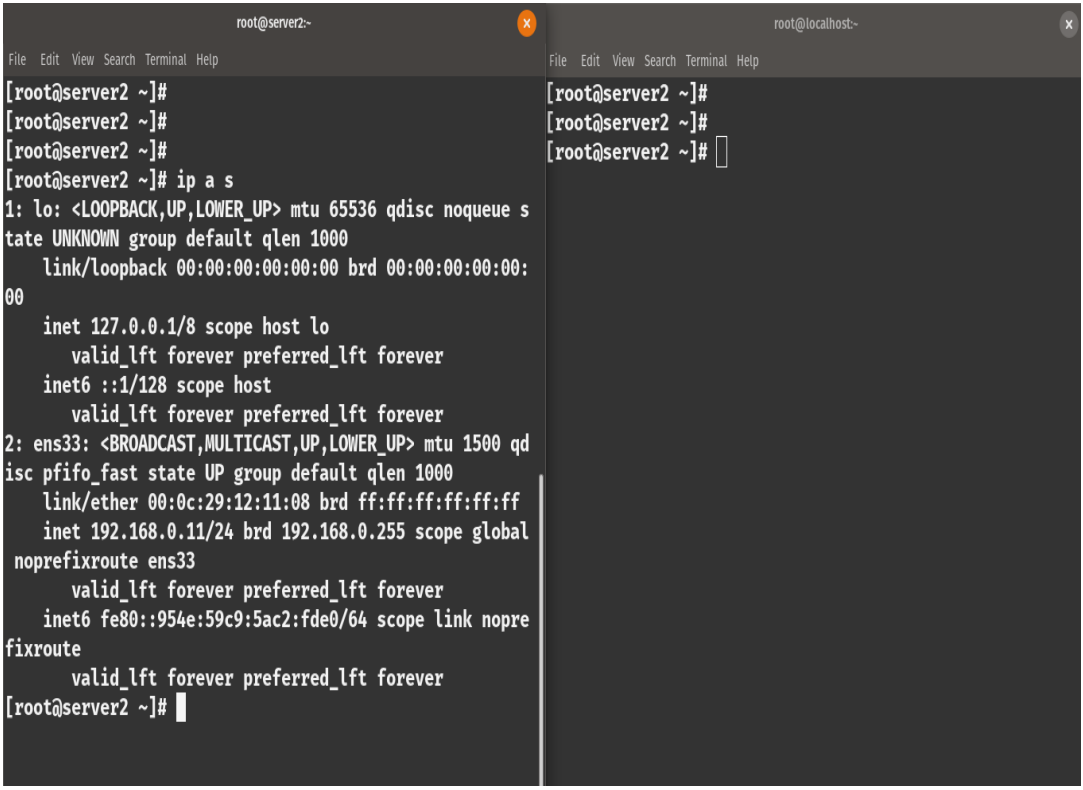
=>ssh root@192.168.0.11

password: <server2 password>



Third Step

now you are logged in in server 2.Check with the ***ifconfig*** and ***hostnamectl*** command


root@server2:~
File Edit View Search Terminal Help
[root@server2 ~]#
[root@server2 ~]#
[root@server2 ~]#
[root@server2 ~]# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host
valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
link/ether 00:0c:29:12:11:08 brd ff:ff:ff:ff:ff:ff
inet 192.168.0.11/24 brd 192.168.0.255 scope global noprefixroute ens33
valid_lft forever preferred_lft forever
inet6 fe80::954e:59c9:5ac2:fde0/64 scope link noprefixroute
valid_lft forever preferred_lft forever
[root@server2 ~]#
root@localhost:~
File Edit View Search Terminal Help
[root@server2 ~]#
[root@server2 ~]#
[root@server2 ~]#

Login with SSH Without Using Password (More Secure Way)

using password to login with ssh is one way but it is not very secure the other way is to use a ***private and public key pair***. we use a public private key pair for login rather than a password.

First Step

see if there is an existing key
=> ***ls -l ~/.ssh***

Second Step

Create the key pair from server1
[syntax:ssh-keygen -t <algorithm> -b <size>]
=>ssh-keygen -t rsa -b 4096

```
[root@server1 ~]#
[root@server1 ~]#
[root@server1 ~]# ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:RQGTsHDt7SkKJFZX9ZbQ0lrjEkE3qi+IfpZwvErq2ng root@server1
The key's randomart image is:
+---[RSA 4096]----+
|     . o++=B=o   |
|     .o..ooo==o  |
|    . ... .o*+.  |
|   o .   .o+..   |
|  . o .  S. o    |
|     o.o...o     |
|     o+.+...     |
|  oE+  *  .      |
| o++ o+          |
+----[SHA256]-----+
[root@server1 ~]#
```

[it will ask you for a passphrase for now we skip it we will discuss it later]

Third Step

we need to send the public key to ther server2.we can do it manually or we can do it using this command

*=>**ssh-copy-id server2@192.168.0.11***

```
[root@server1 ~]#
[root@server1 ~]#
[root@server1 ~]# ssh-copy-id root@192.168.0.11
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/ro
ot/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(
s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if
you are prompted now it is to install the new keys
root@192.168.0.11's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'root@192.168.0.11'
"
and check to make sure that only the key(s) you wanted were added.

[root@server1 ~]# ssh root@192.168.0.11
Last login: Sat Sep  7 15:29:15 2019 from 192.168.0.10
[root@server2 ~]#
```

Fourth Step

login with

=>***ssh*** *root@192.168.0.11*

and this time no password will be asked.

What is a Passphrase?

sometime the ssh connectivity is used by you sometimes not. for example you can make a cron job to connect automatically to a server for data backup. when you are going to use the ssh only its a good idea to use a passphrase .but for automation you should not use it cause there will be no one to type the passphrase .when you use a script to automatically connect to a server don't use any passphrase.

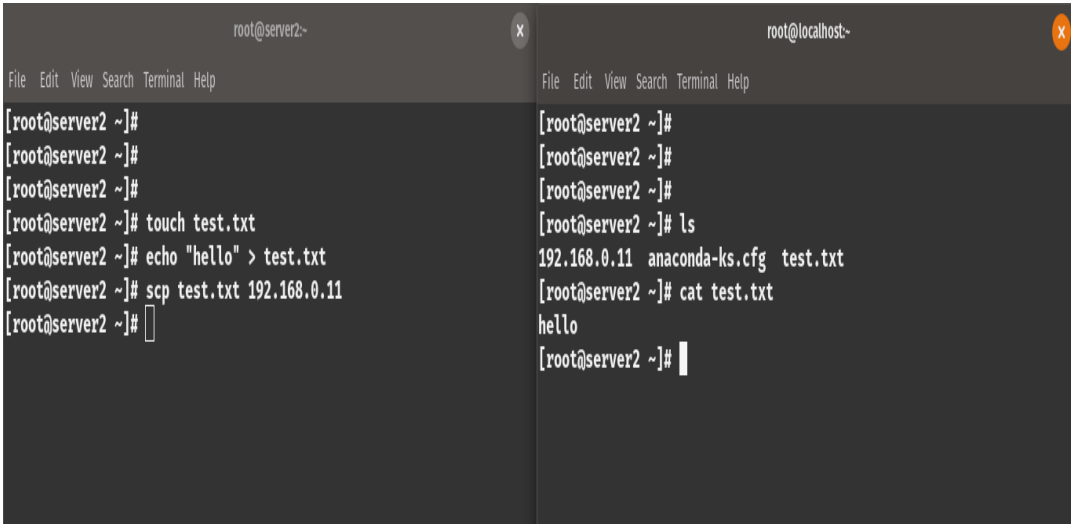
Copy File With SCP(Secure copy and paste)

syntax:

scp <local_file> <destination>

we are going to send a file name '***test.txt***' from server1 to server2

=>scp test.txt 192.168.0.11/test.txt



Copy File With SFTP(Secure File Transfer Protocol)

its a interactive process for sending file over SSH. its a sub system for ssh

=>sftp 192.168.0.10
sftp> cd /etc [go to etc directory]
sftp> get redhat-release [download the file]

Port Forwarding

Port forwarding allows us to access from one system to another system and use their network services .for exmple you are running a web server in the server2 in port 80.you can access it with a browser or see the html using this command in server2

=>***curl localhost***

```
[root@server2 ~]# curl localhost
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd"><html><head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
               <title>Apache HTTP Server Test Page powered by CentOS</title>
               <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

    <!-- Bootstrap -->
    <link href="/noindex/css/bootstrap.min.css" rel="stylesheet">
    <link rel="stylesheet" href="noindex/css/open-sans.css" type="text/css" />

<style type="text/css"><!--

body {
  font-family: "Open Sans", Helvetica, sans-serif;
  font-weight: 100;
  color: #ccc;
  background: rgba(10, 24, 55, 1);
  font-size: 16px;
}

h2, h3, h4 {
  font-weight: 200;
}
```

but you cant browse it with the server1using curl .you have to do port forwarding to established that connection.

```
[root@server1 ~]#
[root@server1 ~]# curl 192.168.0.11
curl: (7) Failed connect to 192.168.0.11:80; No route to host
[root@server1 ~]#
```

So if we forward the port 80 of the server2 to port 8000 in server1 we can access the content of the web server in server2 with server1in port 8000

command from server1:
=>***ssh -L 8000:localhost:80*** *root@192.168.0.11*

```
[root@server2 ~]# ssh -L 8000:localhost:80 root@192.168.0.10
The authenticity of host '192.168.0.10 (192.168.0.10)' can't be established.
ECDSA key fingerprint is SHA256:Vb8jzXFWtxe/Z7yco6NR2IPPJ+1uotVhlseVEx+/e2o.
ECDSA key fingerprint is MD5:bd:62:cb:ab:28:3b:ad:47:61:da:b5:8f:d8:b6:85:4c.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.10' (ECDSA) to the list of known hosts.
root@192.168.0.10's password:
Last login: Sat Sep  7 16:44:29 2019 from 192.168.0.6
[root@server1 ~]#
```

it will forward the port and we can access the resources from server1.it can be very useful for accessing a file that is behind a firewall.

```
[root@server1 ~]#
[root@server1 ~]# curl localhost:8000
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd"><html><head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
               <title>Apache HTTP Server Test Page powered by CentOS</title>
               <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

    <!-- Bootstrap -->
    <link href="/noindex/css/bootstrap.min.css" rel="stylesheet">
    <link rel="stylesheet" href="noindex/css/open-sans.css" type="text/css" />

<style type="text/css"><!--
```

Configuration

ssh server and configuration file is in the '/etc/ssh/' directory.

1) ***'sshd_config'*** is the ssh server configuration file
2) ***'ssh_config'*** is the ssh client configuration file

Lets see the server configuration file and important propertise

vim /etc/ssh/sshd_server

PasswordAuthentication yes

Port 22

PubkeyAuthentication yes

X11Forwarding yes

PermitRootLogin no

→ you can change the port from 22 to any port you want but default is 22
→ password authentication is set to no for some cloud server Because the use public private key pair which is more secure
→ X11 forwarding is by default set to yes. if you want to work with a gui interface this will let you do this
→ Permit root login is set to no. It should be always set to no because root login can make major security risk

***

TELNET

CENTOS CONFIGURATION

What is Telnet?

Telnet is a network protocol that is used to connect to remote computer over TCP/IP based network .it use port ***23*** by default. Its basically used for remote administration .when you connect to the other computer with telnet it will allow you to communicate with the host from your local system.

Problems With Telnet

There are some security vulnerability in telnet Because
→ It Transmit login data in a clear format .Its not encrypted. Everything is sent in plain text
→ it is nor recommended to use telnet over public network (WAN)

→ better alternative is the SSH which is encrypted.

Telnet Server Install(Centos)

First step

1) install the telnet client and the telnet server

=> ***yum install telnet telnet-server***

Second step

2) enable the telnet service in boot time

=> ***systemctl enable telnet.socket***
=> ***systemctl start telnet.socket***

Third step

3) Enable Telnet in Firewall

=> ***firewall-cmd –premanent --add-port=23/tcp***
=> ***firewall-cmd --reload***

Fourth step

4) Create user [root login is disabled by default]

=> ***useradd <user_name>***
=> ***passwd <user_name>***

This is the end of server side configuration

Telnet Client Install (centos)

First step

1) install the telnet client

=> ***yum install telnet***

Second step

2) Connect to the system

=>***telnet <server_ip_address>***

example:

=>***telnet 192.168.0.100***

UBUNTU CONFIGURATION

Telnet Server Install(Ubuntu)

First Step

1) install the telnet client and the telnet server

=> ***apt install telnetd xinetd -y***

Second Step

2) restart xinetd service

=> ***systemctl restart xinetd***

The service should be fired-up automatically once the installation is done.

Third Step

3) check the service status

=>***systemctl status xinetd***

Fourth Step

4) Enable Telnet in Firewall .Telnet works at port ***23***. so add the port

=> ***ufw allow 23***

=> ***ufw reload***

[ufw is the firewall used in ubuntu/debian server]

[root login is disabled by default]

This is the end of server side configuration

Telnet Client Install (Ubuntu)

First Step

1) install the telnet client

=> ***apt install telnet -y***

Second Step

2) Connect to the system

=>***telnet <server_ip_address>***

example:

=>***telnet 192.168.0.100***

***

DISK MANAGEMENT

Hard drive provide spaces .before working with the hard drive we have to divide it into pieces .it can be just one giant piece (means one partition) of it can be divide into multiple pieces (multiple partition).for example we can divide it to four primary partition we wan divide it more with extended partition with different size. And after that each partition could be formatted in an way that windows can recognize it another could be formatted just like the linux and so on. Each individual pieces works as a file system .where different data is stored and we can work with it. To work with the partition we inserted a drive. we can do it physically or if you are on a virtual machine you can add blank drive. After adding the drive (can be physical can be virtual) we can show the status by this command

=>***sudo fdisk -l***

and to see the block drives we can use the command

=>***lsblk***

for my computer I have added two virtual drives so the

results for

my computer is like this

```
[vagrant@localhost ~]$ lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE
MOUNTPOINT
sda               8:0    0  9.9G  0 disk
├─sda1            8:1    0  500M  0 part /boot
└─sda2            8:2    0  9.4G  0 part
  ├─centos-root 253:0    0  8.4G  0 lvm  /
  └─centos-swap 253:1    0 1016M  0 lvm
[SWAP]
sdb               8:16   0   30G  0 disk
sdc               8:32   0   30G  0 disk
sr0              11:0    1 1024M  0 rom
sr1              11:1    1 1024M  0 rom
```

so we have block devices ***sdb*** and ***sdc*** both 30 gigabytes. and its completely blank. its just a raw disk. So these are the block devices it has not done any partition yet. The swap partition in the table are work as a virtual memory to support the ram .in case of ram is out of memory its helps ram to not going out of ram.

to create partition in block sdb

the command is:

=> ***sudo fdisk /dev/sdb***

then to see the command we have to type the 'm'

```
Command (m for help): m
Command action
   a   toggle a bootable flag
   b   edit bsd disklabel
   c   toggle the dos compatibility flag
   d   delete a partition
   g   create a new empty GPT partition table
   G   create an IRIX (SGI) partition table
   l   list known partition types
   m   print this menu
   n   add a new partition
   o   create a new empty DOS partition table
   p   print the partition table
   q   quit without saving changes
   s   create a new empty Sun disklabel
   t   change a partition's system id
   u   change display/entry units
```

to create the partition first enter p to print the table to see weather we are in the wrong block. after assuring that

→ *type 'n' n for new partition*

→ *type 'p' for primary and give the partition number 1*

→ *press enter for starting from the beginning from the drive*

→ *allocate the size*

→ *enter "+<size G/M/K>"*

→ *press enter*

→ *press 'w' to save it*

Command (m for help): n
Partition number (1-128, default 1): 1
First sector (2048-62914526, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-62914526, default 62914526): +10G
Created partition 1

Command (m for help): w

The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.

Partition id

partition id is another important thing by default the partition id is ***83*** which actually for linux partition .to change it on fdisk we have to type 't' for type and press 'L' for the list of the id .then give the partition number and then type the partition id and after that we type 'w' for write. For example we need to make the swap partition so we have to apply the following command.

```
Command (m for help): n
Partition number (2-128, default 2): 2
First s 0973568-62914526, default 20973568):
Last sector, +sectors or +size{K,M,G,T,P} (20973568-
62914526, default 62914526): +4G
Created partition 2

Command (m for help): t
Partition number (1,2, default 2): 2
Partition type (type L to list all types): 14
Changed type of partition 'Linux filesystem' to 'Linux
swap'
```

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

Here 14 is used for swap but for mordan system it is 82 it is always good to check the id .To check the status we have to use the l***sblk*** command

=>***lsblk***

```
sda           8:0    0  9.9G  0 disk
├─sda1        8:1    0  500M  0 part /boot
└─sda2        8:2    0  9.4G  0 part
  ├─centos-root 253:0    0  8.4G  0 lvm  /
  └─centos-swap 253:1    0 1016M  0 lvm  [SWAP]
sdb           8:16   0   30G  0 disk
├─sdb1        8:17   0   10G  0 part
└─sdb2        8:18   0    4G  0 part
sdc           8:32   0   30G  0 disk
sr0          11:0    1 1024M  0 rom
sr1          11:1    1 1024M  0 rom
```

Create File system

after creating partition the next thing we have to do is creating file system. To create an ext4 file system in sdb1

the command is

=> ***sudo mkfs.ext4 /dev/sdb1***

it will make the ext4 file system .To make a swap file system int sdb2 we have to do this command

=> ***sudo mkswap /dev/sdb2***

we can use the ext2 ext3 xfs and riserfs .the command is

→ ***sudo mkfs -t ext2 /dev/sdb2***

→ ***sudo mkfs -t ext3 /dev/sdb2***

→ ***sudo mkfs -t xfs /dev/sdb2***

→ ***sudo mkfs -t riserfs /dev/sdb2***

mounting the drive

after creating the file system we have to mount it on a folder to use it. To mount it

first we have to create a folder then use the command

=> ***sudo mkdir /first_drive***

=> ***sudo mount /dev/sdb1 /first_drive/***

=> ***cd /first_drive/***

if we fount '***lost+found***' directory we can assume that it is successfully added .

RAID

What is RAID ?

RAID stands for "***Redundant Array of Independent Disk***". Fault tolerance is a very important thing in server administration. Data loss like disk failure can have a serious impact on the industry that's why need redundancy for the data to make sure if one disk fails for any reason we must have the backup. That's why system administrators employ multiple hard drive for ensuring the the data reliability and with a organized hard drive .Ina raid setup data is not stored in a single disk it stored in multiple disk.

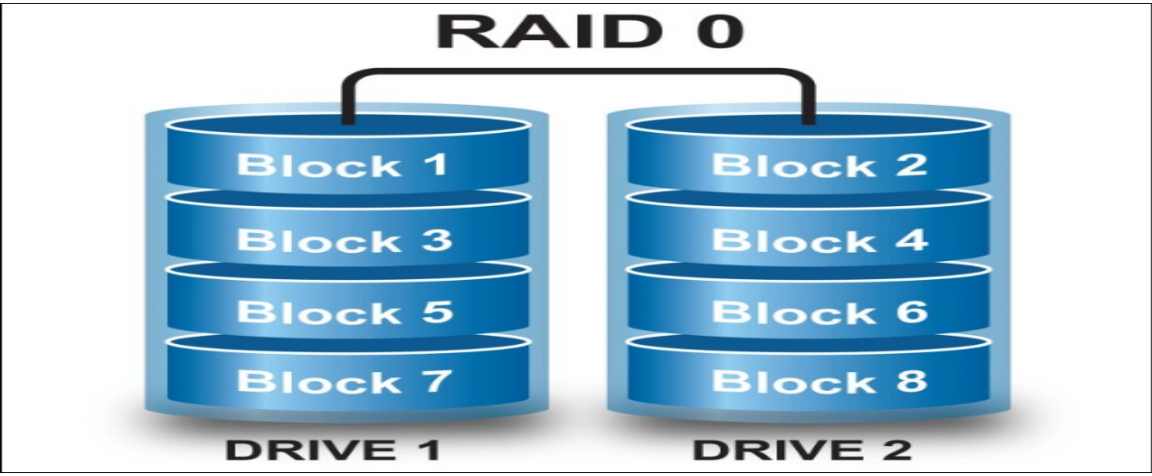
There are Four common Raid

→ ***Raid 0*** ***(Not Fault tolerant)***
→ ***Raid 1*** ***(Fault tolerant)***
→ ***Raid 5*** ***(Fault tolerant)***

→ ***Raid 10 (Fault tolerant)***

Raid 0

Raid 0 is not a fault tolerant .Even the Raid 0 should not be called RAID cause it does not fulfill the main target of RAID. Its actuall y called Striping .In RAID 0 data is stored or spread into two separate disk .It treats the two hard drive like a single hard drive and store the data .So By any chance if any of the disk failed or data is removed or

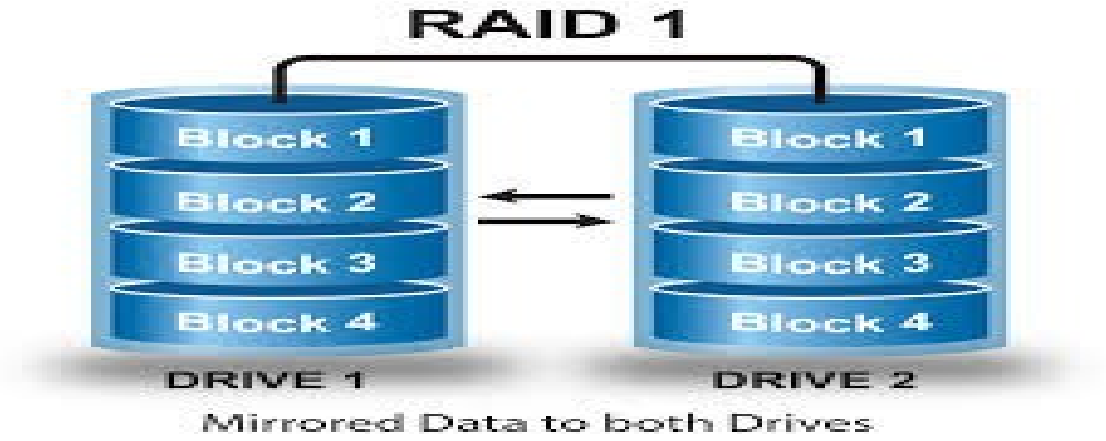


data become damaged there is no way that the data even get recovered ,So now the question arrives why we use the RAID 0

The main advantage of using RAID 0 is “SPEED”.Because when you use multiple disk controller instead of one Accessing data become faster

Raid 1

Raid 0 is fault tolerant . RAID 1 is called MIRRORING .in mirroring data is written to each RAID devices .Each disk has a complete copy of data of the other .so if one disk fails you can access the same data from the other disk.

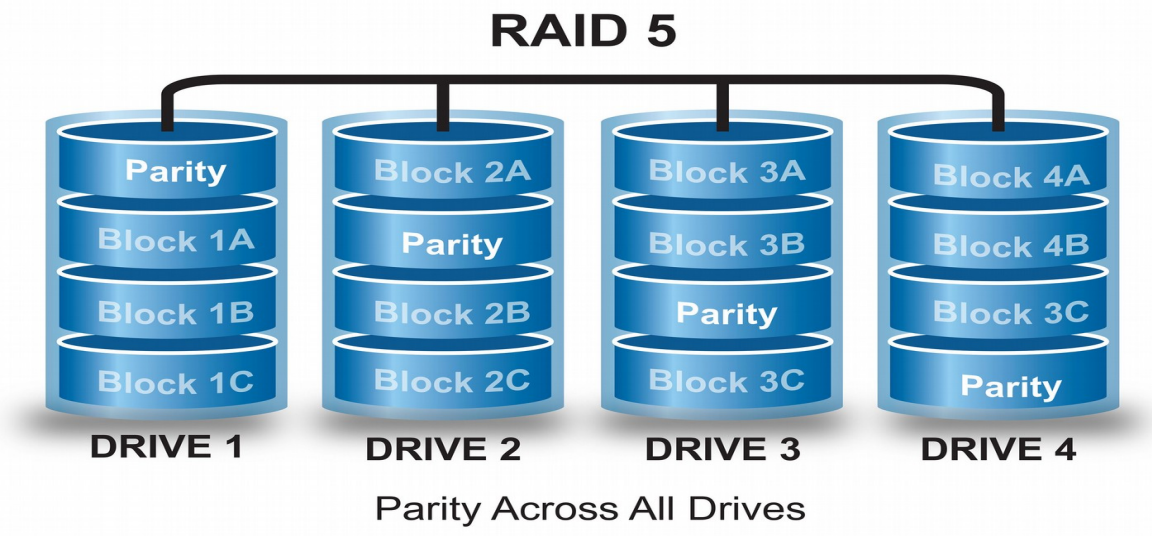


Its extremely safe . But it is very inefficient. Because it consumes Double the size of space for data. For example to store a 80GB of data you need 160 Gb of storage and since data has to write in multiple disk that's why its a slow process

Raid 5:

Raid 5 is also fault tolerant . Its a alternative to the mirroring .It does not save the data with full duplication but with parity information. Parity information takes one drive that can be used to recover the data in case of data loss. Thats why you need to have three or more disk for RAID 5.That's the very popular method for storing disk .The parity in formation is evenly spread through the disk. The downside of the data is the

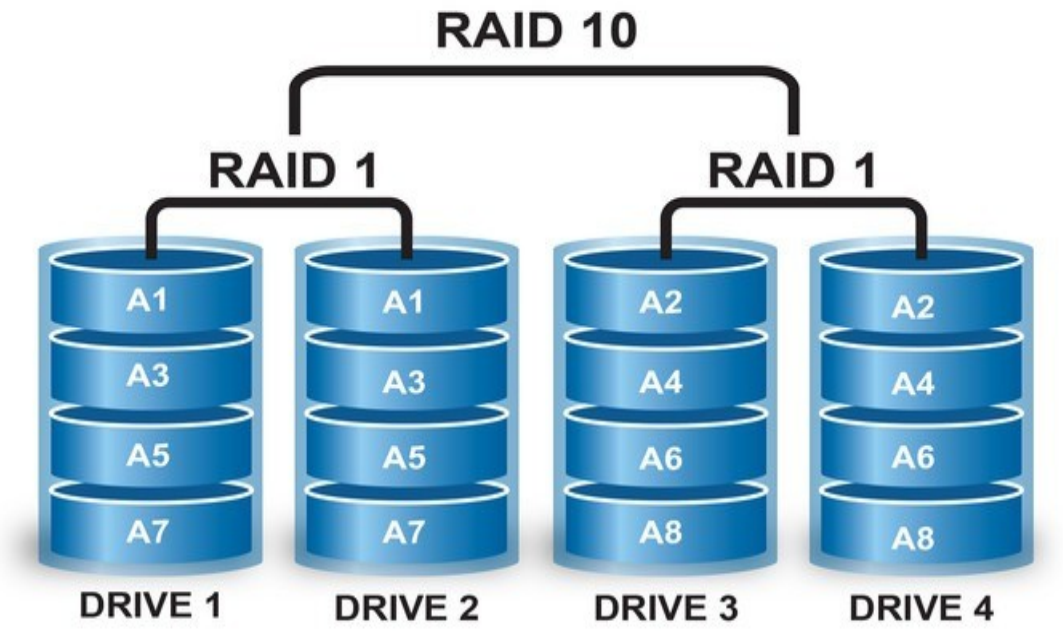
parity takes a complete 1 drive equivalent .That means if you gave 4 disk of 1TB then you can only store 3TB of data in the disk with RAID 5.This is the combination of



the striping and the parity. RAID provides faster access and recover capability making it the most used redundancy approach for servers.

Raid 10 (1+0):

Raid 10 is actually RAID 1 + RAID 0 .It used both technique for storing data. you have to used a minimum 4 disk to implement RAID 10.In RAID 10 data is stripped in multiple disk like RAID 0 but each disk has a exact copy in another disk like raid 1.



So its a combination of striping and mirroring . So RAID 10 gives us the fault tolerance of the RAID 1 and speed of the RAID 0 .But the downside is you can only use the half of your total storage of you implement RAID 10.

Creating RAID 0 in CENTOS 7

RAID 0 is not fault tolerant but it has some advantage

→ ***it is high performance***
→ ***no space will be wasted***
→ ***reading and writing speed will be Fast***

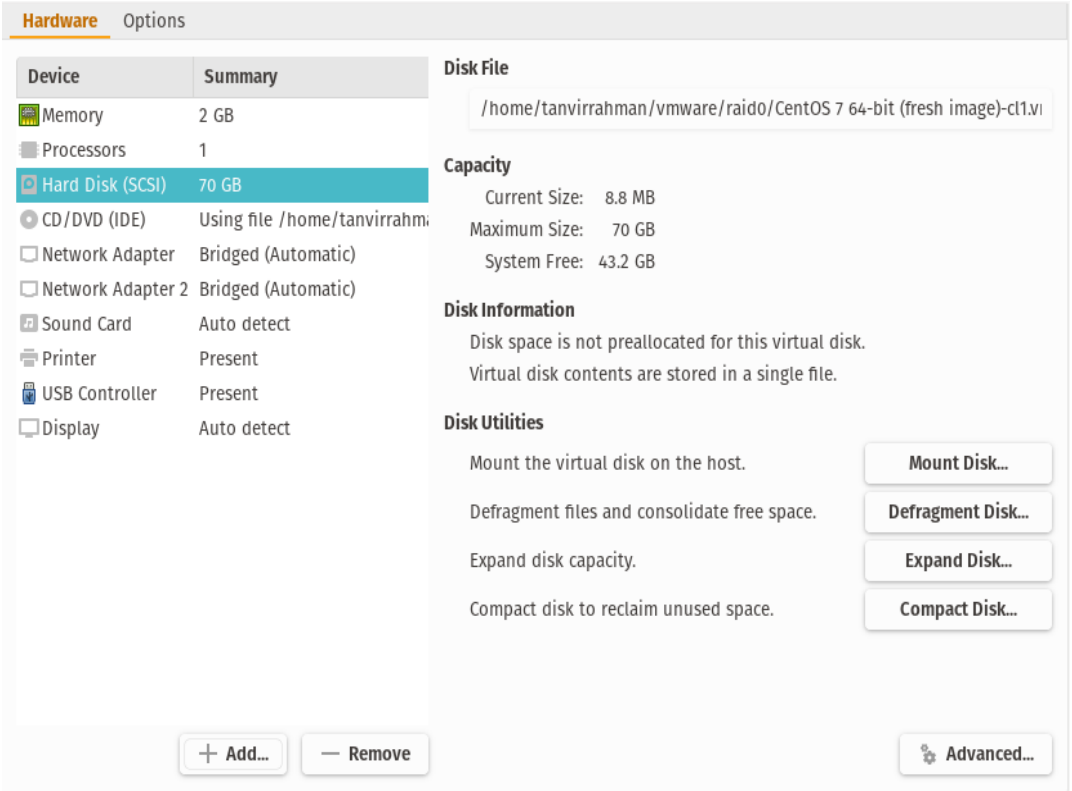
Setting up RAID 0 in Virtual Machine :

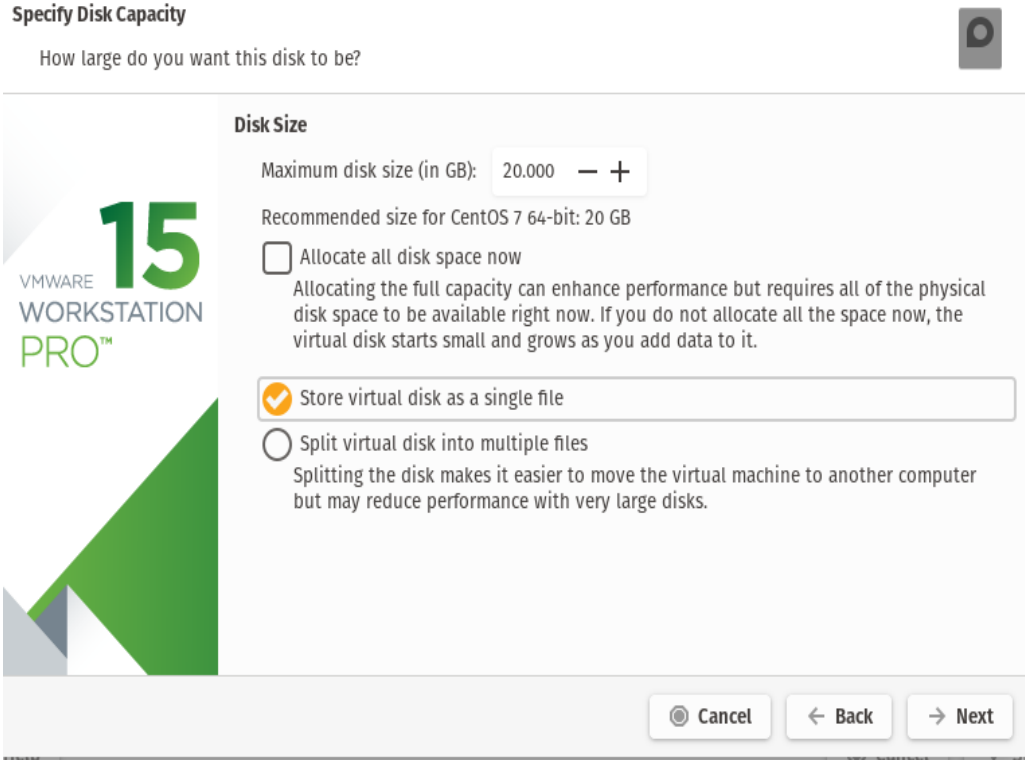
Requirements:

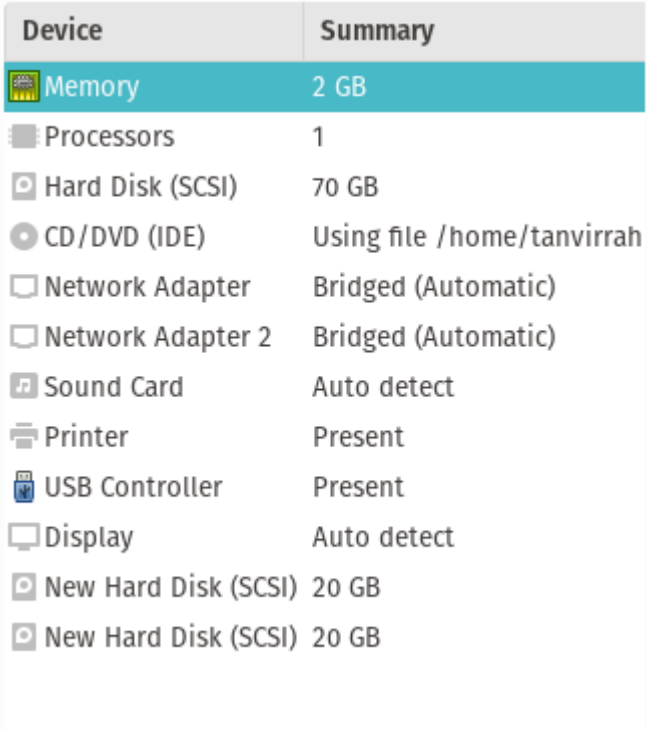
→ ***Virtual Machine***
→ ***Two disk***
→ ***internet connection***
→ ***a static ip address (in case you want to ssh the server)***

Step 1

Adding two 20GB disk in the centos7 Virtual machine.


Hardware
Options
Device
Summary
Memory
2 GB
Processors
1
Hard Disk (SCSI)
70 GB
CD/DVD (IDE)
Using file /home/tanvirrahm
Network Adapter
Bridged (Automatic)
Network Adapter 2
Bridged (Automatic)
Sound Card
Auto detect
Printer
Present
USB Controller
Present
Display
Auto detect
Add...
Remove
Disk File
/home/tanvirrahman/vmware/raid0/CentOS 7 64-bit (fresh image)-cl1.vi
Capacity
Current Size: 8.8 MB
Maximum Size: 70 GB
System Free: 43.2 GB
Disk Information
Disk space is not preallocated for this virtual disk.
Virtual disk contents are stored in a single file.
Disk Utilities
Mount the virtual disk on the host.
Mount Disk...
Defragment files and consolidate free space.
Defragment Disk...
Expand disk capacity.
Expand Disk...
Compact disk to reclaim unused space.
Compact Disk...
Advanced...


Specify Disk Capacity
How large do you want this disk to be?
Disk Size
Maximum disk size (in GB): 20.000
Recommended size for CentOS 7 64-bit: 20 GB
Allocate all disk space now
Allocating the full capacity can enhance performance but requires all of the physical disk space to be available right now. If you do not allocate all the space now, the virtual disk starts small and grows as you add data to it.
Store virtual disk as a single file
Split virtual disk into multiple files
Splitting the disk makes it easier to move the virtual machine to another computer but may reduce performance with very large disks.
VMWARE 15 WORKSTATION PRO™
Cancel
Back
Next


Device
Summary
Memory 2 GB
Processors 1
Hard Disk (SCSI) 70 GB
CD/DVD (IDE) Using file /home/tanvirrah
Network Adapter Bridged (Automatic)
Network Adapter 2 Bridged (Automatic)
Sound Card Auto detect
Printer Present
USB Controller Present
Display Auto detect
New Hard Disk (SCSI) 20 GB
New Hard Disk (SCSI) 20 GB

Step 2

Boot the machine.

Step 3

open Terminal .(or you just ssh the server from the host)

Step 4

apply the ***'lsblk'*** command to see the block devices

=>***lsblk***

```
[root@server2 ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm  /
  ├─centos-swap 253:1    0    2G  0 lvm  [SWAP]
  └─centos-home 253:2    0   22G  0 lvm  /home
sdb               8:16   0   20G  0 disk
sdc               8:32   0   20G  0 disk
sr0              11:0    1  4.3G  0 rom
```

There are two additional block devices name 'sdb' and 'sdc' er use this two drie to make a raid 0.

Step 5

install the ***mdadm*** packge

=>***yum update***

=> ***yum install mdadm -y***

Step 6

check the version in the of the packages

=> ***mdadm –version***

Step 7

Examine the hard drive with mdadm

=> ***mdadm –examine /dev/sd[b-c]***

```
[root@server2 ~]# mdadm --examine /dev/sd[b-c]
mdadm: No md superblock detected on /dev/sdb.
mdadm: No md superblock detected on /dev/sdc.
[root@server2 ~]#
```

Step 8

Create partition for RAID

=>***fdisk /dev/sdb***

Follow below instructions for creating partitions.

1. Press '***n***' for creating new partition.
2. Then choose '***P***' for Primary partition.
3. Next select the partition number as ***1***.
4. Give the default value by just pressing two times ***Enter*** key.

5. Next press '***P***' to print the defined partition.

Follow below instructions for creating Linux raid auto on partitions.

1. Press '***L***' to list all available types.
2. Type '***t***'to choose the partitions.
3. Choose '***fd***' for Linux raid auto and press Enter to apply.
4. Then again use '***P***' to print the changes what we have made.

5. Use '***w***' to write the changes.

[creating partition]

```
[root@server2 ~]#
[root@server2 ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

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

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0xc4707f2b.

Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-41943039, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-41943039, default 41943039):
Using default value 41943039
Partition 1 of type Linux and of size 20 GiB is set

Command (m for help): p

Disk /dev/sdb: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0xc4707f2b

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            2048    41943039    20970496   83  Linux

Command (m for help):
```

[creating raid on that paririon]

```
[root@server2 ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

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


Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): fd
Changed type of partition 'Linux' to 'Linux raid autodetect'

Command (m for help): P

Disk /dev/sdb: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0xc4707f2b

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            2048    41943039    20970496   fd  Linux raid autodetect

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
[root@server2 ~]#
```

[***see the block devices***]

Step 9

Do the step 8 for the 'sdc'
=>***fdisk /dev/sdc***

Step 10

Examine with the '***lsblk***'

=>***lsblk***

```
[root@server2 ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm  /
  ├─centos-swap 253:1    0    2G  0 lvm  [SWAP]
  └─centos-home 253:2    0   22G  0 lvm  /home
sdb               8:16   0   20G  0 disk
└─sdb1            8:17   0   20G  0 part
sdc               8:32   0   20G  0 disk
└─sdc1            8:33   0   20G  0 part
sr0              11:0    1  4.3G  0 rom
```

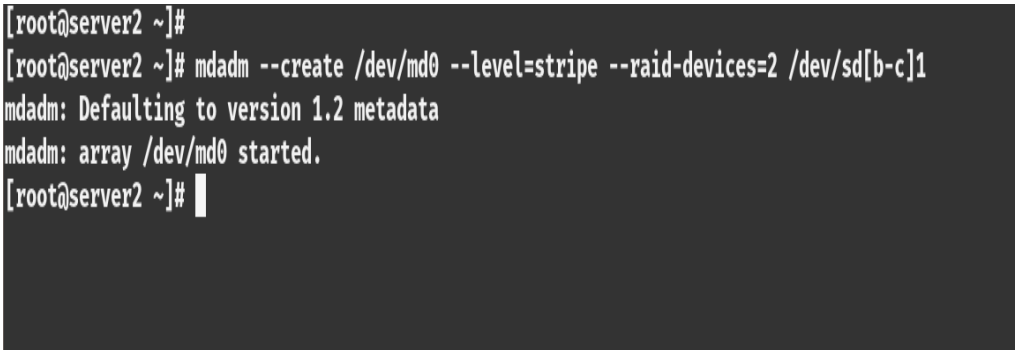
Step 11

Examine with the ***'mdadm'***

```
[root@server2 ~]# mdadm --examine /dev/sd[b-c]1
mdadm: No md superblock detected on /dev/sdb1.
mdadm: No md superblock detected on /dev/sdc1.
[root@server2 ~]#
[root@server2 ~]#
```

Step 12

Create RAID md Devices
=>***mdadm --create /dev/md0 --level=stripe --raid-devices=2 /dev/sd[b-c]1***



Step 13

See the Details of the RAID 0 devices
=>***mdadm –detail /dev/md0***

Step 14

Assigning File partition on the File system
=>***mkfs.ext4 /dev/md0***

```
[root@server2 ~]# mkfs.ext4 /dev/md0
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=128 blocks, Stripe width=256 blocks
2621440 inodes, 10476544 blocks
523827 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2157969408
320 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

Step 15

mount the volume

=>***mkdir /mnt/raid0***

=>***mount /dev/md0 /mnt/raid0***

Step 16

check the mounted volume

=>***df -h***

```
[root@server2 ~]# df -h
Filesystem               Size  Used Avail Use% Mounted on
/dev/mapper/centos-root   45G  3.8G   42G   9% /
devtmpfs                 974M     0  974M   0% /dev
tmpfs                    991M     0  991M   0% /dev/shm
tmpfs                    991M   11M  981M   2% /run
tmpfs                    991M     0  991M   0% /sys/fs/cgroup
/dev/sda1               1014M  166M  849M  17% /boot
/dev/mapper/centos-home   22G   39M   22G   1% /home
tmpfs                    199M   12K  199M   1% /run/user/42
tmpfs                    199M     0  199M   0% /run/user/0
/dev/md0                  40G   49M   38G   1% /mnt/raid0
[root@server2 ~]#
```

Step 17

check the block devices with lsblk

=>***lsblk***

```
[root@server2 ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part  /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm   /
  ├─centos-swap 253:1    0    2G  0 lvm   [SWAP]
  └─centos-home 253:2    0   22G  0 lvm   /home
sdb               8:16   0   20G  0 disk
└─sdb1            8:17   0   20G  0 part
  └─md0           9:0    0   40G  0 raid0 /mnt/raid0
sdc               8:32   0   20G  0 disk
└─sdc1            8:33   0   20G  0 part
  └─md0           9:0    0   40G  0 raid0 /mnt/raid0
sr0              11:0    1  4.3G  0 rom
[root@server2 ~]#
```

Creating RAID 1 in CENTOS 7

RAID 0 is not fault tolerant but it has some advantage

→ ***it is high performance***
→ ***no space will be wasted***
→ ***reading and writing speed will be Fast***

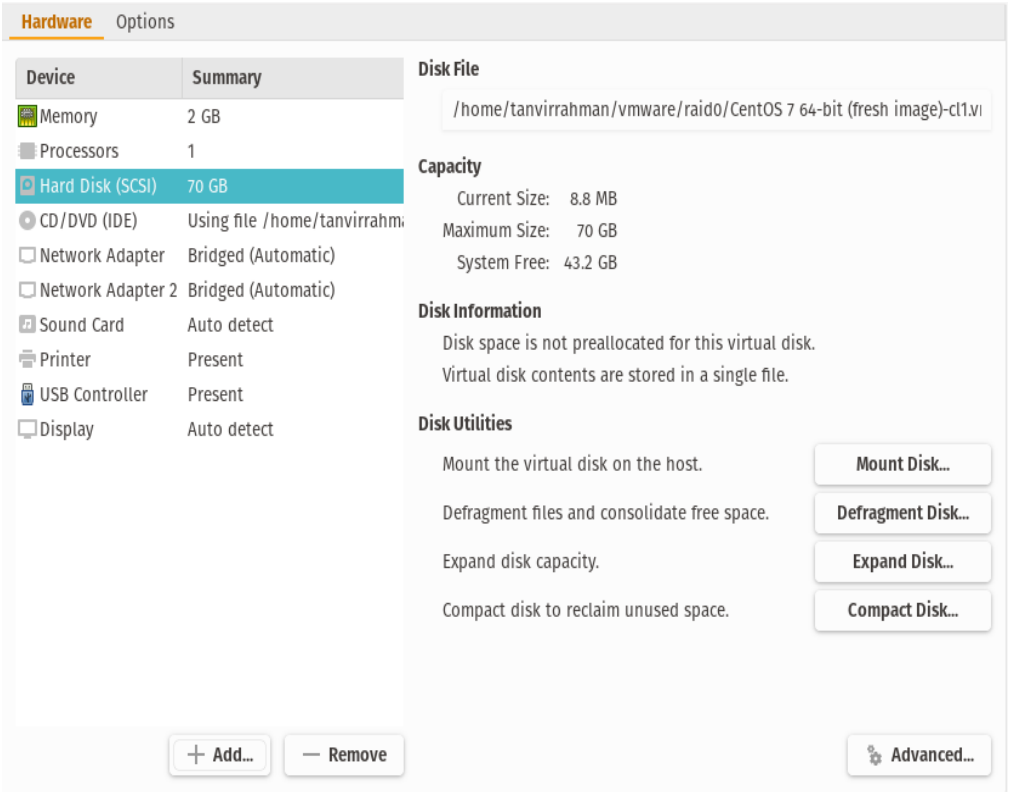
Setting up RAID 0 in Virtual Machine :

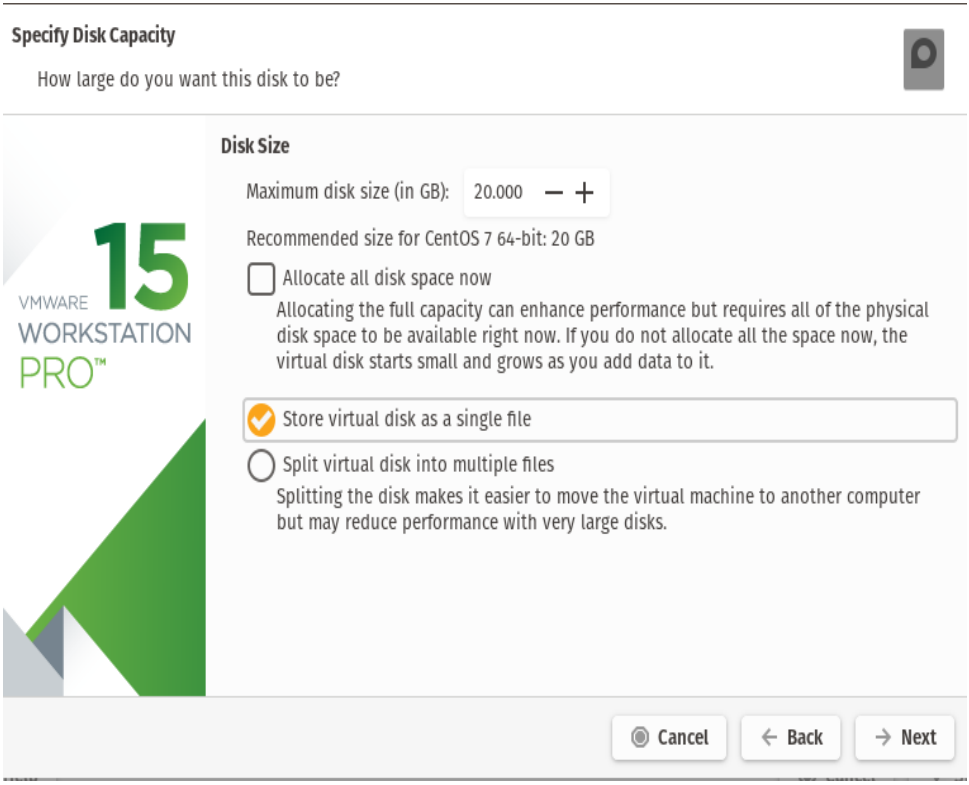
Requirements:

→ ***Virtual Machine***
→ ***Two disk***
→ ***internet connection***
→ ***a static ip address (in case you want to ssh the server)***

Step 1

Adding two 20GB disk in the centos7 Virtual machine.


Hardware
Options
Device
Summary
Memory
2 GB
Processors
1
Hard Disk (SCSI)
70 GB
CD/DVD (IDE)
Using file /home/tanvirrahm
Network Adapter
Bridged (Automatic)
Network Adapter 2
Bridged (Automatic)
Sound Card
Auto detect
Printer
Present
USB Controller
Present
Display
Auto detect
Add...
Remove
Disk File
/home/tanvirrahman/vmware/raid0/CentOS 7 64-bit (fresh image)-cl1.vi
Capacity
Current Size: 8.8 MB
Maximum Size: 70 GB
System Free: 43.2 GB
Disk Information
Disk space is not preallocated for this virtual disk.
Virtual disk contents are stored in a single file.
Disk Utilities
Mount the virtual disk on the host.
Mount Disk...
Defragment files and consolidate free space.
Defragment Disk...
Expand disk capacity.
Expand Disk...
Compact disk to reclaim unused space.
Compact Disk...
Advanced...



Device	Summary
Memory	2 GB
Processors	1
Hard Disk (SCSI)	70 GB
CD/DVD (IDE)	Using file /home/tanvirrah
Network Adapter	Bridged (Automatic)
Network Adapter 2	Bridged (Automatic)
Sound Card	Auto detect
Printer	Present
USB Controller	Present
Display	Auto detect
New Hard Disk (SCSI)	20 GB
New Hard Disk (SCSI)	20 GB

Step 2

Boot the machine.

Step 3

open Terminal .(or you just ssh the server from the host)

Step 4

apply the ***'lsblk'*** command to see the block devices

=>***lsblk***

```
[root@server2 ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm  /
  ├─centos-swap 253:1    0    2G  0 lvm  [SWAP]
  └─centos-home 253:2    0   22G  0 lvm  /home
sdb               8:16   0   20G  0 disk
sdc               8:32   0   20G  0 disk
sr0              11:0    1  4.3G  0 rom
```

There are two additional block devices name 'sdb' and 'sdc' er use this two drive to make a raid 0.

Step 5

install the ***mdadm*** packge

=>***yum update***

=> ***yum install mdadm -y***

Step 6

check the version in the of the packages

=> ***mdadm –version***

Step 7

Examine the hard drive with mdadm

=> ***mdadm –examine /dev/sd[b-c]***

```
[root@server2 ~]# mdadm --examine /dev/sd[b-c]
mdadm: No md superblock detected on /dev/sdb.
mdadm: No md superblock detected on /dev/sdc.
[root@server2 ~]#
```

Step 8

Create partition for RAID

=>***fdisk /dev/sdb***

Follow below instructions for creating partitions.

1. Press '***n***' for creating new partition.
2. Then choose '***P***' for Primary partition.
3. Next select the partition number as ***1***.
4. Give the default value by just pressing two times ***Enter*** key.
5. Next press '***P***' to print the defined partition.

Follow below instructions for creating Linux raid auto on partitions.

1. Press '***L***' to list all available types.
2. Type '***t***'to choose the partitions.
3. Choose '***fd***' for Linux raid auto and press Enter to apply.
4. Then again use '***P***' to print the changes what we have made.
5. Use '***w***' to write the changes.

[creating partition]

```
[root@server2 ~]#
[root@server2 ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

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

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0xc4707f2b.

Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-41943039, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-41943039, default 41943039):
Using default value 41943039
Partition 1 of type Linux and of size 20 GiB is set

Command (m for help): p

Disk /dev/sdb: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0xc4707f2b

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            2048    41943039    20970496   83  Linux

Command (m for help):
```

[creating raid on that partition]

```
[root@server2 ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

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


Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): fd
Changed type of partition 'Linux' to 'Linux raid autodetect'

Command (m for help): P

Disk /dev/sdb: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0xc4707f2b

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            2048    41943039    20970496   fd  Linux raid autodetect

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
[root@server2 ~]#
```

*[**see the block devices**]*

Step 9

Do the step 8 for the ***'sdc'***

*=>**fdisk /dev/sdc***

Step 10

Examine with the ***'lsblk'***

=>***lsblk***

```
[root@server2 ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm  /
  ├─centos-swap 253:1    0    2G  0 lvm  [SWAP]
  └─centos-home 253:2    0   22G  0 lvm  /home
sdb               8:16   0   20G  0 disk
└─sdb1            8:17   0   20G  0 part
sdc               8:32   0   20G  0 disk
└─sdc1            8:33   0   20G  0 part
sr0              11:0    1  4.3G  0 rom
```

Step 11

Examine with the '***mdadm***'

```
[root@server2 ~]# mdadm --examine /dev/sd[b-c]1
mdadm: No md superblock detected on /dev/sdb1.
mdadm: No md superblock detected on /dev/sdc1.
[root@server2 ~]#
[root@server2 ~]#
```

Step 12

Create RAID md Devices (with mirror)

=>***mdadm --create /dev/md0 --level=mirror --raid-devices=2 /dev/sd[b-c]1***

Step 13

See the Details of the RAID 0 devices

=>***mdadm –detail /dev/md0***

Step 14

Assigning File partition on the File system

*=>**mkfs.ext4 /dev/md0***

```
[root@server2 ~]# mkfs.ext4 /dev/md0
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=128 blocks, Stripe width=256 blocks
2621440 inodes, 10476544 blocks
523827 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2157969408
320 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

Step 15

mount the volume

*=>**mkdir /mnt/raid0***
*=>**mount /dev/md0 /mnt/raid0***

Step 16

check the mounted volume

=>***df -h***

```
[root@server2 ~]# df -h
Filesystem               Size  Used Avail Use% Mounted on
/dev/mapper/centos-root   45G  3.8G   42G   9% /
devtmpfs                 974M     0  974M   0% /dev
tmpfs                    991M     0  991M   0% /dev/shm
tmpfs                    991M   11M  981M   2% /run
tmpfs                    991M     0  991M   0% /sys/fs/cgroup
/dev/sda1               1014M  166M  849M  17% /boot
/dev/mapper/centos-home   22G   39M   22G   1% /home
tmpfs                    199M   12K  199M   1% /run/user/42
tmpfs                    199M     0  199M   0% /run/user/0
/dev/md0                  20G   45M   19G   1% /mnt/raid1
[root@server2 ~]#
```

Step 17

check the block devices with lsblk

=>***lsblk***

```
[root@server2 ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part  /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm   /
  ├─centos-swap 253:1    0    2G  0 lvm   [SWAP]
  └─centos-home 253:2    0   22G  0 lvm   /home
sdb               8:16   0   20G  0 disk
└─sdb1            8:17   0   20G  0 part
  └─md0           9:0    0   20G  0 raid1 /mnt/raid1
sdc               8:32   0   20G  0 disk
└─sdc1            8:33   0   20G  0 part
  └─md0           9:0    0   20G  0 raid1 /mnt/raid1
sr0              11:0    1  4.3G  0 rom
[root@server2 ~]#
```

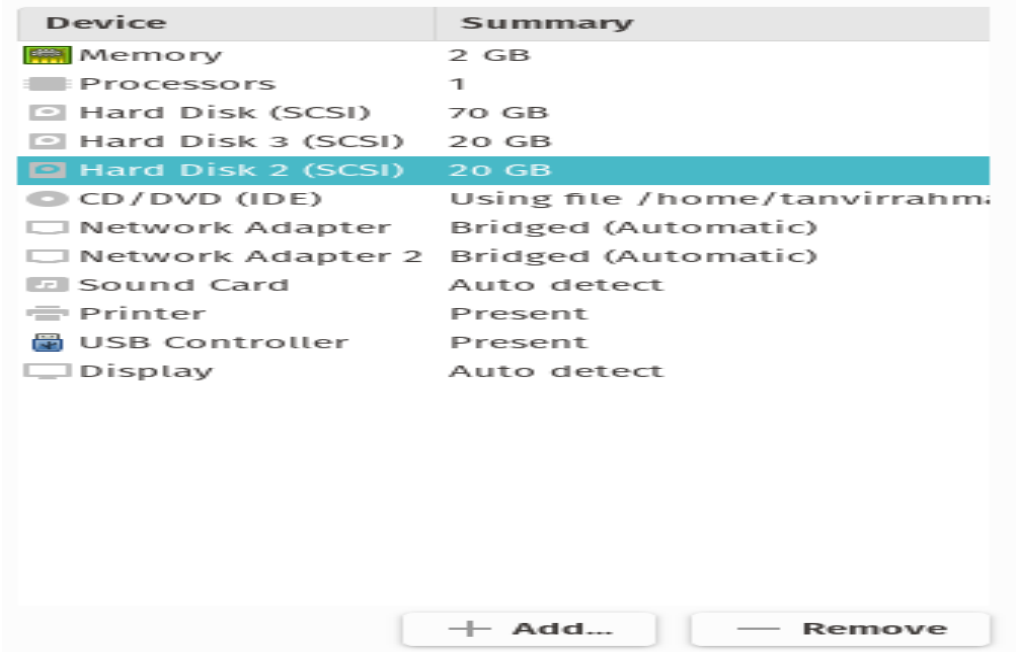
Step 18

Create a file inside the raid devices. To check that if one device is unplugged if the other have it.

```
[root@server2 raid1]# pwd
/mnt/raid1
[root@server2 raid1]# ls
hello.txt  lost+found
[root@server2 raid1]# cat hello.txt
hello
[root@server2 raid1]#
```

Step 19

unplug one device



Step 20

reboot the system and check the drive that is still connected and see if the backup is still there

```
[root@server2 ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part  /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm   /
  ├─centos-swap 253:1    0    2G  0 lvm   [SWAP]
  └─centos-home 253:2    0   22G  0 lvm   /home
sdb               8:16   0   20G  0 disk
└─sdb1            8:17   0   20G  0 part
  └─md0           9:0    0   20G  0 raid1
sr0              11:0    1  4.3G  0 rom
[root@server2 ~]#
[root@server2 ~]#
[root@server2 ~]# mount /dev/md
md/  md0
[root@server2 ~]# mount /dev/md0 /mnt/raid1
[root@server2 ~]# cd /mnt/raid1
[root@server2 raid1]# ls
hello.txt  lost+found
[root@server2 raid1]#
```

Data is still there even one disk is unplugged

Creating RAID 5 in CENTOS 7

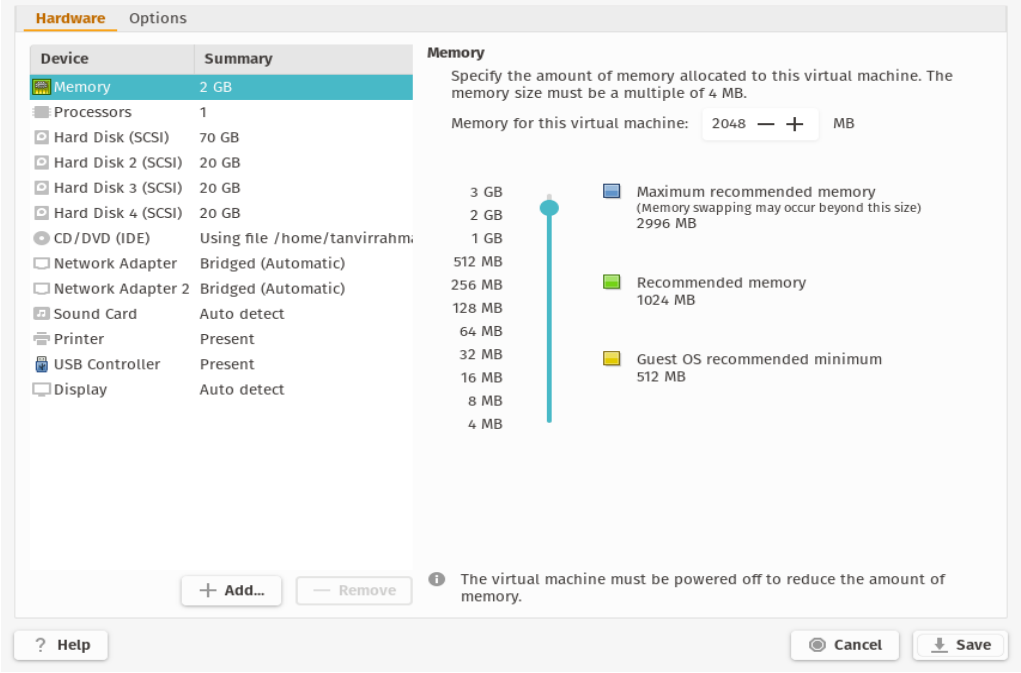
Setting up RAID 5 in Virtual Machine :

Requirements:

- → ***Virtual Machine***
- → ***Three disk***
- → ***internet connection***
- → ***a static ip address (in case you want to ssh the server)***

Step 1

Adding three 20GB disk in the centos7 Virtual machine.



Step 2

Boot the machine.

Step 3

Open Terminal .(or you just ssh the server from the server) [in this case I ssh to the server]

Step 4

apply the '***lsblk***' command to see the block devices

=>***lsblk***

```
[root@localhost ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm  /
  ├─centos-swap 253:1    0    2G  0 lvm  [SWAP]
  └─centos-home 253:2    0   22G  0 lvm  /home
sdb               8:16   0   20G  0 disk
sdc               8:32   0   20G  0 disk
sdd               8:48   0   20G  0 disk
sr0              11:0    1  4.3G  0 rom  /run/media/root/CentOS 7 x86_64
[root@localhost ~]#
```

There are three additional block devices name 'sdb' and 'sdc' and 'sdd' we use this three drive to make a raid 5.

Step 5

install the ***mdadm*** package

=>***yum update***
=> ***yum install mdadm -y***

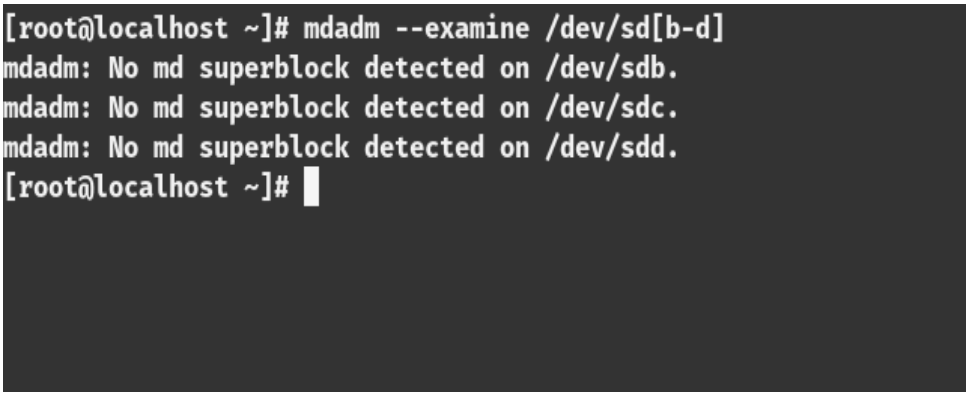
Step 6

check the version in the of the packages
=> ***mdadm –version***

Step 7

Examine the hard drive with mdadm

=> ***mdadm –examine /dev/sd[b-d]***



Step 8

Create partition for RAID

=>***fdisk /dev/sdb***

[creating raid on that partition]

```
[root@server2 ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

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


Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): fd
Changed type of partition 'Linux' to 'Linux raid autodetect'

Command (m for help): P

Disk /dev/sdb: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0xc4707f2b

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            2048    41943039    20970496   fd  Linux raid autodetect

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
[root@server2 ~]#
```

[see the block devices]

Step 9

Do the step 8 for the ‘sdc’ and ‘sdd’

=>***fdisk /dev/sdc***

=>***fdisk /dev/sdd***

Step 10

Examine with the ***‘lsblk’***

=>***lsblk***

```
[root@localhost ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm  /
  ├─centos-swap 253:1    0    2G  0 lvm  [SWAP]
  └─centos-home 253:2    0   22G  0 lvm  /home
sdb               8:16   0   20G  0 disk
└─sdb1            8:17   0   20G  0 part
sdc               8:32   0   20G  0 disk
└─sdc1            8:33   0   20G  0 part
sdd               8:48   0   20G  0 disk
└─sdd1            8:49   0   20G  0 part
sr0              11:0    1  4.3G  0 rom  /run/media/root/CentOS 7 x86_64
[root@localhost ~]#
```

Step 11

Examine with the ***'mdadm'***

```
[root@localhost ~]# mdadm --examine /dev/sd[b-d]1
mdadm: No md superblock detected on /dev/sdb1.
mdadm: No md superblock detected on /dev/sdc1.
mdadm: No md superblock detected on /dev/sdd1.
[root@localhost ~]#
```

Step 12

Create RAID md Devices (with miror)

=>mdadm --create /dev/md0 --level=5 --raid-devices=3 /dev/sd[b-d]1

```
[root@localhost ~]#
[root@localhost ~]# mdadm --create /dev/md0 --level=5 --raid-devices=3 /dev/sd[b-d]1
```

Step 13

See the Details of the RAID 0 devices

=>***mdadm –detail /dev/md0***

Step 14

Varify with this command

=>***mdadm -E /dev/sd[b-d]1 | grep raid5***

```
[root@localhost ~]# mdadm -E /dev/sd[b-d]1 | grep raid5
     Raid Level : raid5
     Raid Level : raid5
     Raid Level : raid5
[root@localhost ~]#
```

Step 15

Assigning File partition on the File system

```
[root@server2 ~]# mkfs.ext4 /dev/md0
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=128 blocks, Stripe width=256 blocks
2621440 inodes, 10476544 blocks
523827 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2157969408
320 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

=>***mkfs.ext4 /dev/md0***

Step 15

mount the volume

=>***mkdir /mnt/raid5***
=>***mount /dev/md0 /mnt/raid5***

Step 17

check the mounted volume

=>***df -h***

```
[root@localhost ~]# df -h
Filesystem               Size  Used Avail Use% Mounted on
/dev/mapper/centos-root   45G  3.6G   42G   8% /
devtmpfs                 974M     0  974M   0% /dev
tmpfs                    991M     0  991M   0% /dev/shm
tmpfs                    991M   11M  980M   2% /run
tmpfs                    991M     0  991M   0% /sys/fs/cgroup
/dev/sda1               1014M  166M  849M  17% /boot
/dev/mapper/centos-home   22G   33M   22G   1% /home
tmpfs                    199M  4.0K  199M   1% /run/user/42
tmpfs                    199M   28K  199M   1% /run/user/0
/dev/sr0                 4.3G  4.3G     0 100% /run/media/root/CentOS 7 x86_64
/dev/md0                  40G   49M   38G   1% /mnt/raid5
[root@localhost ~]#
```

Step 18

check the block devices with ***lsblk***

=>***lsblk***

```
[root@localhost ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part  /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm   /
  ├─centos-swap 253:1    0    2G  0 lvm   [SWAP]
  └─centos-home 253:2    0   22G  0 lvm   /home
sdb               8:16   0   20G  0 disk
└─sdb1            8:17   0   20G  0 part
  └─md0           9:0    0   40G  0 raid5 /mnt/raid5
sdc               8:32   0   20G  0 disk
└─sdc1            8:33   0   20G  0 part
  └─md0           9:0    0   40G  0 raid5 /mnt/raid5
sdd               8:48   0   20G  0 disk
└─sdd1            8:49   0   20G  0 part
  └─md0           9:0    0   40G  0 raid5 /mnt/raid5
sr0              11:0    1  4.3G  0 rom   /run/media/root/CentOS 7 x86_64
[root@localhost ~]#
```

Tanvir Rahman

Creating RAID 10 in CENTOS 7

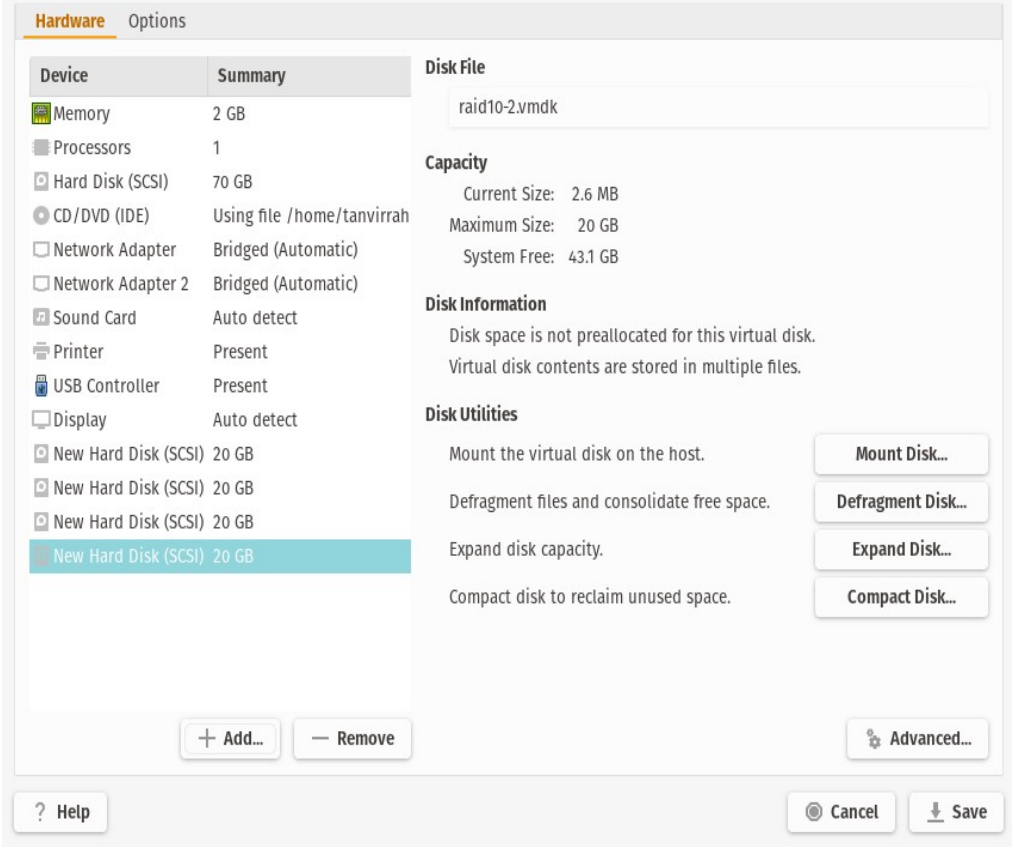
Setting up RAID 10(1+0) in Virtual Machine :

Requirements:

- → ***Virtual Machine***
- → ***Four disk(minimum)***
- → ***internet connection***
- → ***a static ip address (in case you want to ssh the server)***

Step 1

Adding four 20GB disk in the centos7 Virtual machine.



Step 2

Boot the machine.

Step 3

open Terminal .(or you just ssh the server from the server) [in this case I ssh to the server]

Step 4

apply the ***'lsblk'*** command to see the block devices

=>***lsblk***

```
[root@localhost ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm  /
  ├─centos-swap 253:1    0    2G  0 lvm  [SWAP]
  └─centos-home 253:2    0   22G  0 lvm  /home
sdb               8:16   0   20G  0 disk
sdc               8:32   0   20G  0 disk
sdd               8:48   0   20G  0 disk
sde               8:64   0   20G  0 disk
sr0              11:0    1  4.3G  0 rom
[root@localhost ~]#
```

There are three additional block devices name 'sdb' and 'sdc' and 'sdd' we use this three drive to make a raid 5.

Step 5

install the ***mdadm*** package

=>***yum update***

=> ***yum install mdadm -y***

Step 6

check the version in the of the packages

=> ***mdadm –version***

Step 7

Examine the hard drive with mdadm

=> ***mdadm –examine /dev/sd[b-e]***

Step 8

Create partition for RAID

=>***fdisk /dev/sdb***

Follow below instructions for creating partitions.

1. Press '***n***' for creating new partition.
2. Then choose '***P***' for Primary partition.
3. Next select the partition number as ***1***.

4. Give the default value by just pressing two times ***Enter*** key.

5. Next press '***P***' to print the defined partition.

Follow below instructions for creating Linux raid auto on partitions.

1. Press '***L***' to list all available types.
2. Type '***t***'to choose the partitions.
3. Choose '***fd***' for Linux raid auto and press Enter to apply.
4. Then again use '***P***' to print the changes what we have made.

5. Use '***w***' to write the changes.

[creating partition]

```
[root@server2 ~]#
[root@server2 ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

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

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0xc4707f2b.

Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-41943039, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-41943039, default 41943039):
Using default value 41943039
Partition 1 of type Linux and of size 20 GiB is set

Command (m for help): p

Disk /dev/sdb: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0xc4707f2b

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            2048    41943039    20970496   83  Linux

Command (m for help):
```

[creating raid on that partition]

```
[root@server2 ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

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


Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): fd
Changed type of partition 'Linux' to 'Linux raid autodetect'

Command (m for help): P

Disk /dev/sdb: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0xc4707f2b

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            2048    41943039    20970496   fd  Linux raid autodetect

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
[root@server2 ~]#
```

[see the block devices]

Step 9

Do the step 8 for the 'sdc' ,'sdd','sde'

=>fdisk /dev/sdc
=>fdisk /dev/sdd
=>fdisk /dev/sde

Step 10

Examine with the 'lsblk'

=>***lsblk***

```
[root@localhost ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE   MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part   /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm    /
  ├─centos-swap 253:1    0    2G  0 lvm    [SWAP]
  └─centos-home 253:2    0   22G  0 lvm    /home
sdb               8:16   0   20G  0 disk
└─sdb1            8:17   0   20G  0 part
  └─md0           9:0    0   40G  0 raid10
sdc               8:32   0   20G  0 disk
└─sdc1            8:33   0   20G  0 part
  └─md0           9:0    0   40G  0 raid10
sdd               8:48   0   20G  0 disk
└─sdd1            8:49   0   20G  0 part
  └─md0           9:0    0   40G  0 raid10
sde               8:64   0   20G  0 disk
└─sde1            8:65   0   20G  0 part
  └─md0           9:0    0   40G  0 raid10
sr0              11:0    1  4.3G  0 rom
[root@localhost ~]#
```

Step 11

Examine with the ‘mdadm’

=>***mdadm –examine /dev/sd[b-e]1***

Step 12

Create RAID md Devices (with miror)

=>mdadm --create /dev/md0 --level=10 --raid-devices=4 /dev/sd[b-e]1

Step 13

See the Details of the RAID 0 devices

=>mdadm –detail /dev/md0

```
[root@localhost ~]# mdadm --detail /dev/md0
/dev/md0:
           Version : 1.2
     Creation Time : Thu Sep  5 09:24:51 2019
        Raid Level : raid10
        Array Size : 41906176 (39.96 GiB 42.91 GB)
     Used Dev Size : 20953088 (19.98 GiB 21.46 GB)
      Raid Devices : 4
     Total Devices : 4
       Persistence : Superblock is persistent

       Update Time : Thu Sep  5 09:25:49 2019
             State : clean, resyncing
    Active Devices : 4
   Working Devices : 4
    Failed Devices : 0
     Spare Devices : 0

            Layout : near=2
        Chunk Size : 512K

Consistency Policy : resync

     Resync Status : 28% complete

              Name : localhost.localdomain:0  (local to host localhost.localdomain)
              UUID : 87cff83b:0213c1c1:bc932f37:1ae1b93d
            Events : 4

    Number   Major   Minor   RaidDevice State
       0       8       17        0      active sync set-A   /dev/sdb1
       1       8       33        1      active sync set-B   /dev/sdc1
       2       8       49        2      active sync set-A   /dev/sdd1
       3       8       65        3      active sync set-B   /dev/sde1
[root@localhost ~]#
```

Step 14

Varify with this command

*=>**mdadm -E /dev/sd[b-d]1 | grep raid5***

```
[root@localhost raid10]# mdadm -E /dev/sd[b-e]1 | grep raid10
     Raid Level : raid10
     Raid Level : raid10
     Raid Level : raid10
     Raid Level : raid10
[root@localhost raid10]#
```

Step 15

Assigning File partition on the File system

=>***mkfs.ext4 /dev/md0***

Step 16

mount the volume

=>***mkdir /mnt/raid10***

=>***mount /dev/md0 /mnt/raid10***

```
[root@localhost ~]# mkdir /mnt/raid10
[root@localhost ~]# mount /dev/md0 /mnt/raid10/
[root@localhost ~]# cd /mnt/raid10/
[root@localhost raid10]# ls
lost+found
[root@localhost raid10]#
```

Step 17

check the mounted volume

*=>**df -h***

```
[root@localhost raid10]# mdadm -E /dev/sd[b-e]1 | grep raid10
     Raid Level : raid10
     Raid Level : raid10
     Raid Level : raid10
     Raid Level : raid10
[root@localhost raid10]#
```

Step 18

check the block devices with ***lsblk***

=>***lsblk***

```
[root@localhost raid10]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE   MOUNTPOINT
sda               8:0    0   70G  0 disk
├─sda1            8:1    0    1G  0 part   /boot
└─sda2            8:2    0   69G  0 part
  ├─centos-root 253:0    0   45G  0 lvm    /
  ├─centos-swap 253:1    0    2G  0 lvm    [SWAP]
  └─centos-home 253:2    0   22G  0 lvm    /home
sdb               8:16   0   20G  0 disk
└─sdb1            8:17   0   20G  0 part
  └─md0           9:0    0   40G  0 raid10 /mnt/raid10
sdc               8:32   0   20G  0 disk
└─sdc1            8:33   0   20G  0 part
  └─md0           9:0    0   40G  0 raid10 /mnt/raid10
sdd               8:48   0   20G  0 disk
└─sdd1            8:49   0   20G  0 part
  └─md0           9:0    0   40G  0 raid10 /mnt/raid10
sde               8:64   0   20G  0 disk
└─sde1            8:65   0   20G  0 part
  └─md0           9:0    0   40G  0 raid10 /mnt/raid10
sr0              11:0    1  4.3G  0 rom
[root@localhost raid10]#
```

***

could be a subtitle

ENAS PORTITOR ANTE. Pellentesque tincidunt metus vel congue hendrerit. Aenean vehicula, odio nec dapibus auctor, mi nibh rutrum felis, nec molestie ipsum nisi nec purus. Ut felis mauris, volutpat at vulputate et, sodales sed arcu. Nunc ac velit condimentum, imperdiet turpis id, accumsan dolor. Ut ornare velit in tempus ullamcorper. Vivamus mauris velit,

congue eget ultrices eu, pharetra id nisi. Curabitur eget lacus quis justo ultricies consequat id ultricies lacus. Donec et tempor augue, at consequat ipsum. Donec sapien urna, feugiat id interdum vitae, pretium vel nisi. Nullam suscipit consectetur pulvinar. Aliquam ac libero nisl. Ut pretium nisi cursus scelerisque dictum. Sed vitae bibendum tortor.

Cras a dui at quam fringilla conse- quat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Duis at ornare libero. Nunc rutrum volutpat tortor, a faucibus augue congue non. Integer ultricies et magna nec iaculis. Sed tincidunt quis felis vel euismod. Nam pretium elit elementum est ullamcorper dic- tum. Praesent dapibus dolor ut lorem hendrerit volutpat. Curabitur vulputate placerat turpis at luctus. Suspendisse vel aliquam nulla. Nam tristique sed ipsum eget cursus. Aliquam in adipiscing magna. In quis elit nisi. Donec eu lorem ut nisl adipiscing suscipit eget eu libero.

> *The rest of this **TEMPLATE** is just dummy text that you can replace or delete.*

Maecenas scelerisque felis sed diam pretium, ac adipiscing neque elementum. Quisque mi sapien, tristique et tristique a, euismod vitae lectus. Aenean sed cursus neque. Etiam molestie luctus egestas. Sed molestie iaculis consequat. Quisque euismod erat est, sed vehicula mi convallis viverra. Cum sociis natoque penatibus et magnis dis par- turient montes, nascetur ridiculus mus. Cras et odio iaculis, eleifend justo at, pretium

elit. Phasellus tincidunt sit amet nibh quis gravida. Quisque auctor aliquet justo et volutpat. Aenean a felis in tortor ultricies feugiat non et eros. Phasellus sapien justo, congue sit amet diam ut, euismod ornare arcu. Proin mollis justo ut nulla lacinia, at pellen- tesque sem fringilla. Donec vel feugiat odio, vel viverra sapien. Curabi- tur sed lacus enim.

Cras lectus ante, egestas quis blandit non, mattis quis metus. Viva- mus sapien dui, ornare nec massa eget, porttitor gravida tellus. Donec hendrerit posuere placerat. Nulla facilisi. Cras non orci at nisi rutrum sagittis ac congue dolor. Aliquam pellentesque libero sed libero male- suada laoreet. In venenatis dignissim sagittis. Mauris ligula elit, accum- san vel arcu et, dignissim rutrum dolor. Praesent vehicula lacus nunc, sed feugiat ligula semper et. Nunc id massa venenatis magna ornare faucibus quis sit amet arcu. Nam hendrerit mauris vitae urna pharetra lacinia. Aenean blandit nisi quis urna porta, id euismod magna condi- mentum.

Vivamus non arcu dolor. Integer in pulvinar orci, ac viverra purus. Pellentesque congue eleifend elit, et blandit arcu fringilla sit amet. Nulla blandit neque sed nisi tempus pretium. Aenean lacinia vel mau- ris sit amet tincidunt. Sed eget quam rhoncus, egestas mi id, facilisis sapien. Donec ut risus pulvinar enim imperdiet suscipit eget vitae lec- tus.

Vestibulum vehicula tortor ut metus tincidunt, sed feugiat do- lor tempus. Class aptent taciti sociosqu ad litora torquent per conu- bia nostra, per inceptos himenaeos. Aliquam ut erat ipsum. Donec blandit, sem quis adipiscing adipiscing, diam magna laoreet purus, eu gravida dui justo sit

amet risus. Nulla varius a libero eget ultricies. Etiam in pharetra augue. Pellentesque dui dui, egestas eu nisi vel, euismod feugiat ante. Sed consequat euismod nisl a sollicitudin. Nunc ac nisl cursus, mollis eros at, imperdiet ipsum. Fusce vehicula neque sit amet tortor bibendum interdum. Aliquam posuere elit eget tellus eleifend, eget molestie nunc scelerisque. Cras ut commodo enim, eget semper magna. Fusce ut pretium augue. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris in scelerisque quam.

Praesent laoreet ante et ornare mollis. Mauris libero leo, accum- san vitae leo sit amet, aliquam bibendum diam. Sed in odio eget tel- lus tempus tincidunt. Aenean et nunc quis erat hendrerit imperdiet nec vitae felis. Vivamus pharetra diam aliquam nisi tempus gravida. Vestibulum posuere auctor iaculis. Etiam convallis nunc ac accumsan vehicula. Etiam varius bibendum nisl non molestie. Phasellus facilisis rhoncus suscipit. Donec vel felis eget quam imperdiet eleifend.

In hac habitasse platea dictumst. Sed cursus ornare fermentum. Sed tempus sodales sapien id ultricies. Sed nec varius orci. Nullam consequat est ac velit posuere elementum. Proin nec lorem sceler- isque lacus vehicula facilisis sit amet at ligula. Pellentesque volutp feugiat suscipit. Integer lacinia est sed placerat scelerisque. In tem- por scelerisque faucibus. Cras eu turpis ut mi tincidunt facilisis a sed est. Vestibulum ultricies, tellus eget dignissim hendrerit, lacus ligula vulputate leo, pulvinar tincidunt dui ligula vitae dui. Ut sit amet con- vallis tellus. Vestibulum ante ipsum primis in faucibus orci luctus et ul- trices posuere cubilia Curae; Aliquam ligula lacus, luctus ut

tellus nec, ultricies imperdiet urna. Phasellus fermentum lacinia arcu, nec hen- drerit mauris euismod tempor. Nullam quis felis eleifend urna laoreet consectetur vitae quis orci.

In a sapien tincidunt, ultrices magna ultrices, eleifend urna. In sed nisi sapien. Donec eu tortor et erat dictum elementum sed nec justo. Etiam sit amet dignissim erat. Fusce elementum eget augue vel fer- mentum. Vestibulum bibendum nec nulla sit amet lobortis. Aenean ac hendrerit risus. Nullam placerat suscipit enim et fringilla. Donec luctus enim eget ornare ullamcorper. Vivamus dolor lectus, varius at mi la- cinia, auctor lobortis sem.

Maecenas aliquam sagittis augue in condimentum. Etiam ac enim quis mi mollis rutrum a eget metus. Sed ut orci libero. Donec pretium fringilla enim id posuere. Nulla rutrum pulvinar laoreet. Nam iaculis metus convallis massa molestie tristique. Etiam at dolor at massa euis- mod luctus vitae non sapien.

Nunc id libero et mi dictum aliquam a sed nunc. Sed congue me- tus nec leo pretium, eu pulvinar neque porttitor. Sed mauris lectus, ultrices ut ultricies in, facilisis nec lorem. Suspendisse ornare vel massa eu convallis. Phasellus consectetur nunc et tristique tempor. Integer lorem risus, sagittis et adipiscing eu, imperdiet id est. Proin in tristique turpis, lobortis imperdiet sem.

Fusce lacinia mi quis sollicitudin auctor. Aenean eget odio orci. Sed vulputate scelerisque diam, id mollis do- lor. Nullam id felis porta, malesuada est sed, feugiat sem. Suspendisse at neque sed est viverra consequat. Curabitur ac libero non lorem

lo- bortis vestibulum vel quis sapien. Vivamus quis aliquam libero. Morbi viverra eleifend nibh eu tristique.

Maecenas pulvinar erat quis libero molestie, id consequat est tem- por. Nunc elementum orci ullamcorper convallis laoreet. In ac justo commodo, ornare dolor sit amet, viverra leo. Nam dictum mauris tor- tor, eu commodo neque venenatis eget. Pellentesque habitant mor- bi tristique senectus et netus et malesuada fames ac turpis egestas. Praesent sodales est nec ipsum semper consectetur. Integer pulvinar molestie luctus. Vivamus quis aliquam lorem, in interdum nisi. Morbi iaculis non mi porta consectetur. Sed vulputate viverra felis, eu adipiscing arcu facilisis ut. Aliquam nec massa rhoncus, ullamcorper ante sed, commodo tellus. Nullam libero quam, volutpat in metus nec, condimentum malesuada est. Proin scelerisque dapibus purus, eu elementum elit consectetur in. Nulla blandit ligula non est rhoncus suscipit. Aenean pretium ligula vel enim hendrerit, ac interdum lectus vestibulum. Duis varius metus sit amet tortor feugiat mattis.

Aliquam eget faucibus orci, eu mollis dolor. Aenean sed libero porttitor, faucibus purus et, auctor nisl. Aliquam tincidunt nulla felis, ac tempus lorem suscipit vitae. Integer rhoncus molestie leo, id ul- lamcorper lorem ullamcorper convallis. Duis dapibus, ligula sit amet vehicula auctor, nisi ligula bibendum turpis, eget gravida lectus augue non arcu. Pellentesque lacus nisi, tristique dictum tincidunt id, adipiscing eu mi. Phasellus erat ligula, consectetur in aliquam rutrum, com- modo fringilla nibh.

Ut sit amet orci ligula. Ut tincidunt metus vitae libero

condimentum molestie. Proin tristique, nibh quis vulputate porttitor, magna mi vestibulum est, sit amet imperdiet nunc urna et elit. Fusce ultrices, libero a luctus bibendum, dolor nibh porttitor lig- ula, vitae condimentum lacus felis lobortis odio. Mauris rhoncus tur- pis dolor, eget varius risus sagittis eget. Mauris placerat convallis risus eget rhoncus. Duis ullamcorper orci orci, eu volutpat nunc porta non. Nunc eget adipiscing quam, eget vestibulum turpis. In aliquam ornare libero, in dignissim velit elementum eget.

Aliquam tempus, diam sit amet pharetra egestas, diam massa grav- ida nunc, eget aliquet leo libero in elit. Aenean eget neque et magna tincidunt ornare non id erat. Quisque vel dolor turpis. Pellentesque hendrerit dignissim auctor. Mauris et justo et odio dapibus blandit. Nunc vestibulum, nunc et commodo venenatis, orci justo fermentum eros, accumsan tincidunt velit augue sed elit. Aliquam erat volutpat. Sed leo velit, semper sed magna tristique, rhoncus vehicula velit. Aliquam suscipit pulvinar risus sollicitudin fermentum. Praesent quis tel- lus vitae justo accumsan fringilla.

ABOUT THE AUTHOR

Your book should have two "about the author" sections. The little blip on the back of your book cover is part of the sales copy: You add anything that makes you sound or appear more attractive to your target audience. You want to seem human. It's short: a sentence or two about where you live, what you enjoy, preferably something that ties in with your genre.

This section in the back of the book can be longer: you can give more history. This is for readers who have finished your book and want to get to know you better. It's also where you should remind them to leave a book review, or go sign up on your email list for a special bonus, or follow you on Facebook or Twitter. You can add links in here, and they'll work when you convert to ebook, (so link to your Twitter and Facebook as well if you want). But remember the links won't work on the print book, so you'll need to edit it to display the full link.

www.website.com

Acknowledgments

Aliquam tempus, diam sit amet pharetra egestas, diam massa grav- ida nunc, eget aliquet leo libero in elit. Aenean eget neque et magna tincidunt ornare non id erat. Quisque vel dolor turpis. Pellentesque hendrerit dignissim auctor. Mauris et justo et odio dapibus blandit. Nunc vestibulum, nunc et commodo venenatis, orci justo fermentum eros, accumsan tincidunt velit augue sed elit. Aliquam erat volutpat. Sed leo velit, semper sed magna tristique, rhoncus vehicula velit. Ali- quam suscipit pulvinar risus sollicitudin fermentum. Praesent quis tel- lus vitae justo accumsan fringilla.

Thanks for reading! Please add a short review on Amazon

and let me know what you thought!

It's nice to have a final call to action like this in the back to drive reviews. I would also add an optin offer, like "download the next book in the series for free!"/ "Get the quickstart guide to implementing the tools in this book!" A very simple 10 page PDF can be a great optin offer for non-fiction. For fiction you can offer future books, prequels or short stories. Also add an optin offer at the front of the book; people who just "Look Inside" can still see your offer and click the link, leading to more subscribers.

If this template was useful for you, why not share it with your writer friends? You could even write a blog post about your book formatting learning curve, and link to www.diybookformats.com so more authors can get the free templates and tutorials.

Thanks and good luck!
Derek Murphy