Linux OS

- **Opensource** – free to download, free share & change the code.
  - Linux OS
  - URL: https://www.kernel.org/
- **Freeware** - free to download, free share
  - UNIX
  - URL: https://www.oracle.com/solaris/solaris11/downloads/solaris-downloads.html

UNIX:

- Full form: **Un**iplexed **I**nformation and **C**omputing **S**ervice (UNI*CS*)
- 1969 by Dennis Ritchie & Kemp Thomson

Linux

- By Linus B. Torvalds. Released on 17th sept 1991.
- Derived from UNIX.

Multi-users:

- Virtual terminal / virtual consoles / pseudo terminal.
- Ctrl + atl + <f1, f2…f7> OR chvt <>
- 7 VC

Linux: - top 500 super computers

Secure:

- Permissions
- SELinux – security enhanced linux
- Firewall (netfilters, iptables/ip6tables, firewalld)

MBR

- Master Boot Record.
- 4 primary partitions
- HDD: 2TB
- 512 bytes
- 3 sub-divisions
  - 446 bytes → boot related info
  - 64 bytes → (16x4)
  - 2 bytes → magical bytes

GPT

- GUID partition table
- GUID = globally unique identifier
- 128 primary partitions
- HDD: 18 exabyte (approx. 18x10^6 GB)

GRUB

- Grand unified boot loader
- RHEL 6 → GRUB 0.97
- RHEL 7 → GRUB 2.0

1st process

- RHEL 6 → init process (PID 0)
- RHEL 7 → systemd (PID 0)

Runlevel

- Init 0 → runlevel0.target
- Init 1 → runlevel1.target
- Init 2 → runlevel2.target/multi-user.target
- Init 3 → runlevel3.target/multi-user.target
- Init 4 → runlevel4.target/unused till date
- Init 5 → runlevel5.target/graphical.target
- Init 6 → reboot.target

FHS

14 directories (standard), / → nameless root/root directory

- /bin → cmd binary for normal users
- /sbin → cmd binary for super/root users
    - #which useradd
        - /usr/sbin/useradd
    - #which ping
        - /usr/bin/ping
- /etc → configuration file
- /boot → boot related info/file
- /kernel → kernel related file.
- /mnt → temporary mounts
    - /mnt/abc
    - /mnt/dvd-rom
- /run → /mnt
- /dev → device related
    - /dev/sr0 → CD/DVD
    - /dev/sda or /dev/had → hard disk
- /tmp → temporary files
- /var → logs
- /lib or /libx64 → library
- /home → home directory for normal users
    - /home/Jeetu (~)
    - /home/hulk (~)
- /root → home directory for root user (~)
- /proc → running process related info.
    - CPU → /proc/cpuinfo
    - Memory → /proc/meminfo
- /usr → user installed packages/application

File systems in linux:

- Ext, Ext2
- Ext3, Ext4 → RHEL 6
- XFS → RHEL 7 (back compatible)

Kernel version: "uname -a"

User IDs:

- RHEL 6
  - Root → 0
  - System users → 1-499
  - Normal → 500+
- RHEL 7 (root → 0-999)
  - Root → 0
  - System users → 1-999
  - Normal → 1000+

Installation of RHEL:

1. CD-DVD or ISO.
   a. GUI
   b. Text-based (CLI)
2. Network boot
3. Kickstart

[Task-1: install a new RHEL VM in text mode (CLI).]

Installation destination mount point from 50 GB:

1. / = 35GB
2. /boot = 500mb
3. Swap (1.5xRAM) = 2xRAM = 4GB = virtual memory

File management:

- Create a file / Modify a file
  - Touch
    - Create blank file (1,2,3…n)
    - Update the time stamp of a file.
    - Ex:
      - #touch <file-name>
  - Cat
    - Create
      - #cat > cat1.txt
    - View
      - #cat cat1.txt
    - Append
      - cat >> cat1.txt
    - Overwriting
      - cat > cat1.txt
  - Vi / Vim
    - #vi <file-name>
      - Press ‘i’, to enter into insert mode.
      - …
      - Press “:wq!” To save & quit.
        - W = write / save
        - Q = quit / come out of vi editor
        - ! = forcefully perform the action.
    - #vim <file-name>
      - Press ‘i’, to enter into insert mode.
      - …
      - Press “:wq!” To save & quit.
        - W = write / save
        - Q = quit / come out of vi editor
      - ! = forcefully perform the action
  - Gedit
    - Like notepad in windows
    - #gedit <file-name>
  - Nano
    - #nano <file-name>
- Delete a file
  - #rm <file-name>
  - #rm -f <file-name> //w/o asking Y/N
  - #rm -f *.txt //everything that start with anything & ends with.txt
  - #rm -f f?.txt //delete files with 1 character after ‘f’ &ends with anything.
  - #rm -f f??.txt //delete files with 2 characters after ‘f’ &ends with anything.
- Copy a file
  - #cp <source> <destination>

Path in linux

- Relative path
  - Revolves around the pwd
  - Ex: #cp t1 d1/
- Absolute path
  - Work with from the beginning of the path till the end.
  - Ex: #cp /root/Desktop/t3 /root/Desktop/d1/

File mgmt – conti…

- Move a file
  - #mv <source> <destination>
- Rename
  - #mv <old-name> <new-name>
- Hide
  - #mv <old-name> .<old-name>
- Unhide
  - #mv .<old-name> <old-name>
- Single '.'
  - Current directory
- Double ".."
  - Parent directory or 1 step above.

Directories

- Create
  - #mkdir <dir-name>
- Change the location
  - #cd
- delete
  - #rmdir <dir-name> //works with empty directories.
  - #rm -rf <dir-name> //works with empty & non-empty directories.
    - r= recursive
    - f= forcefully (without prompt)

Permissions

For file/directory

- read (r) → 4
- write (w) → 2
- execute (x) → 1
- all perms → 7

to view the permissions

#ls -l or #ll

```
[root@svr Desktop]# ll
total 0
drwxr-xr-x. 2 root root 15 Apr 14 12:13 lti
-rw-r--r--. 1 root root  0 Apr 14 11:58 t2
-rw-r--r--. 1 root root  0 Apr 14 11:58 t3
[root@svr Desktop]#
```

Column1:

- type of file
- and permission allocation to that file/directory

(d/-) → d= directory, - → file

(rwxr-xr-x) → permission to owner, group and others

Owner = rwx, means =>u

- owner can read the f/d
- owner can write into f/d
- owner can execute the f/d

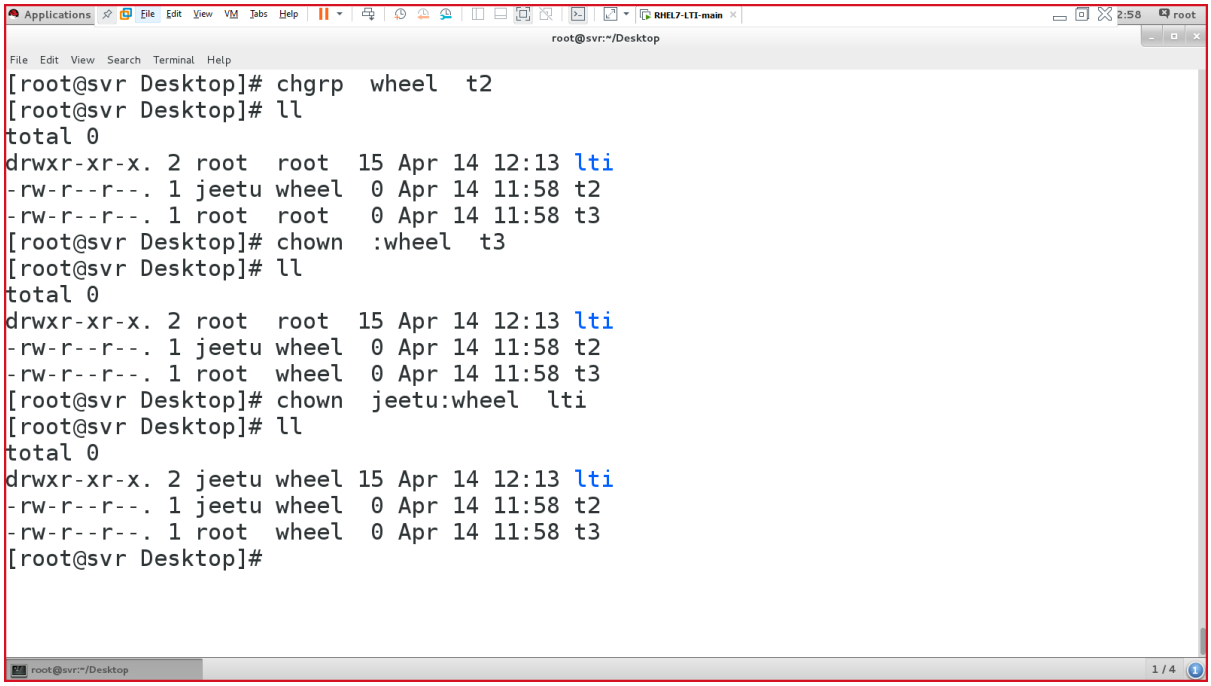
group = r-x, means =>g

- group can read the content of f/d
- group can execute the contents of f/d
- but group cannot write anything in the f/d

other = r-x, means =>o

- other can read the content of f/d
- other can execute the contents of f/d
- but other cannot write anything in the f/d
- Column2: number of links occupied in the memory block.
- Column3: owner of the file/directory → to change the owner: chown
- Column4: group owner of the file/directory → chown, chgrp
- Column5: file/directory size in bytes.
- Column6: time stamp of file/directory.
- Column7: name of the file/directory.

To change the file/directory owner & group owner:



Changing the permissions of a f/d:

1. Changing via numerical method.
    a. #chmod <number> <file-name>
    b. 000
    c. 111
    d. 222
    e. …
    f. 777
2. Changing via symbolic/alphabetical method.
    a. #chmod <user><op><perms> <file-name>
    b. #chmod ugo=x <file-name>
        i. = means assigning new perms
        ii. + means save existing perms + add new perms
        iii. - means save existing perms + subs new perms
    c. Ex: #chmod u=rwx,g=rx,o=x t2

User management:

Important files in user mgmt:

- /etc/passwd → holds user related information
- /etc/shadow → holds users passwd detailed information.

Tasks:

1. create a user
2. modify a user
3. delete a user
4. sudo user → EDITOR=vim visudo

/etc/passwd: (7 columns)

1. login name
2. x →referenced password → /etc/shadow
3. user id.
4. Group id of the user.
5. Description/full name /GECOS → "chfn"
6. Default home directory of a user.
7. Default SHELL.

/etc/shadow: (9 columns)

1. Login name
2. True encrypted password (MD5 or SHA512)
    a. $6 = SHA512 algo
    b. $1 = MD5 algo
3. Last password change (since 1st Jan 1970 in number of days)
4. Minimum passwd age (0)
5. Maximum passwd age (99999 in days) ~273yrs → 30 days
6. Warning days → 7 days
7. Empty → inactive
8. Empty → account expiry
9. Empty → not used till date / for future purpose.

To check the passwd algo:

```
# authconfig --test | grep hashing
```

Creating a user:

a. #useradd <option> <username>
b. Ex: #useradd -u 1024 -s /bin/csh rocky
c. Areas where new user's entry is gonna create:
    a. /etc/passwd
    b. /etc/shadow
    c. /home/<user-name-dir>
    d. /var/mail/

Modify an existing user:

a. #usermod
b. Ex: #usermod -c "Ittalian Sttalion" rocky

Delete a user:

a. Partial deletion (used for deleting users' credentials)
    a. #userdel <username>
        i. /etc/passwd
        ii. /etc/shadow
b. Full deletion (used for deleting user completely from the Linux system)
    a. #userdel -r <username>
        i. /etc/passwd
        ii. /etc/shadow
        iii. /home/<user-name-dir>
        iv. /var/mail/

To search a specific keyword in a file:

- #grep rocky /etc/passwd OR
- #cat /etc/passwd | grep rocky

To search a file or directory in linux:

- find / -name content.txt // searching a file
- find / -type d -name lti // searching a directory

Groups in Linux:

- File: /etc/group
- Consists of 4 columns
  - Group name
  - X → referenced passed → /etc/gshadow
  - Group ID
  - Group members

Sudo users:

- User with limited admin rights.

Sudo user = normal user + limited admin permissions.

- Root user => #
- Normal user => $
- Sudo user => $

File → /etc/sudoers

Commands → #visudo or EDITOR=vim visudo

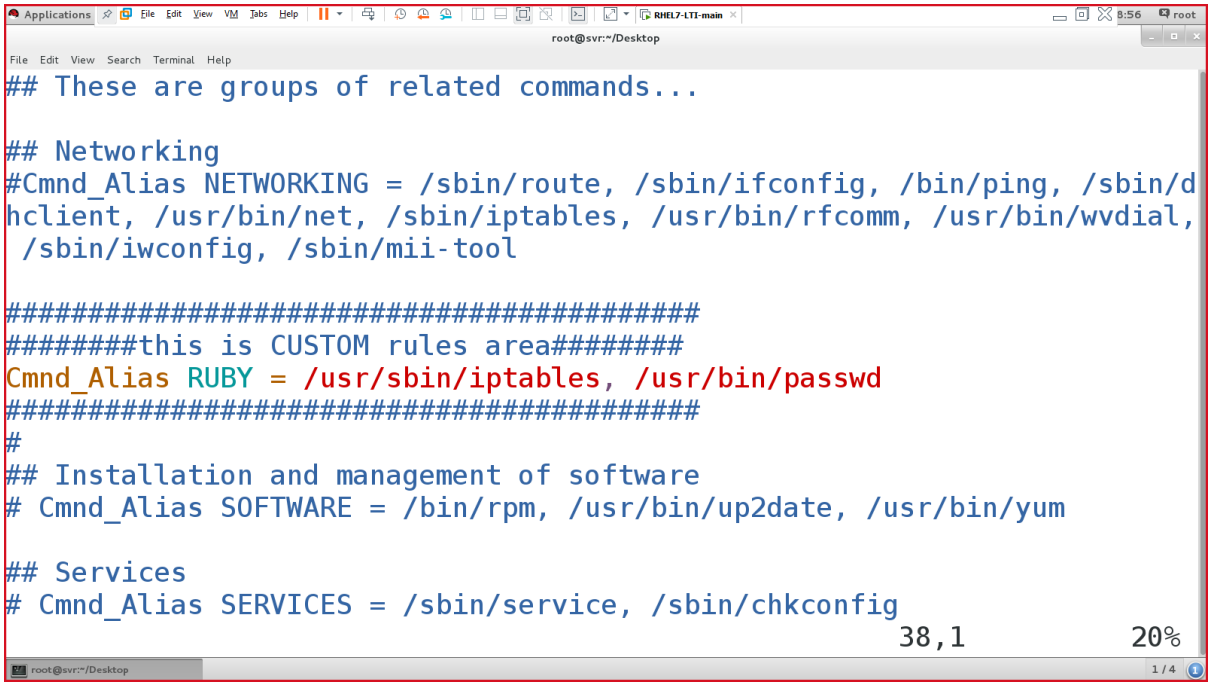
root ALL=(ALL) ALL

- col1: user name
- col2: machine=terminal
- col3: commands given to the user

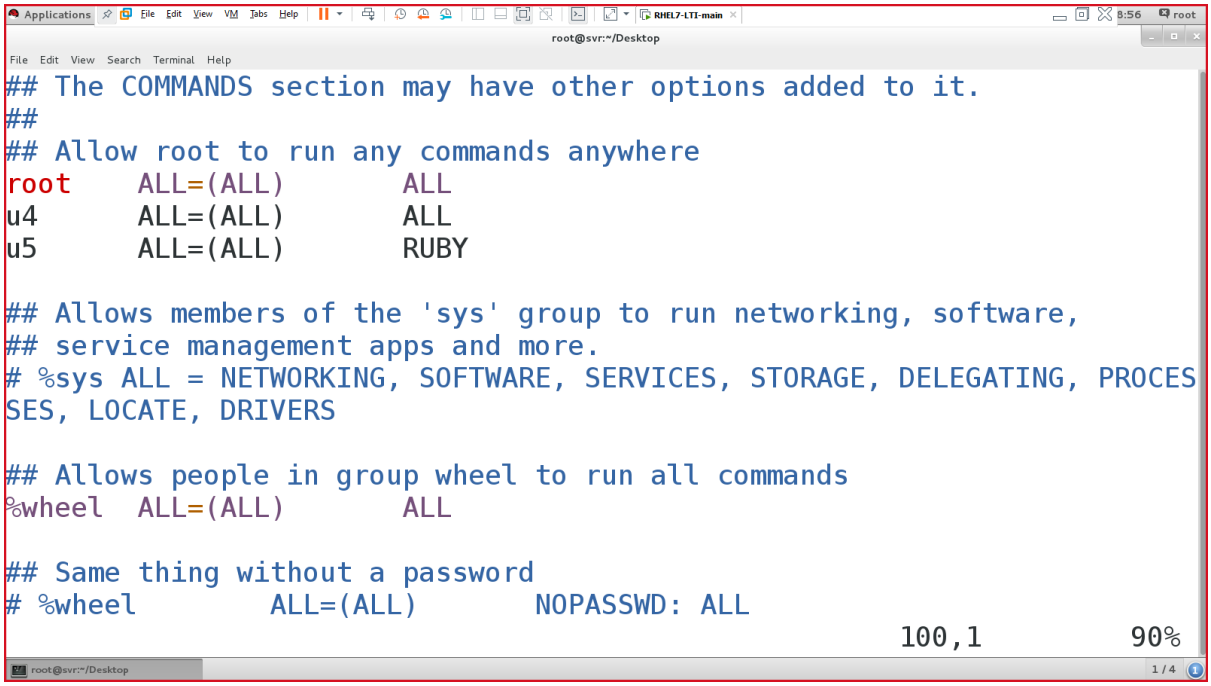
sudo tasks:

a. create a new sudo user with ALL permission/cmd.
b. create a new sudo user with PREDIFINED cmd.
c. Create a new sudo user with CUSTOM cmd.

Defining the commands:



Allocating the commands to the users.



Packages management:

- Installing packages/software on linux machine.
- Local repository (repository = location where all packages are present).
- Internet based repository.

Repo:

- RedHat repo = local (DVD), RedHat satellite server (paid).
- Centos repo = local (DVD), Centos mirror server (free).

Packages managers – responsible for package mgmt (install, remove, update…)

- RPM
  - Redhat package manager
  - **Default method** to manage a package in RHEL/Centos/Fedora…
  - Extension is .rpm
- YUM
  - Yellowdog update manager OR Yellowdog update modifier.
  - A server that WE need to install.
  - To install YUM server, there are 2 methods.
    - **Full** – 8/9 steps, independent of DVD, permanent.
    - **Shortcut** – 1 step, temporary (dependent on DVD)
  - Extension is .rpm
- APT
  - Advanced packaging tool
  - Ubuntu server.
  - Free of cost, no charges to download or install any packages.
  - Extension is .rpm, .deb, .dpkg
- DPKG
  - Debian-based package manager.
  - Extension as .deb, .dpkg
  - Kali, backtrack.
  - Free of cost

Naming convention of a package in Linux: Name-major-version-architecture.rpm

Win → XShell_5.0-Build-1019.exe

- XShell = name of the software
- _5.0 = versions
- Build-1019 = release/built
- .exe = extension

Linux → zip-3.0-10.el7.x86_64.rpm

- Zip = name of the package
- '3.0' = major release/version
- 10.el7 = minor release
- X86_64 = architecture of the package
- .rpm = extension.

Architecture of any linux package:

a. 32bit = i686
b. 64bit = x86_64
c. No architecture = noarch = 32bit and 64bit

RPM – Redhat package manager.

- Install
  - #rpm -ivh <package-full-name>.rpm
    - I = install
    - V = verbose
    - H = hashing (##################################)
  - Ex: #rpm -ivh /run/media/root/RHEL-7.0\ Server.x86_64/Packages/dhcp-4.2.5-27.el7.x86_64.rpm OR
  - #rpm -ivh dhcp-4.2.5-27.el7.x86_64.rpm
- Query
  - rpm -q <package-name> → list single installed package.
  - rpm -qa → list all installed packages.
- Uninstall/remove
  - #rpm -e <package-name>
- Update
  - #rpm -Uvh <package-name>
    - U = means, update

Problems of RPM:

- *Full name* → dhcp-4.2.5-27.el7.x86_64.rpm
- *Location* → /run/media/root/RHEL-7.0\ Server.x86_64/Packages/dhcp-4.2.5-27.el7.x86_64.rpm (absolute path or relative path)
- *Dependency*
  - #rpm -ivh system-config-kickstart-2.9.2-4.el7.noarch.rpm → dependency error while installing
  - #rpm -ivh --nodeps system-config-kickstart-2.9.2-4.el7.noarch.rpm → installation done w/o dependency.

YUM – yellowdog update manager

- Install a YUM server
- It consists of 8/9 steps.

Steps to install YUM server:

Step 1: check the firewall & SELinux

- #systemctl status firewalld → check FW status
- #systemctl stop firewalld → stop FW

SELinux:

- setenforce 0 → will make SELinux permissive
- getenforce → output will permissive.

Step 2: check the list of packages.

- rpm -q createrepo deltarpm python-deltarpm vsftpd

Step 3: create a local repo directory

- mkdir /var/ftp/pub/yumserver

Step 4: copy all the packages from DVD to the local repo (created in step 3)

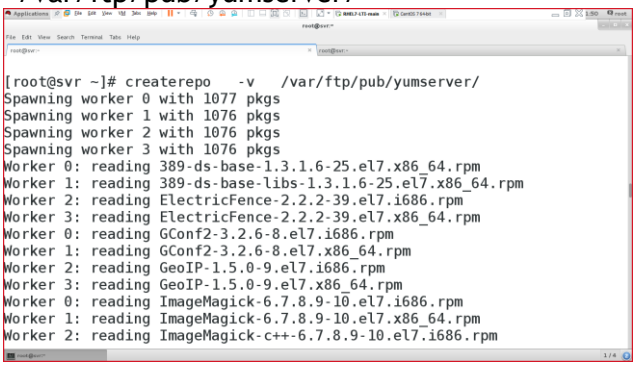
- cp -var /run/media/root/RHEL-7.0\ Server.x86_64/Packages/* /var/ftp/pub/yumserver/
- v= verbose
- a= all (hidden & visible)
- r= recursively

Step 5: create yum/client answer file.

- To go to /etc/yum.repos.d directory
- List the contents of the directory, if any file named package… or Redhat…. Then delete those files.
- Create a new file 'yum.repo' And write below lines:
- [yumserver]
- name="Joker SVR"
- baseurl=ftp://192.168.10.10/pub/yumserver
- gpgcheck=0
- enabled=1

Step 6: create index for the packages.

- #createrepo -v /var/ftp/pub/yumserver/

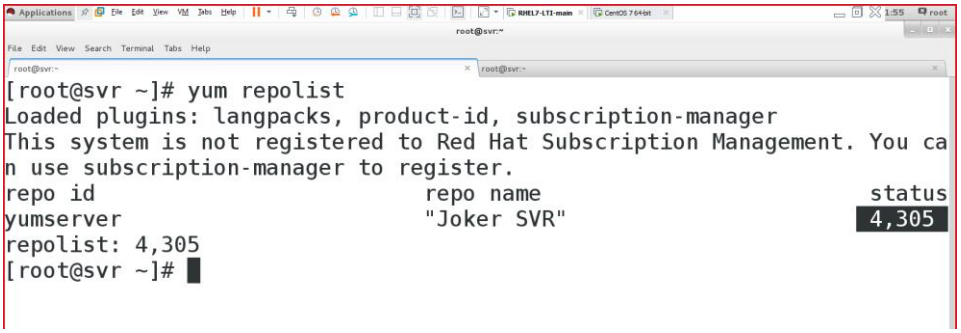


Step 7: enable, start & check the status of vsftpd service.

- #systemctl enable vsftpd → ensure to start vsftpd service at next boot time.
- #systemctl start vsftpd → start the vsftpd service for NOW.
- #systemctl status vsftp → running status of vsftpd

Step 8: clean, update & verify.

- #yum clean all
- #yum update all
- #yum repolist



END OF FULL YUM SERVER

- Install
  - #yum install -y <package-name>
  - #yum install -y dhcp system-config-kickstart…
- Query
  - #yum info <package-name>
  - #yum info finger
- Uninstall/remove
  - #yum remove <package-name>
- Upgrade/update
  - #yum update <package-name>

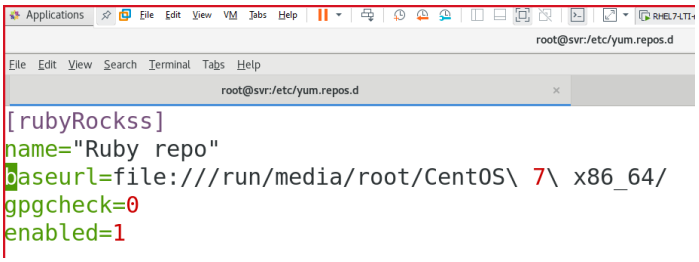
For shortcut YUM method

Step 1: DVD must be present. Go to /etc/yum.repo.d

Step 2: remove/move all the file.

Step 3: create a new client answer file as yum.repo & write/paste below code:

- [rubyRockss]
- name="Ruby repo"
- baseurl=file:///run/media/root/CentOS\ 7\ x86_64/
- gpgcheck=0
- enabled=1
    - :wq!



Save & quit

Step 4: run cmd → #yum repolist

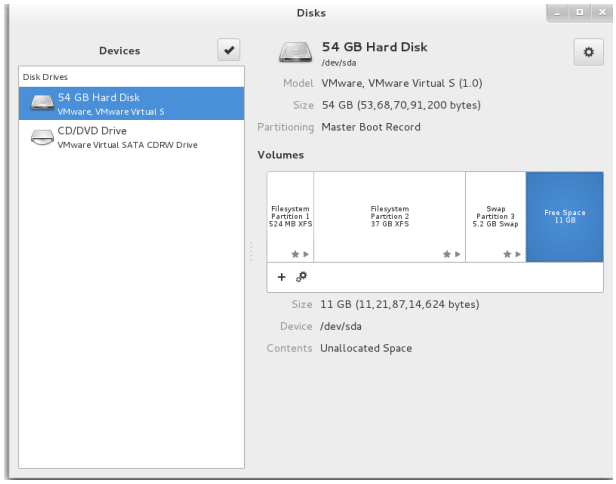
```
[root@svr yum.repos.d]# yum repolist
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
repo id                                repo name                                status
rubyRockss                             "Ruby repo"                              3,831
repolist: 3,831
[root@svr yum.repos.d]#
```

Partitioning:

a. Simple partitioning → 1 blank storage disk
b. Advanced partitioning → logical volume manager (LVM) → 3 blank storage disks.

Simple partitioning

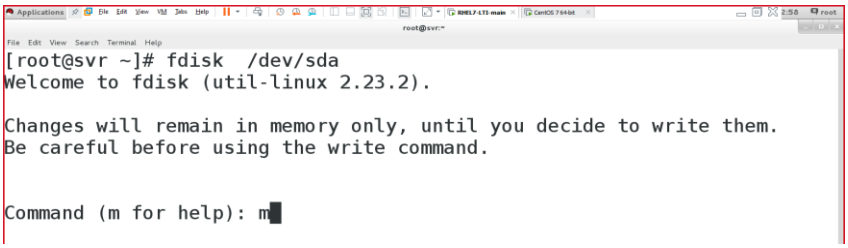
Before:



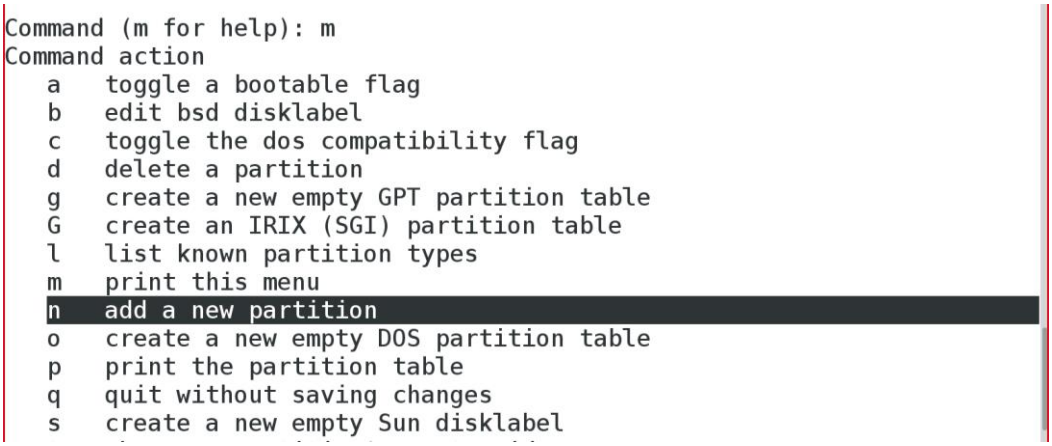
a. Create a partition for the disk → fdisk
b. Create a file system for the partition (created in step 1) → mkfs
c. Create a mount point to access the disk. → mkdir
d. Edit /etc/fstab file → critical.
e. Mount & verify

Step1: Create a partition for the disk

"fdisk" works on hard disk → /dev/sda



Listing help commands:



Create primary partition:

```
Command (m for help): n
Partition type:
   p   primary (3 primary, 0 extended, 1 free)
   e   extended
Select (default e): p
Selected partition 4
First sector (82946048-104857599, default 82946048):
Using default value 82946048
Last sector, +sectors or +size{K,M,G} (82946048-104857599, default 10485
7599): +10G
Partition 4 of type Linux and of size 10 GiB is set

Command (m for help):
```

Again press ‘m’ for help & select ‘w’.

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

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or
resource busy.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
[root@svr ~]#
```

Either reboot your system or execute the command:

- #partprobe /dev/sda

Step2: Create a file system for the partition

“mkfs” works on partition, you created in step 1.

```
[root@svr ~]# mkfs.
mkfs.btrfs   mkfs.ext3    mkfs.gfs2    mkfs.vfat
mkfs.cramfs  mkfs.ext4    mkfs.minix   mkfs.xfs
mkfs.ext2    mkfs.fat     mkfs.msdos
[root@svr ~]# mkfs.xfs /dev/sda4
meta-data=/dev/sda4              isize=256    agcount=4, agsize=655360 b
lks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=0
data     =                       bsize=4096   blocks=2621440, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=0
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
[root@svr ~]#
```

Step 3: Create a mount point to access the disk

- #mkdir /warmachine

Step 4: edit /etc/fstab

```
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=8d0e66cc-bca0-4f93-b3f7-24293f7acb58 /                       xfs     defaults        1 1
UUID=81cde27d-893c-4600-add6-b5a737969567 /boot                   xfs     defaults        1 2
UUID=a321d2c8-c116-4385-8e65-b2e42fe52c84 swap                    swap    defaults        0 0
```

1 2 3 4 5

- Col1: what to mount? → /dev/sda4
- Col2: where to mount → /warmachine
- Col3: file system of the hard disk → xfs
- Col4: how to mount/permission → defaults
- Col5: priority at boot time → 2 2

```
#
UUID=8d0e66cc-bca0-4f93-b3f7-24293f7acb58 /                       xfs     defaults        1 1
UUID=81cde27d-893c-4600-add6-b5a737969567 /boot                   xfs     defaults        1 2
UUID=a321d2c8-c116-4385-8e65-b2e42fe52c84 swap                    swap    defaults        0 0
/dev/sda4       /warmachine     xfs     defaults        2 2
```

Save & quit

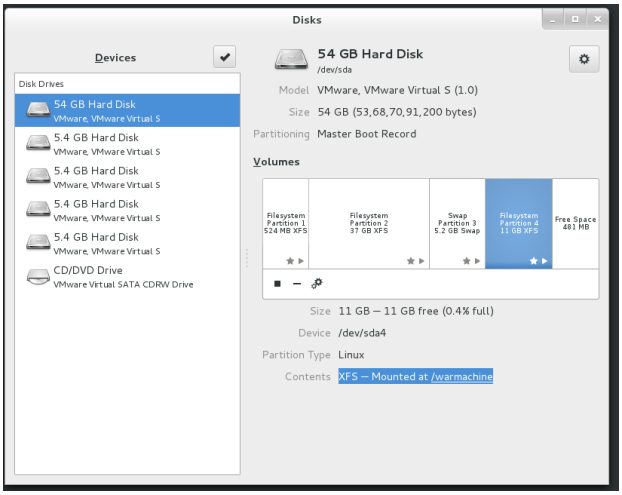
Step 5: Mount & verify

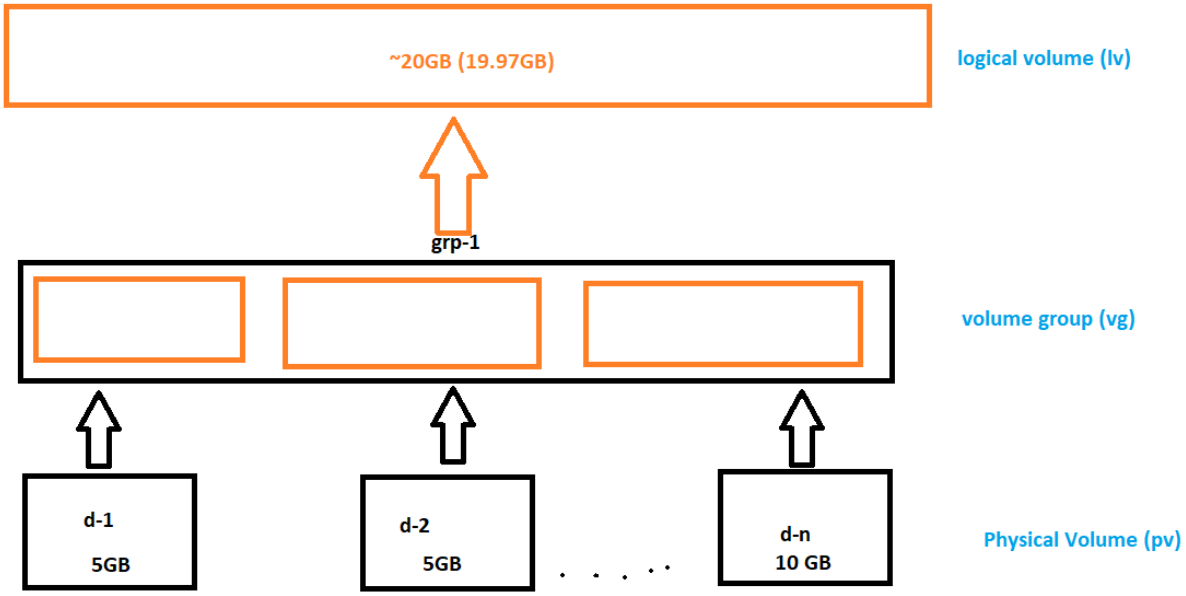
```
[root@svr ~]# mount -a
[root@svr ~]# mount | grep /warmachine
/dev/sda4 on /warmachine type xfs (rw,relatime,seclabel,attr2,inode64,no
quota)
[root@svr ~]#
```

Verify:

```
[root@svr ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2        35G  7.7G   27G  23% /
devtmpfs        1.9G     0  1.9G   0% /dev
tmpfs           1.9G  140K  1.9G   1% /dev/shm
tmpfs           1.9G  8.9M  1.9G   1% /run
tmpfs           1.9G     0  1.9G   0% /sys/fs/cgroup
/dev/sda1       497M  119M  379M  24% /boot
/dev/sda4        10G   33M   10G   1% /warmachine
[root@svr ~]#
```

After the partitioning:





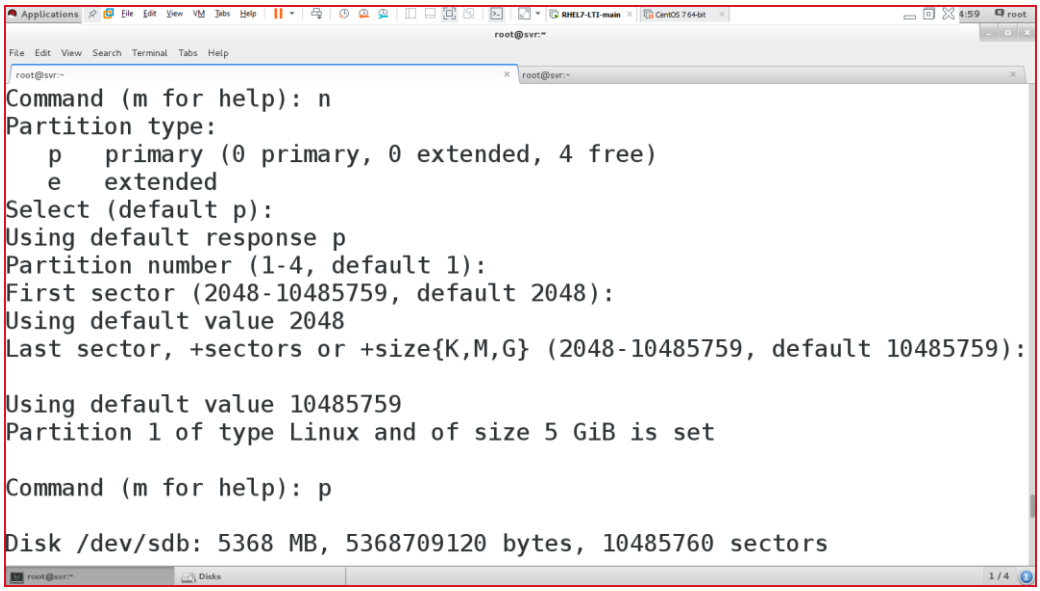
Steps to perform LVM.

- Create a partition for the disk → fdisk
- pv → pvcreate, pvdisplay/pvs, pvremove
- vg → vgcreate, vgdisplay/vgs, vgremove
- lv → lvcreate, lvdisplay/lvs, lvremove
- Create a file system for the partition (created in step 1) → mkfs
- Create a mount point to access the disk. → mkdir
- Edit /etc/fstab file → critical.
- Mount & verify

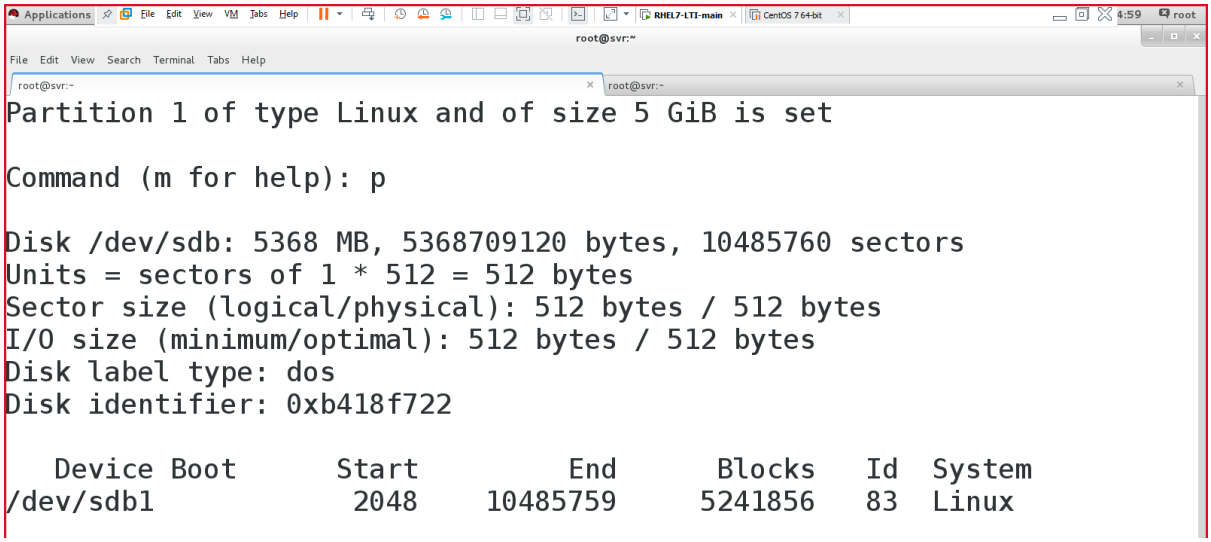
Disks for LVM → /dev/sdb & /dev/sdc

Step 1: Create a partition for the disk (use FDISK utility)

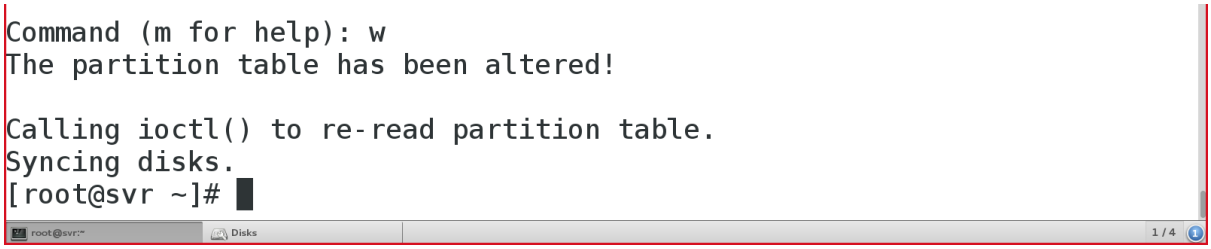
#fdisk /dev/sdb



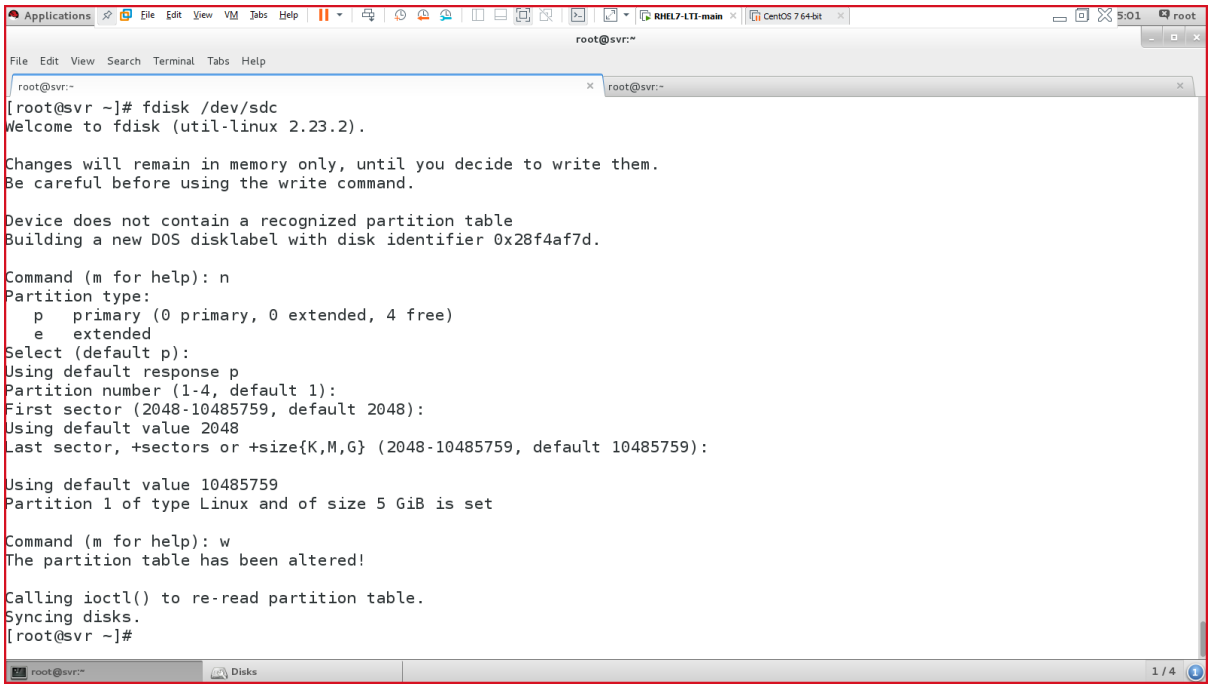
```
Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p):
Using default response p
Partition number (1-4, default 1):
First sector (2048-10485759, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-10485759, default 10485759):

Using default value 10485759
Partition 1 of type Linux and of size 5 GiB is set

Command (m for help): p

Disk /dev/sdb: 5368 MB, 5368709120 bytes, 10485760 sectors
```



Then select ‘w’ to save & quit the utility

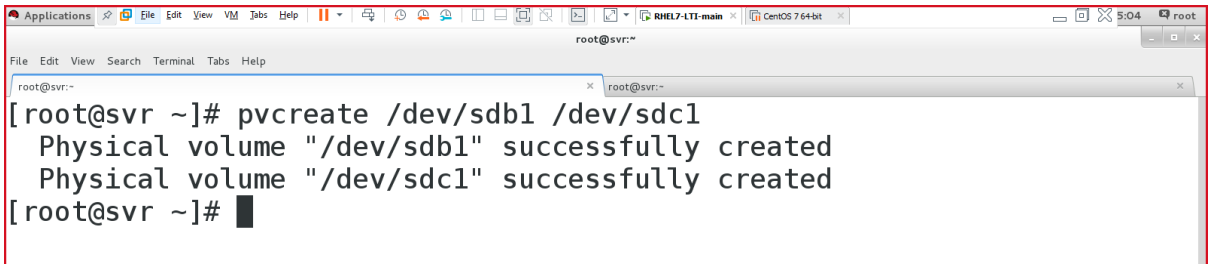


#fdisk /dev/sdc

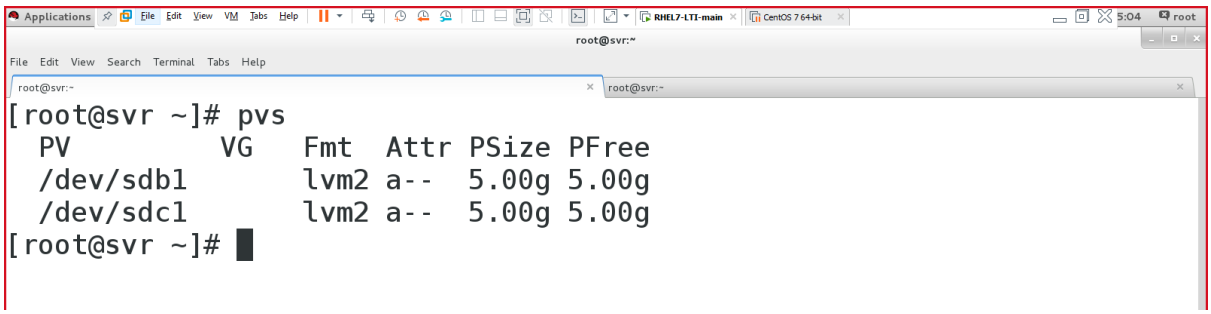


Step 2: create a physical volume (PV)

#pvcreate <disk-1> <disk-2>…



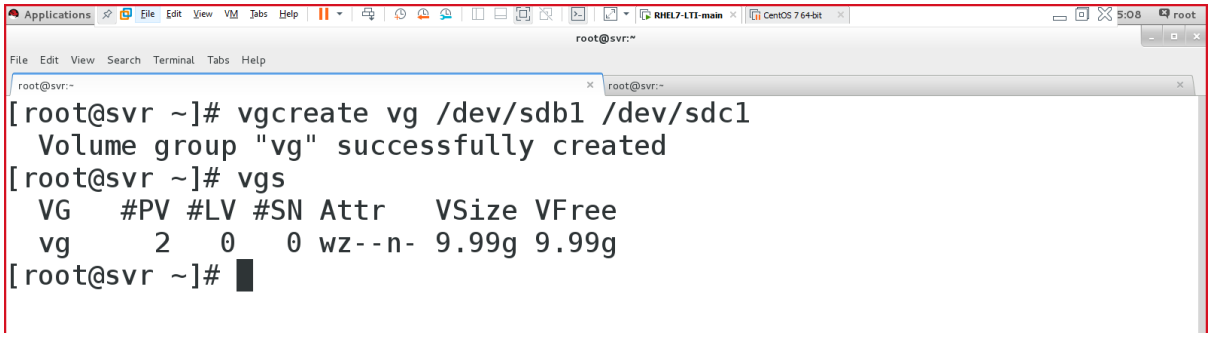
Display the pv:



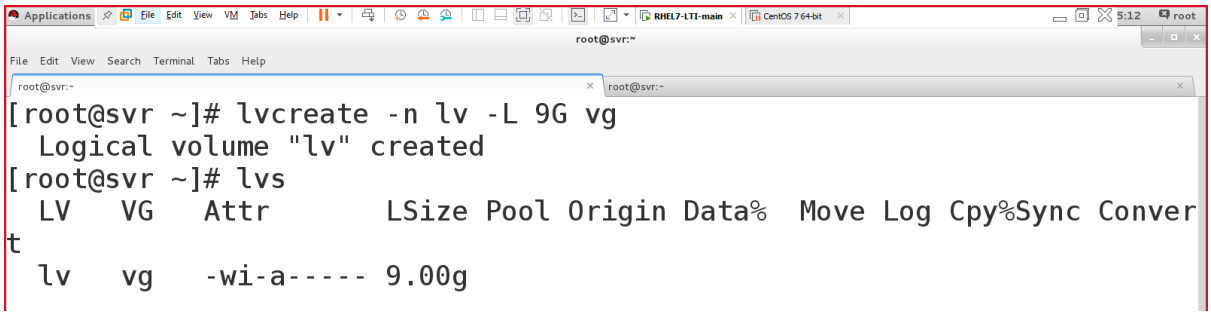
Step 3: create a volume group (vg)

#vgcreate <name-of-the-vg> <pv1> <pv2>…<pvn>

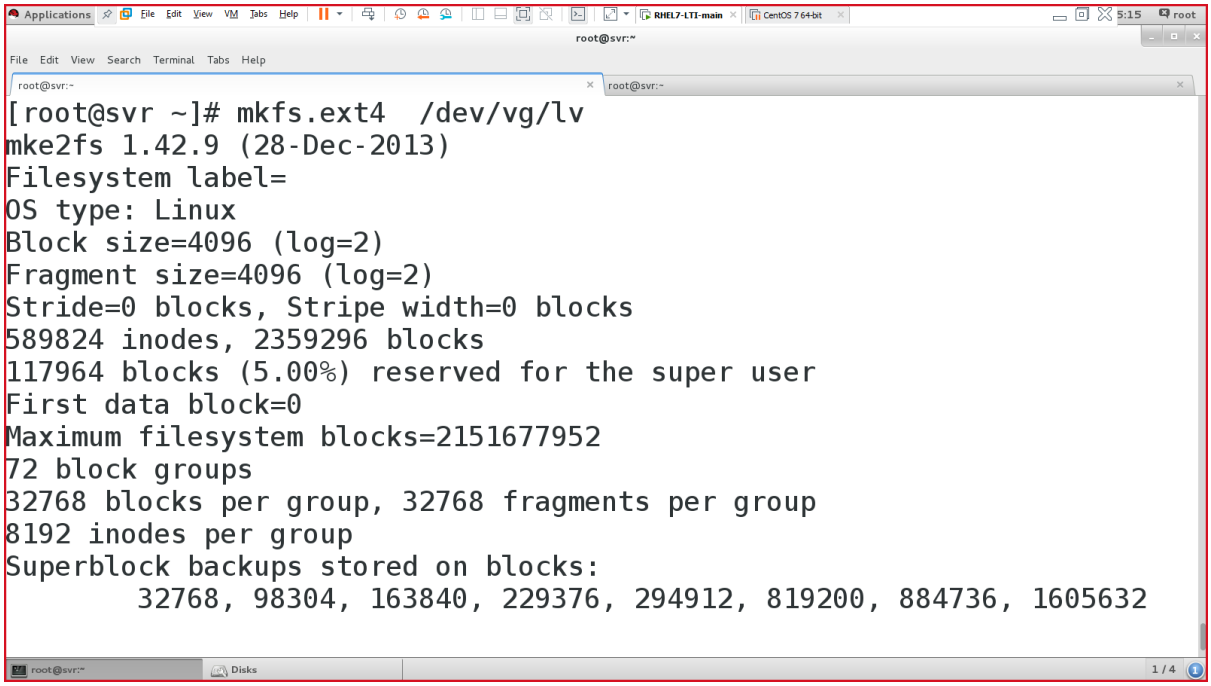
#vgcreate vg /dev/sdb1 /dev/sdc1



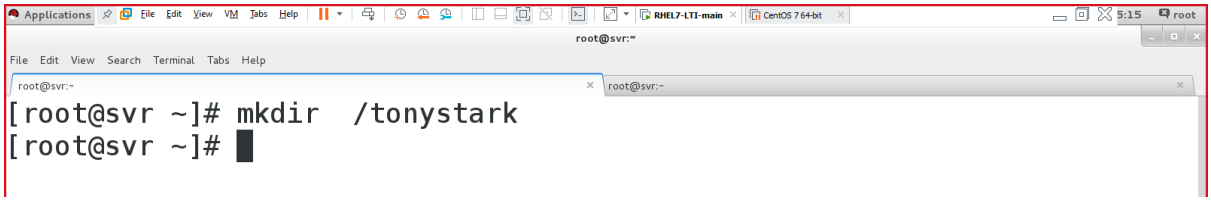
Step 4: create a logical volume (lv)



Step 5: create a file system (.ext4)



Step 6: create a mount point



Step 7: mount the hard drive

a. Permanent
    a. Edit /etc/fstab

```
 defaults        0 0
/dev/sda4       /warmachine     xfs     defaults        2 2
/dev/vg/lv      /tonystark      ext4    defaults        2 2
```

Save & quit

NOTE: EITHER PERFORM PERMANENT MOUNT OR TEMPORARY MOUNT (1 at a time)

b. Temporary
    a. Use mount command.

```
[root@svr ~]# mkdir /joker
[root@svr ~]# mount   /dev/vg/lv  /joker
[root@svr ~]# df -h
Filesystem          Size  Used Avail Use% Mounted on
/dev/sda2            35G  7.7G   27G  23% /
devtmpfs            1.9G     0  1.9G   0% /dev
tmpfs               1.9G  472K  1.9G   1% /dev/shm
tmpfs               1.9G  8.9M  1.9G   1% /run
tmpfs               1.9G     0  1.9G   0% /sys/fs/cgroup
/dev/sda1           497M  119M  379M  24% /boot
/dev/sda4            10G   33M   10G   1% /warmachine
/dev/mapper/vg-lv  8.8G   37M  8.3G   1% /joker
[root@svr ~]#
```

Step 8: mount & verify

```
[root@svr ~]# mount -a
[root@svr ~]#
```

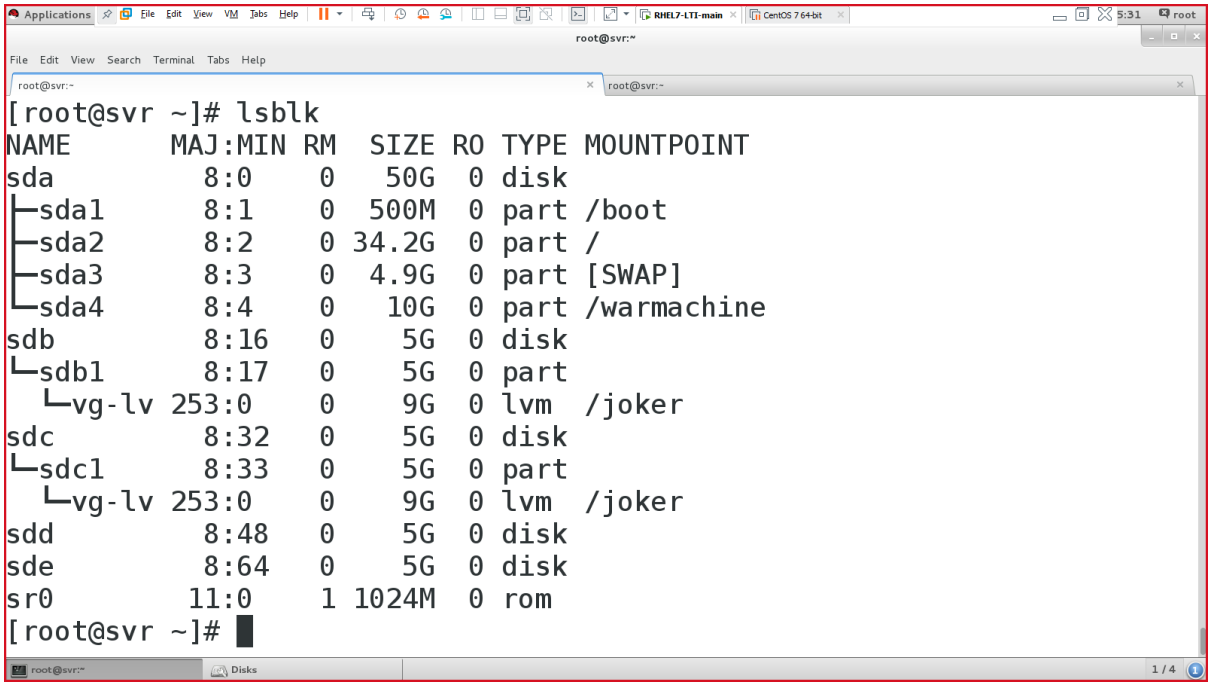
And verify:

```
[root@svr ~]# mount | grep /tony
/dev/mapper/vg-lv on /tonystark type ext4 (rw,relatime,seclabel,data=ord
ered)
[root@svr ~]#
```

Verify by saving the file in the mounted drive

```
[root@svr ~]# touch /tonystark/mach42
[root@svr ~]# ls /tonystark/
lost+found  mach42
[root@svr ~]#
```

Verify using “df -h”.


Applications File Edit View VM Tabs Help
RHEL7-LTI-main
CentOS 7 64-bit
5:31 root
root@svr:~
File Edit View Search Terminal Tabs Help
root@svr:~
root@svr:~
[root@svr ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 50G 0 disk
├─sda1 8:1 0 500M 0 part /boot
├─sda2 8:2 0 34.2G 0 part /
├─sda3 8:3 0 4.9G 0 part [SWAP]
└─sda4 8:4 0 10G 0 part /warmachine
sdb 8:16 0 5G 0 disk
└─sdb1 8:17 0 5G 0 part
└─vg-lv 253:0 0 9G 0 lvm /joker
sdc 8:32 0 5G 0 disk
└─sdc1 8:33 0 5G 0 part
└─vg-lv 253:0 0 9G 0 lvm /joker
sdd 8:48 0 5G 0 disk
sde 8:64 0 5G 0 disk
sr0 11:0 1 1024M 0 rom
[root@svr ~]#
root@svr:~
Disks
1 / 4

RAID

- redundant array of independent disks or
- redundant array of inexpensive disks.

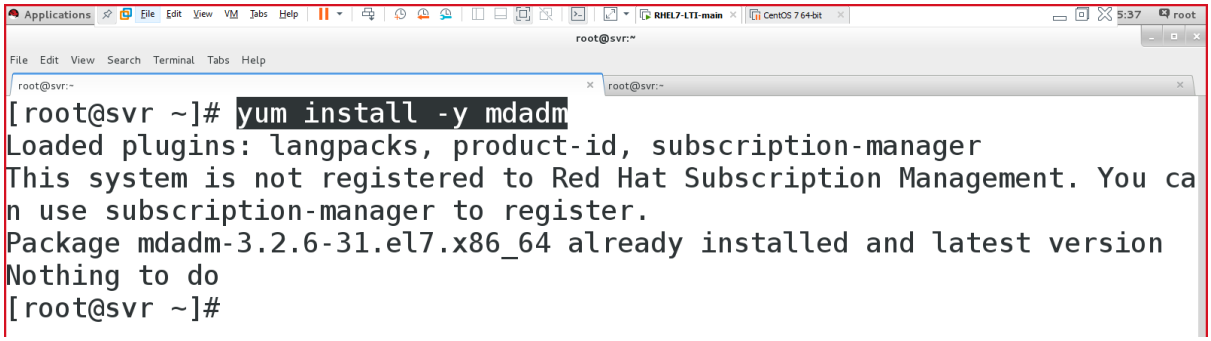
Types of RAID:

- Raid 0 – striping
- Raid 1 – mirroring
- Raid 5 – striping with parity
- Raid 6 - striping with dual/double parity
- Raid 10 – combining striping with mirroring.

Disks to be in the RAID 0 - /dev/sdd & /dev/sde

Package required for RAID

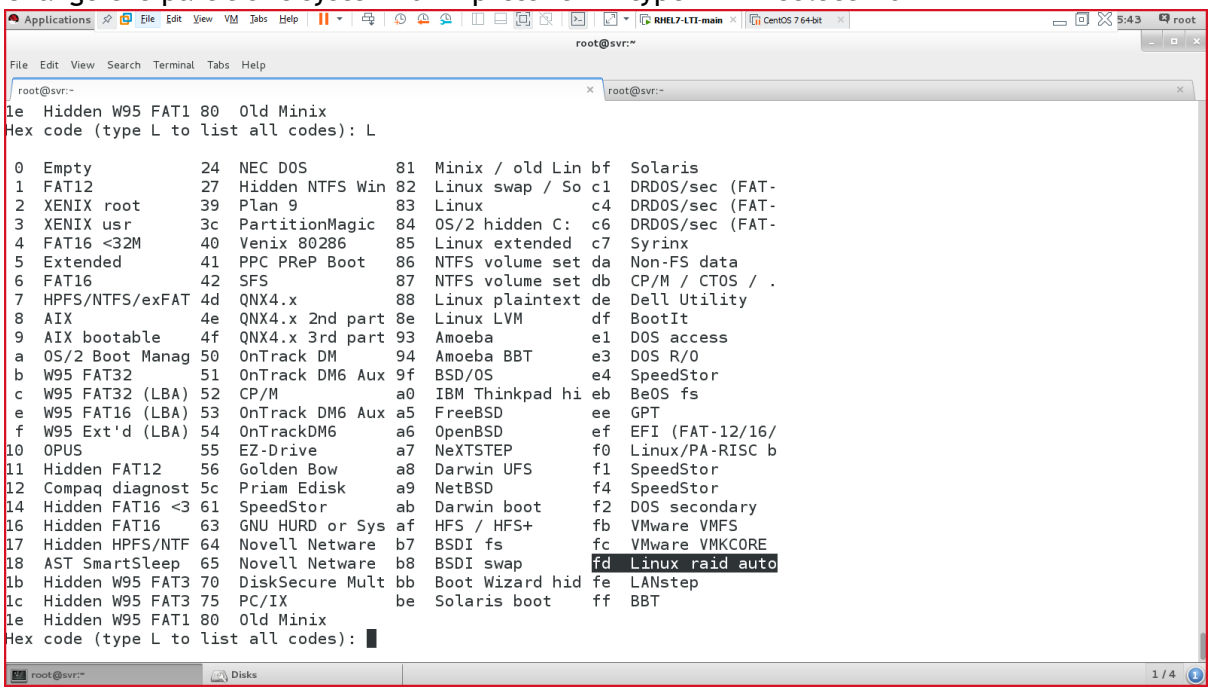
#yum install -y mdadm



Format the disk in the required raid format.

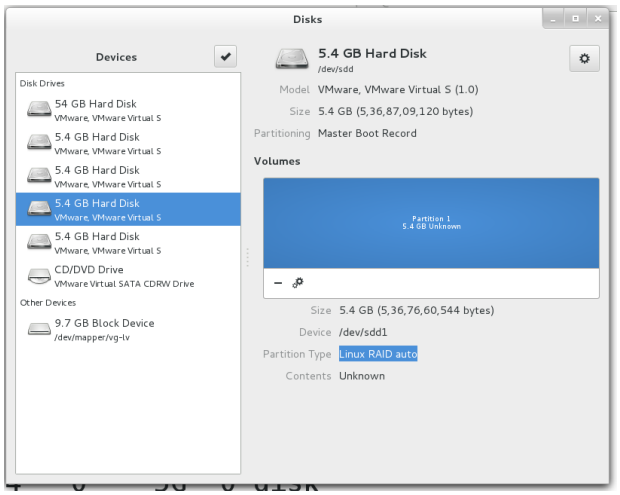
#fdisk /dev/sdd

- Format the disk in the default setting (done earlier in standard partitioning)
- N = new, p = primary, 1, select 1st sector & last sector.
- Change the partition's system id → press 't' → type 'L' → select 'fd'



```
Hex code (type L to list all codes): fd
Changed type of partition 'Linux' to 'Linux raid autodetect'

Command (m for help):
```

Verify:



Perform the same for the 2nd disk in RAID:

#fdisk /dev/sde

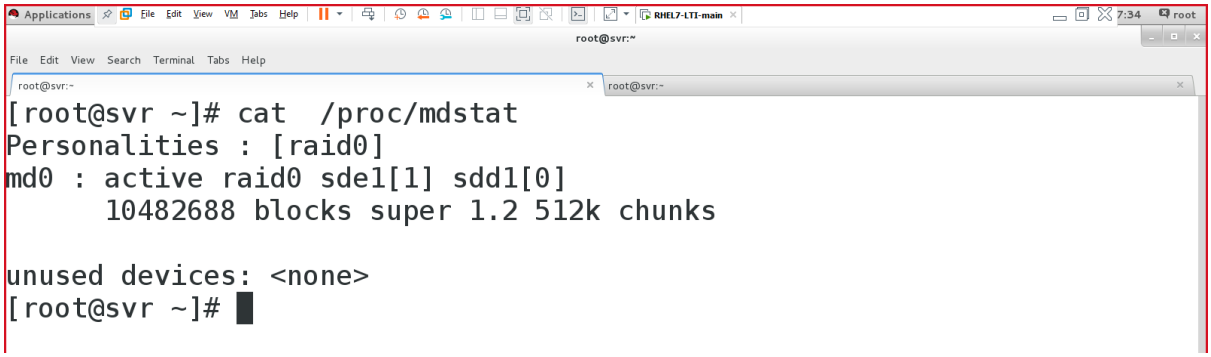
Execute the below cmd to examine (-E) the disks:

```
[root@svr ~]# mdadm -E /dev/sdd1 /dev/sde1
mdadm: No md superblock detected on /dev/sdd1.
mdadm: No md superblock detected on /dev/sde1.
[root@svr ~]#
```

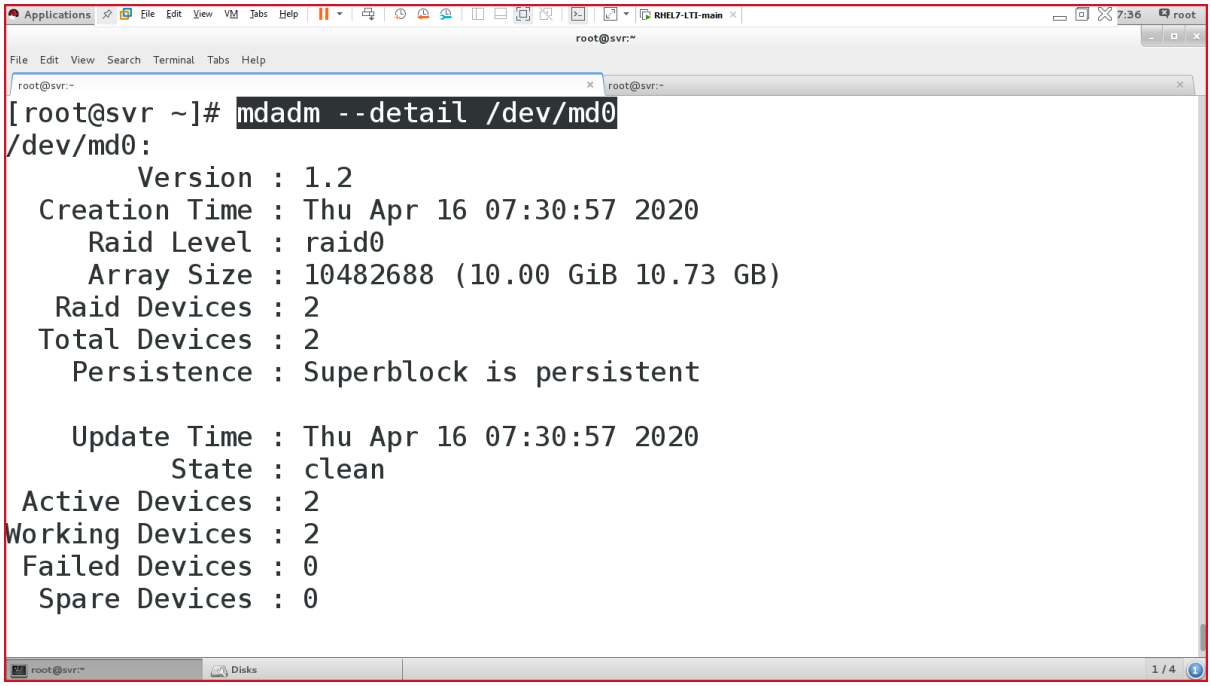
Create raid 0 on /dev/md0

```
[root@svr ~]# mdadm --create /dev/md0 --level=stripe --raid-device=2 /dev/sdd1 /dev/sde1
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
[root@svr ~]# mdadm -E /dev/sdd1 /dev/sde1
/dev/sdd1:
          Magic : a92b4efc
        Version : 1.2
    Feature Map : 0x0
     Array UUID : 8c818db8:1cc3508b:aa22769e:2303fabf
           Name : svr.alpha.corp:0  (local to host svr.alpha.corp)
  Creation Time : Thu Apr 16 07:30:57 2020
     Raid Level : raid0
   Raid Devices : 2

 Avail Dev Size : 10483696 (5.00 GiB 5.37 GB)
    Data Offset : 16 sectors
   Super Offset : 8 sectors
          State : clean
    Device UUID : ac721758:17d43c04:c3808029:3aafd972

    Update Time : Thu Apr 16 07:30:57 2020
       Checksum : 3879271b - correct
         Events : 0
```

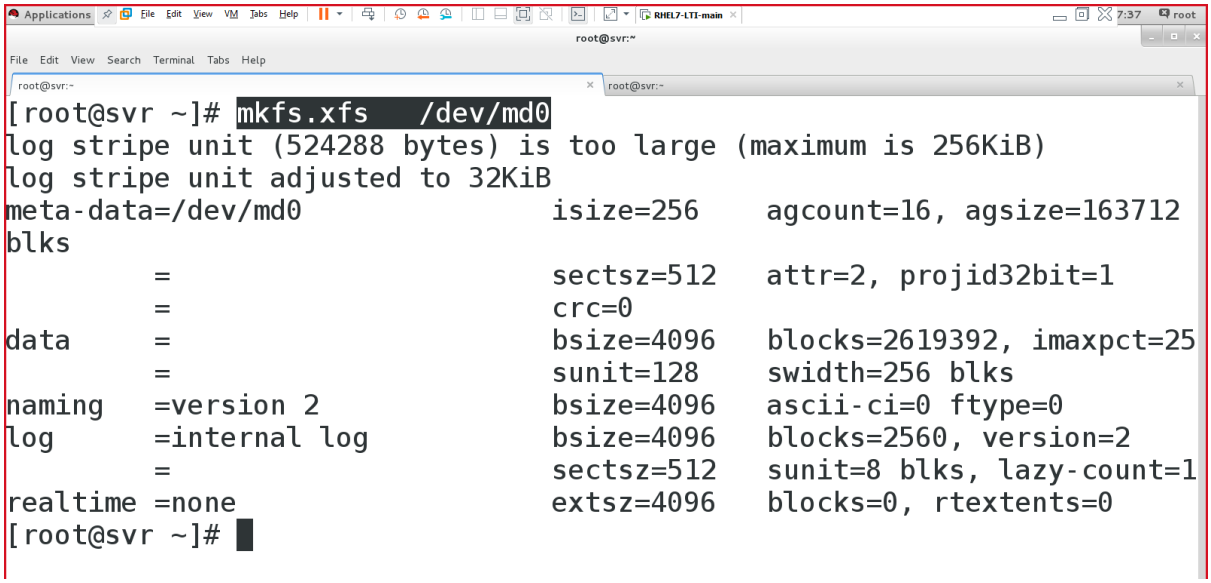
Check the raid status using below command:



Verify raid status:



Create a filesystem for RAID (you can use any FS, i.e, xfs, ext3, ext4):



Steps for RAID 0:

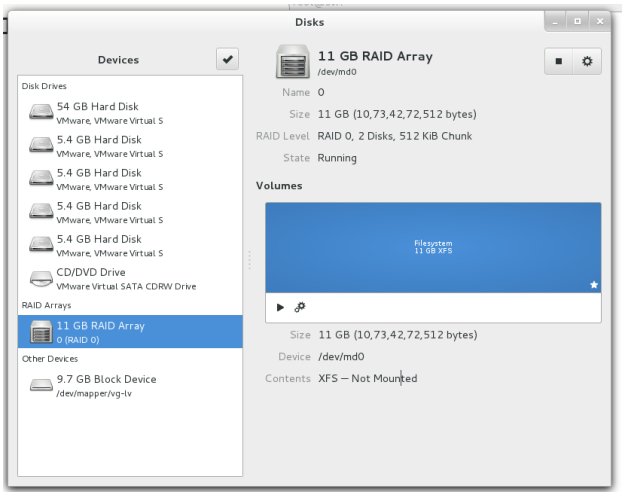
- Attach at least 2 disks, and reboot.
- Use FDISK utility to create a new partition & then convert it into the raid format (fd).
- Examine the disk → #mdadm -E /dev/sdd1 /dev/sde1
- Create a raid 0:
  #mdadm --create=/dev/md0 --level=stripe --raid-device=2 /dev/sdd1 /dev/sde1
- Examine the disk again → #mdadm -E /dev/sdd1 /dev/sde1
- Check the details:
  - #cat /proc/mdstat
  - #mdadm --detail /dev/md0
- Create a file system:
  - #mkfs.xfs /dev/md0
- Edit fstab file:
  - Either you can use /dev/md0
    /dev/md0 /raid0 xfs defaults 1 1
  - Or use UUID (extracted from [blkid /dev/md0])
    UUID=70acac47-840d-434a-ac6d-baa1a6738f60 /raid0 xfs defaults 1 1

```
#
# /etc/fstab
# Created by anaconda on Tue Apr 14 10:32:47 2020
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=8d0e66cc-bca0-4f93-b3f7-24293f7acb58 /                       xfs     defaults        1 1
UUID=81cde27d-893c-4600-add6-b5a737969567 /boot                   xfs     defaults        1 2
UUID=a321d2c8-c116-4385-8e65-b2e42fe52c84 swap                    swap    defaults        0 0
/dev/sda4       /warmachine     xfs     defaults        2 2
#/dev/vg/lv     /tonystark      ext4    defaults        2 2
#/dev/md0       /raid0  xfs     defaults        1 1
UUID=70acac47-840d-434a-ac6d-baa1a6738f60       /raid0  xfs     defaults        1 1
```

- Mount & verify
  - #mount -a
  - #mount
- Verify: → df -h

```
[root@svr ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2        35G  7.7G   27G  23% /
devtmpfs        1.9G     0  1.9G   0% /dev
tmpfs           1.9G  524K  1.9G   1% /dev/shm
tmpfs           1.9G  8.9M  1.9G   1% /run
tmpfs           1.9G     0  1.9G   0% /sys/fs/cgroup
/dev/sda1       497M  119M  379M  24% /boot
/dev/sda4        10G   33M   10G   1% /warmachine
/dev/mapper/vg-lv  8.8G   37M  8.3G   1% /joker
/dev/md0         10G   33M   10G   1% /raid0
[root@svr ~]#
```

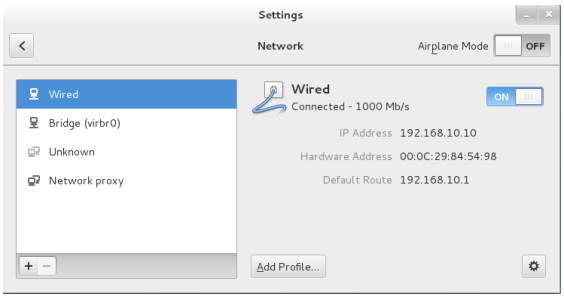
Verify using GUI.



RAID CONFIGURATION OVER

Networking:

- #ifconfig
- #ifconfig | grep inet
- **NMCLI** – Network Manager Command Line Interface (RHEL 5, 6, 7, 8)
  - nmcli connection show
  - nmcli device status
  - nmcli device show
  - nmcli device show eno16777736
- **NMTUI** – Network Manager Text User Interface (RHEL 7, RHEL 6 = setup)
  - #nmtui
  - Then change the required settings (IP address)
  - Then restart the network service
    - #systemctl restart network
- **File**
  - /etc/sysconfig/network-scripts/ifcfg-eno16777736
  - Search using → #find /etc -name ifcfg*
  - Edit the file (change the IP address & gateway)
  - Then restart the network service again.
- GUI



Working with services:

RHEL 6

- #service <service-name> <action>
- Ex:
  - #service network restart
  - #service network start
  - #service network stop

RHEL 7

- #systemctl <action> <service-name>
- Ex:
  - #systemctl restart network → restart the service
  - #systemctl start network → start the service
  - #systemctl stop network → stop the service
  - #systemctl disable network → disable the service, after the reboot
  - #systemctl enable network → enable the service, after the reboot
- Service
  - Network
  - Dhcp
  - Dns
  - Vsftpd
  - …

Firewall:

RHEL 6

- Netfilter / iptables (IPv4 traffic) / ip6tables (IPv6 traffic)

To list firewall rules (RHEL 6 method)

- #iptables -L //L = list the rules
- #iptables -F //F = flush/remove the rules.
- iptables -I INPUT -s 10.10.10.10 -j ACCEPT //accepting any traffic from specific IP address.
    - I = insert
    - 's' = source
    - 'j' = jump
    - ACCEPT = allow, REJECT = deny, DROP = discard.

Blocking complete IP address range for SMTP protocol.

- iptables -I INPUT -s 192.168.11.0/24 -p tcp --destination-port 25 -j DROP
- iptables-save > /etc/iptables/rules.v4 → save iptables rules in RHEL 6.

RHEL 7

- firewall-cmd
- To list all the zones in RHEL 7 system:
    - #firewall-cmd --get-zones
- To list default zone:
    - #firewall-cmd --list-all
- Change the default 'public' zone to 'external'
    - #firewall-cmd --set-default-zone=external
    - To verify:
        - #firewall-cmd --list-all
    - To add http/https service
        - #firewall-cmd --add-service=http –permanent
        - #firewall-cmd --add-service=https –permanent
        - #firewall-cmd --add-service=ftp –permanent
        - #firewall-cmd --reload
    - To add port numbers in the firewall
        - #firewall-cmd --add-port=21/tcp --permanent
        - #firewall-cmd --add-port=53/tcp --permanent
        - #firewall-cmd --add-port=53/udp --permanent
        - #firewall-cmd --reload
    - To list all services and ports allowed in firewall
        - #firewall-cmd --list-all
    - To add specific IP address or address range in firewall
        - #firewall-cmd --add-source=10.10.10.10/8 --permanent → IP addr
        - #firewall-cmd --add-source=192.168.10.0/24 --permanent → range
        - #firewall-cmd --reload → reload
        - #firewall-cmd --list-all → verify/listing.

NTP:

- NTP = Network Time Protocol
- Time syncing between systems in the network.
- Port number = 123/udp

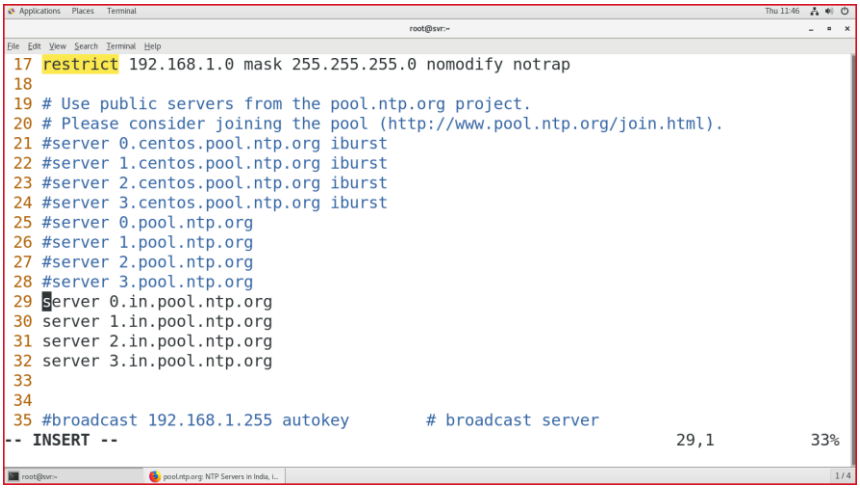
Lab setup:

- 1 NTP server
  - Centos OS 7
  - Internet is allowed.
- 1 NTP client
  - Windows based machine
  - Internet is allowed.
- Packages required: ntp
- Service for NTP server: ntpd
- Config file for NTP: /etc/ntp.conf

Steps:

- Install the NTP package.
  - #yum install ntp -y
- Edit the configuration file of NTP
  - #vim /etc/ntp.conf

Add below entries from line 25:
server 0.in.pool.ntp.org
server 1.in.pool.ntp.org
server 2.in.pool.ntp.org
server 3.in.pool.ntp.org



On line 17, allow your IP address range:

```
13 restrict 127.0.0.1
14 restrict ::1
15
16 # Hosts on local network are less restricted.
17 restrict 192.168.6.0 mask 255.255.255.0 nomodify notrap
18
```

At the last line, add file to capture NTP logs:

#logfile /var/log/ntp.log

```
 67 disable monitor
 68
 69 logfile /var/log/ntp.log
-- INSERT --                                                    69,25         Bot
```

Save & quit.

Add the service in the firewall

- #firewall-cmd --add-service=ntp --permanent
- #firewall-cmd --add-port=123/udp --permanent
- #firewall-cmd --reload

Enable, start & status for NTP

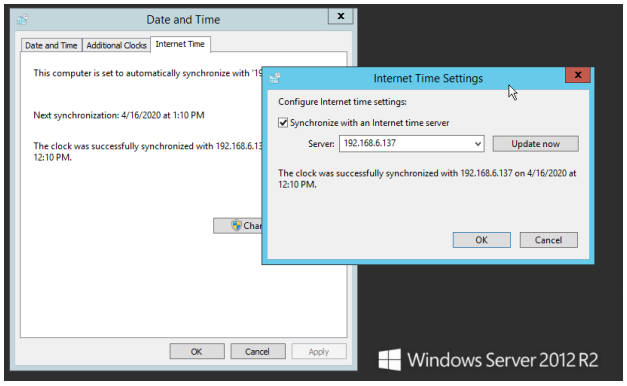
- #systemctl enable ntpd
- #systemctl start ntpd
- #systemctl status ntpd

#############NTP SERVER CONFIG OVER################

<u>On NTP-Client machine</u>,

- Go to the date & time settings.
- And add the NTP server IP address.

In windows system



On linux system,

"-u" = update from NTP server.

#############################NTP SERVER CONFIG DONE#########################

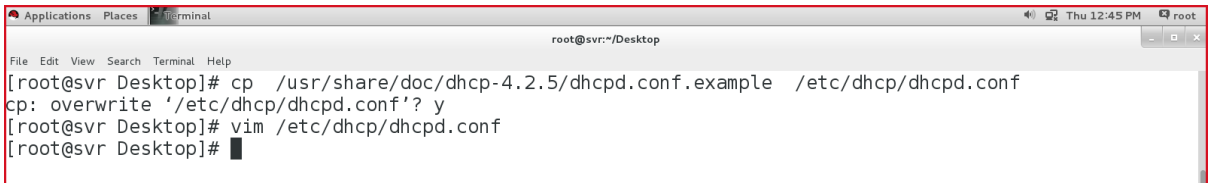
DHCP server:

- Systems required: 2
- System 1: DHCP server, 192.168.10.10/24 (static IP), no internet required.
- System 2: DHCP client, (have to provide IP dynamically), no internet required.
- Packages: dhcp
- Port number: 67, 68
- Config file for DHCP: /etc/dhcp/dhcpd.conf

On DHCP server:

Install DHCP package:

- #yum install -y dhcp
- Copy the sample DHCP file to the DHCP config directory.



Perform the changes in the "dhcpd.conf" file:

On line 7, change the domain name to your working domain:
→ option domain-name "alpha.corp";

On line 27, make the below changes:

```
subnet 192.168.10.0 netmask 255.255.255.0 {
        option routers 192.168.10.254;
        option subnet-mask      255.255.255.0;
        option domain-name-servers      192.168.10.10;
        option time-offset      -18000;
        range   192.168.10.100 192.168.10.200;
}
```

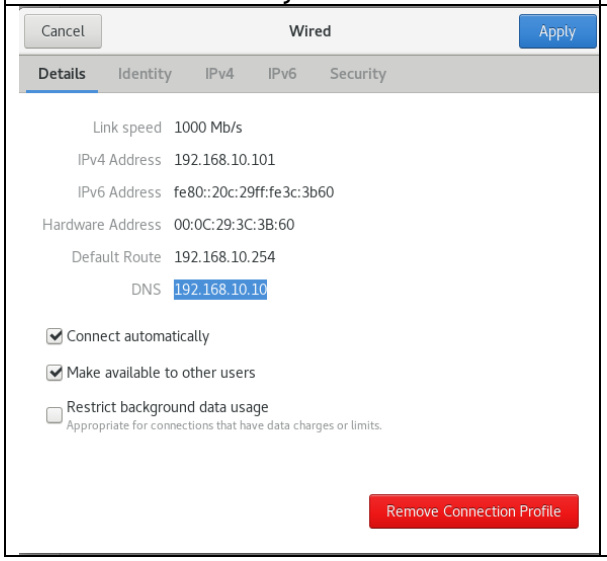
:wq!
Save & quit:
#systemctl enable dhcpd
#systemctl start dhcpd
#systemctl status dhcpd → Active: (running)

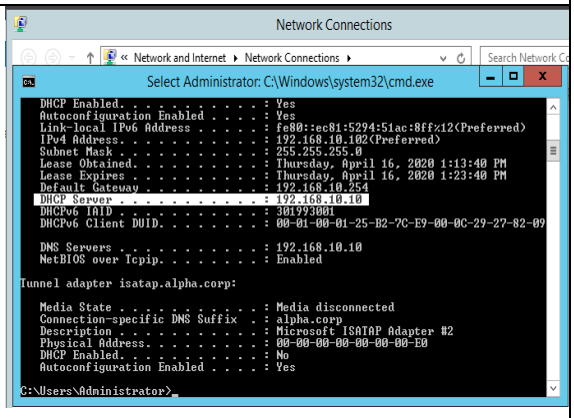
On DHCP server – config file

```
#

# option definitions common to all supported networks...
option domain-name "alpha.corp";
option domain-name-servers svr.alpha.corp;
default-lease-time 600;
max-lease-time 7200;

# Use this to enble / disable dynamic dns updates globally.
#ddns-update-style none;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
#authoritative;

# Use this to send dhcp log messages to a different log file (you also
# have to hack syslog.conf to complete the redirection).
log-facility local7;

# No service will be given on this subnet, but declaring it helps the
# DHCP server to understand the network topology.

subnet 192.168.10.0 netmask 255.255.255.0 {
        option routers  192.168.10.254;
        option subnet-mask      255.255.255.0;
        option domain-name-servers      192.168.10.10;
        option  time-offset     -18000;
        range   192.168.10.100 192.168.10.200;
}

# This is a very basic subnet declaration.
```

root@svr:~

On linux machine, change the IP settings to dynamic	On windows machine:





##################DHCP CONFIG OVER#########################

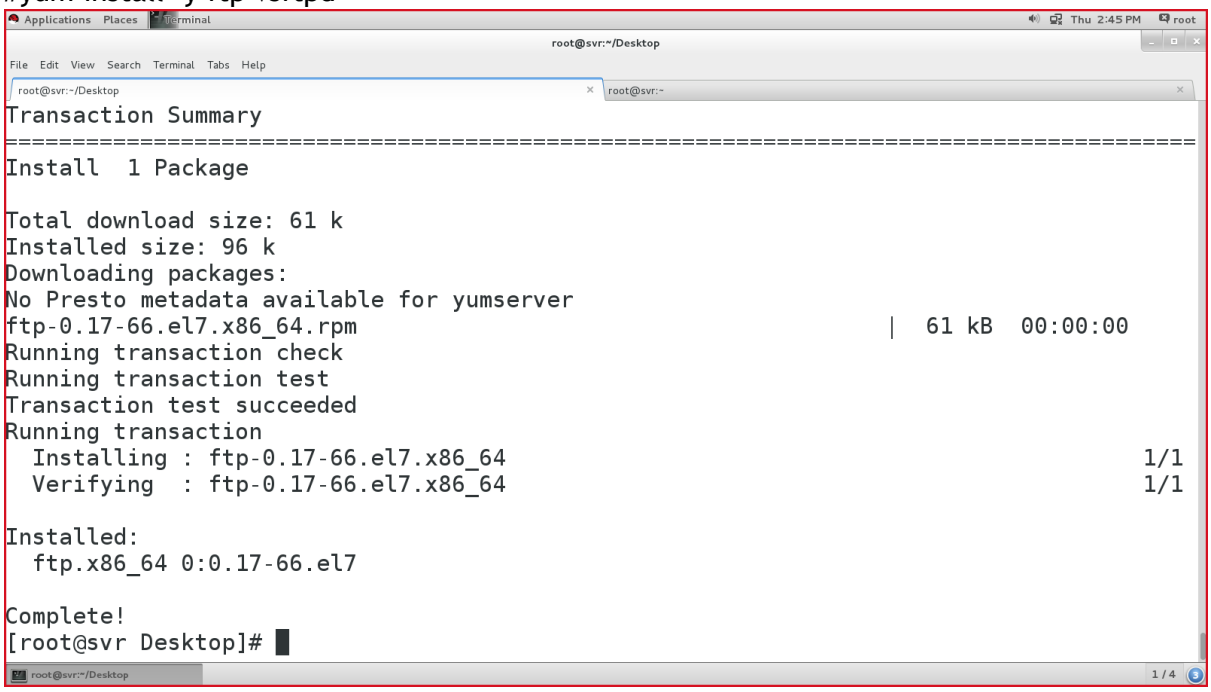
FTP:

- File transfer protocol.
- Port no: 21
- Packages: ftp

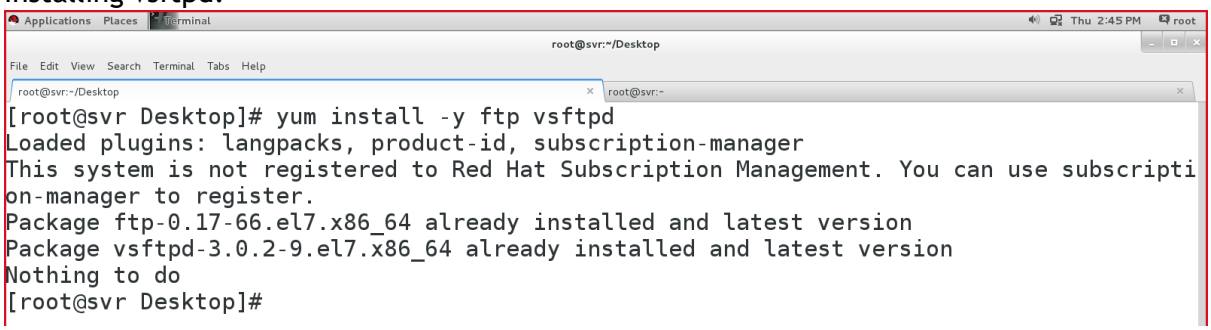
[scenario 1]: install & access simple FTP server

Install FTP packages:

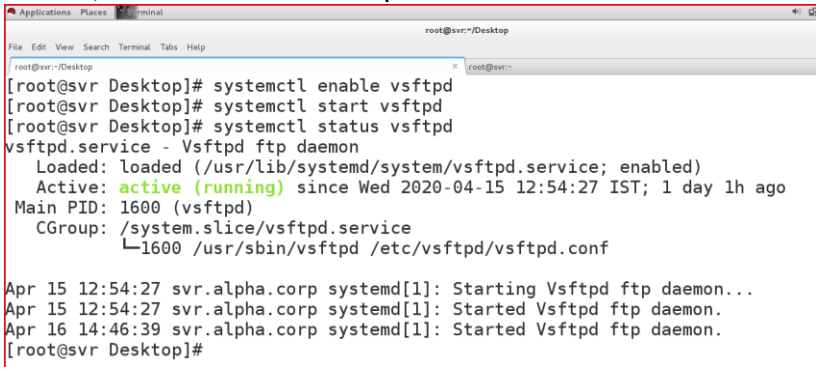
- #yum install -y ftp vsftpd



- Installing vsftpd:



- Enable, start & status of vsftpd:



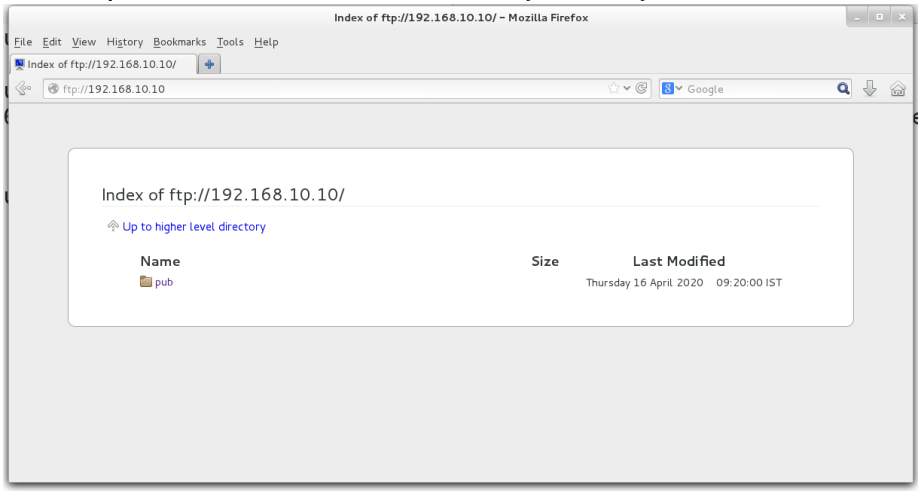
- Make the FTP entry in the firewall.
  #firewall-cmd --list-all

```
[root@svr Desktop]# firewall-cmd --add-port=20/tcp --permanent
success
[root@svr Desktop]# firewall-cmd --reload
success
[root@svr Desktop]# firewall-cmd --list-all
public (default, active)
  interfaces: eno16777736 virbr0
  sources: 192.168.10.0/24 10.10.10.10/8
  services: dhcpv6-client ftp http https ntp ssh
  ports: 21/tcp 123/udp 53/udp 20/tcp 53/tcp
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:

[root@svr Desktop]#
```

- Create a folder in /var/ftp/pub directory, & copy some random data in it.

```
[root@svr Desktop]# cd /var/ftp/pub/
[root@svr pub]# ls
yumserver
[root@svr pub]# mkdir boot-data
[root@svr pub]# cp -var  /boot/* boot-data/
```

- Open firefox & visit ftp://<ip-address-of-ftp-server>
- Either open it on same FTP server or any other system within the network.

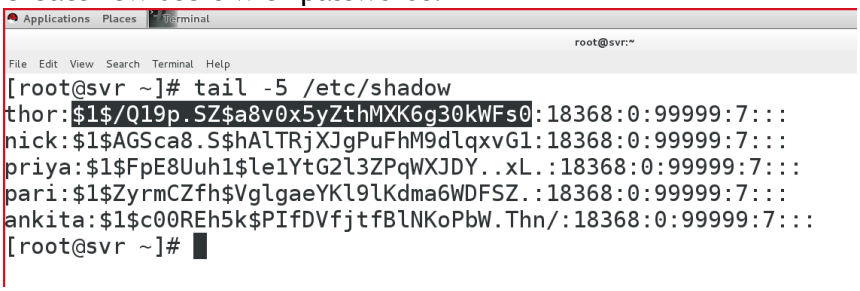


[scenario 2]: secure the FTP server using username & password

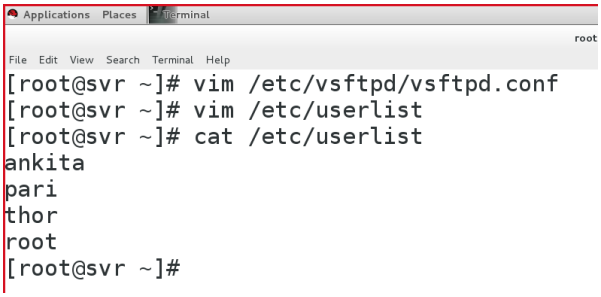
- Install vsftpd package.
- Enable, start & status of VSFTPD service.
- Configure the FTP config file → /etc/vsftpd/vsftpd.conf
- Before you proceed, take the backup of the config file.
  # cp /etc/vsftpd/vsftpd.conf /etc/vsftpd/vsftpd.conf-backup

  #vim /et/vsftpd/vsftpd.conf

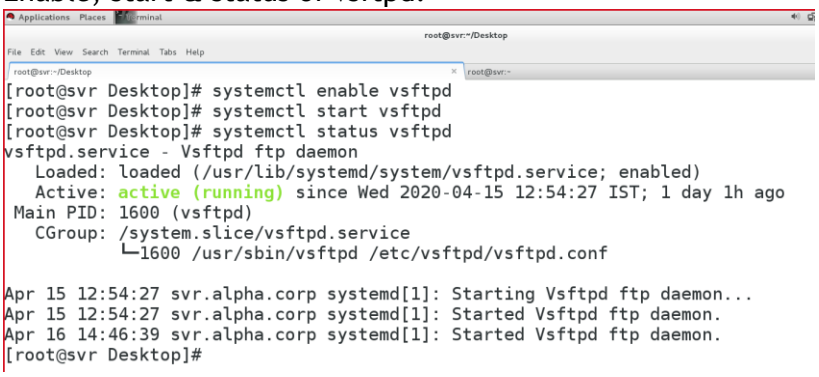
  - Edit line 12, from YES to NO
    anonymous_enable=NO
  - Append (in the bottom of the file), new lines:
    userlist_file=/etc/userlist → allow list of users to access FTP
    userlist_deny=NO → perms to not to block the list.
  - Save & quit.
- Create few users with passwords.



- Create /etc/userlist file & add abv user names in it.

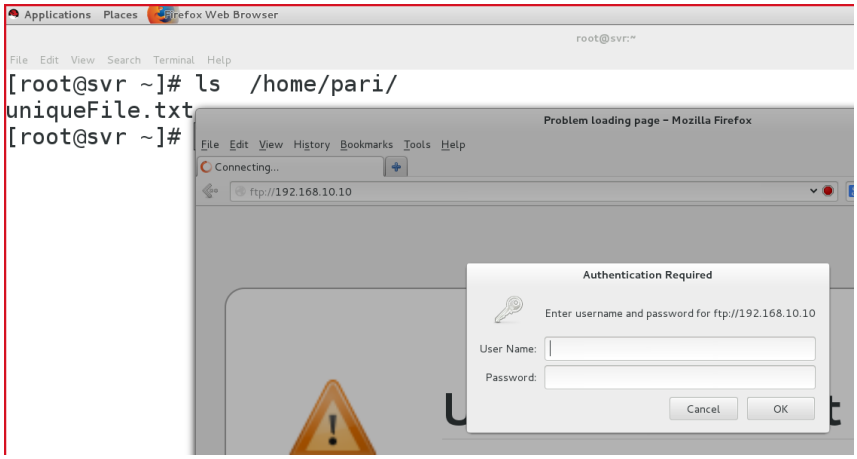


- Enable, start & status of vsftpd:

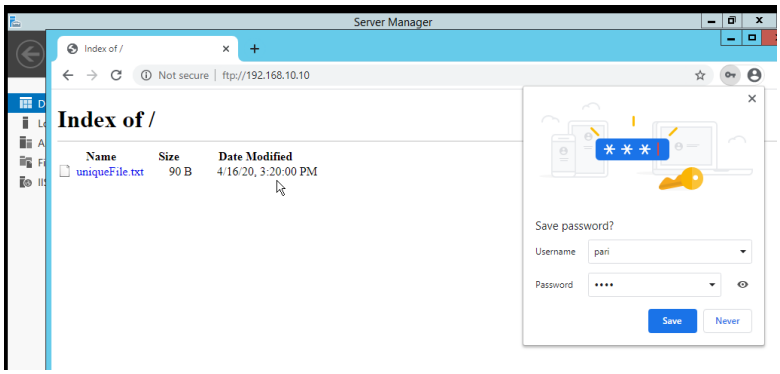


- #systemctl restart vsftpd

On client's web browser:



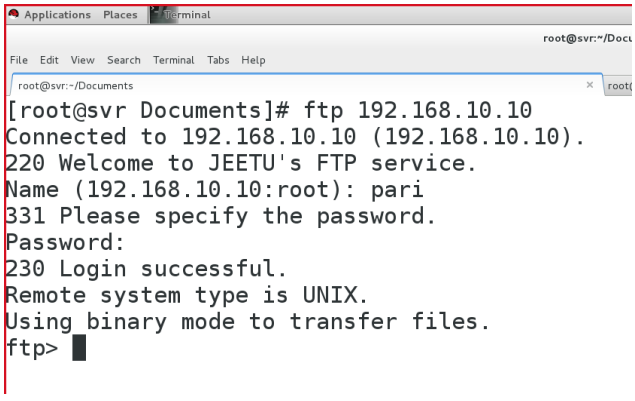
OR



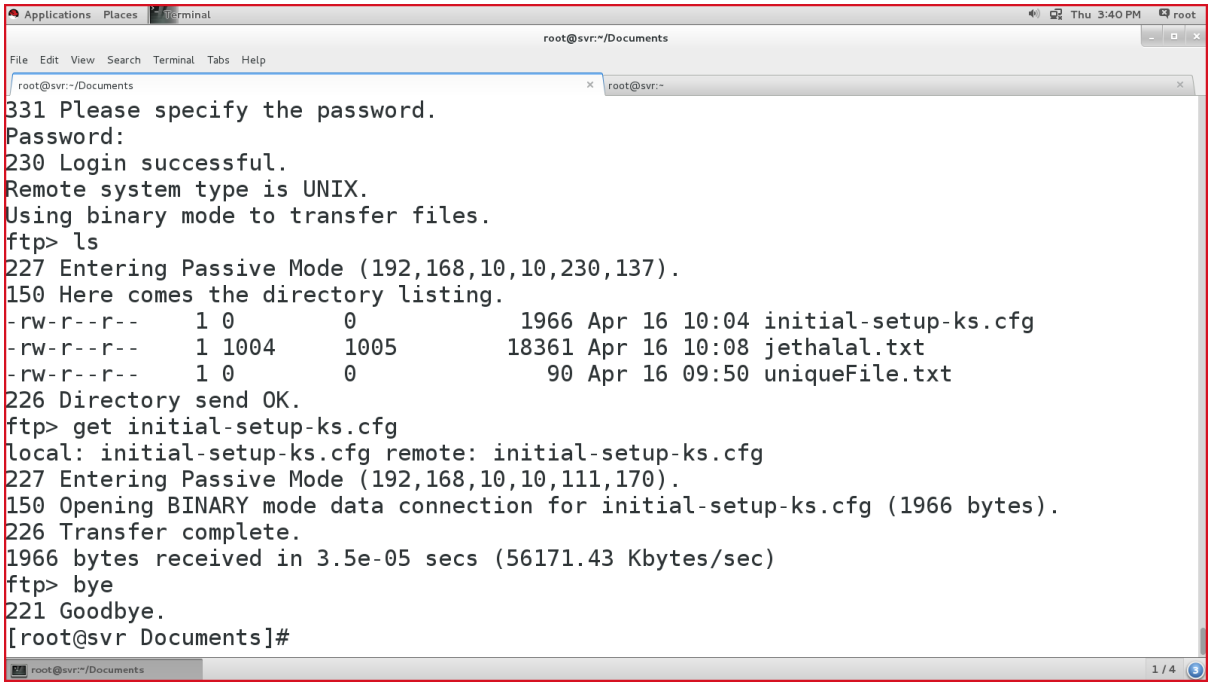
Login to user using linux terminal:

#ftp <ftp-server-ip>

type password of pari user



List & download a file and then quit the ftp console:



SSH

- Secure Shell
- Secure Remote access to the remote systems.
- Port number: 22
- Service name for SSH: sshd

Steps to configure SSH on server & client :

- Install SSH packages:
  #yum install -y openssh*
- Enable, start & status of SSH
  #systemctl enable sshd
  #systemctl start sshd
  #systemctl status sshd

Connect to SSH server from SSH client:

Syntax: #ssh <username>@<ip-address>

```
[root@cli .ssh]# ssh  root@svr.alpha.corp
The authenticity of host 'svr.alpha.corp (192.168.10.10)' can't be established.
ECDSA key fingerprint is SHA256:YNdfQObNFKe6lJgQ8Gs2Uq3zj9IopUKiQAND2ly6DeI.
ECDSA key fingerprint is MD5:6f:97:17:4e:ec:50:96:cf:63:4d:ba:84:28:8d:fa:e9.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'svr.alpha.corp,192.168.10.10' (ECDSA) to the list of known h
osts.
root@svr.alpha.corp's password:
Last login: Wed Apr 15 12:54:54 2020
[root@svr ~]# ls
```

Password-less SSH:

- Ping the two servers (either name or IP)
- Ssh-keygen

```
[root@cli ~]# 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:
SHA256:1KDTBzVX8J+Hjdm7/fO86Uq5HZWPSCiwKr4rHIUZ8Bc root@cli.alpha.corp
The key's randomart image is:
+---[RSA 4096]----+
|o   E   o.o oo.  |
| o   . o + o .   |
|  = . + o o   .  |
| o o   =       Bo|
```

Verify:

```
[root@cli .ssh]# ls
id_rsa  id_rsa.pub  known_hosts
[root@cli .ssh]#
```

- Copy the generated keys to the remote computer.

```
[root@cli .ssh]# ssh-copy-id u5@svr
The authenticity of host 'svr (192.168.10.10)' can't be established.
ECDSA key fingerprint is SHA256:YNdfQObNFKe6lJgQ8Gs2Uq3zj9IopUKiQAND2ly6DeI.
ECDSA key fingerprint is MD5:6f:97:17:4e:ec:50:96:cf:63:4d:ba:84:28:8d:fa:e9.
Are you sure you want to continue connecting (yes/no)? yes
/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 t
he new keys
u5@svr's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'u5@svr'"
and check to make sure that only the key(s) you wanted were added.

[root@cli .ssh]#
```

- Login:

```
[root@cli .ssh]# ssh u5@svr
Last login: Fri Apr 17 08:58:09 2020 from cli
[u5@svr ~]$ ls
not-one-can-hack-me.txt
[u5@svr ~]$ mv not-one-can-hack-me.txt  u-r-hacked.txt
[u5@svr ~]$ ls
u-r-hacked.txt
[u5@svr ~]$ exit
logout
Connection to svr closed.
[root@cli .ssh]# cd
[root@cli ~]#

[root@cli ~]#
```

SCP:

- In windows, we have WinSCP.
- Secure copy.

Copy syntax:

- #cp <source> <destination>

Secure copy syntax:

- #scp <source> <destination>

Source: either on local machine or remote machine

Destination: either on remote machine or local machine.

Use SCP to copy a remote file on local system:

- #scp root@cli.alpha.corp:/root/pwsh.rpm /root/
  - *Source* = root@cli.alpha.corp:/root/pwsh.rpm
  - *Destination* = /root/

Use SCP to copy a local file on to remote system:

- #scp <local-file> <user>@<ip>:<absolute-path>
- scp /root/corona.tar root@cli:/root/tar-file

rsync:

- remote sync
- copy & paste file/directories to & from local m/c & remote m/c.
- It is faster that SCP.
- Uses less bandwidth.
- Incremental backup.

Syntax: #rsync -<option> <source> <destination>

Package for rsync: rync

Cmd:

#rsync -zvh /root/remote/* root@cli:/root/rsync-dir

- 'z' = compress
- 'v' = verbose
- 'h'= human readable

Using rsync with ssh:

- # rsync -avzhe ssh root@cli:/root/rsync-dir/file.tar .

NFS:

- Network file system
- File sharing, L-L or W-W
- Packages: rpcbind, nfs-utils
- File: /etc/exports
- Port number: 2049/tcp

Installing NFS on server:

- # yum install -y nfs-utils rpcbind

Create a directory to be shared on the network.

- #mkdir /nfs
- Create some random files in it.

Edit NFS config file.

- #vim /etc/exports
  - Col1. What to share.
  - Col2. Where(how to share.)
    - * = share to all in the network
    - no_root_squash,rw,sync
      - no_root_squash = all user will have permission as root
      - rw = read/write
      - sync = synchronization
- /nfs *(no_root_squash,rw,sync)
  :wq!

Make the entry in the SVR firewall:

- # firewall-cmd --add-port=2049/tcp --permanent
- #firewall-cmd --add-service=nfs --permanent
- #firewall-cmd --permanent --add-service=rpc-bind
- #firewall-cmd --permanent --add-service=mountd
- #firewall-cmd --permanent --add-port=2049/udp
- #firewall-cmd --reload
- #firewall-cmd --list-all

Export the NFS file system:

- #exportfs

Enable, start & status of the NFS services.

- #systemctl enable rpcbind
- #systemctl start rpcbind
- #systemctl status rpcbind
- #systemctl enable nfs-server.service
- #systemctl start nfs-server.service
- #systemctl status nfs-server.service

####NFS SERVER CONFIG OVER#####

Now switch to NFS client:

Install nfs packages

- #yum install -y nfs-utils rpcbind

Create a mount point to access the NFS server directory

- Mkdir /Lakshman

Enable, start & status nfs-server, rpcbind services:

- #systemctl status rpcbind nfs-server
- #systemctl enable rpcbind nfs-server
- #systemctl start rpcbind nfs-server
- #systemctl status rpcbind nfs-server

Show the NFS shared directories for SVR (192.168.10.10)

- #showmount -e 192.168.10.10

```
File  Edit  View  Search  Terminal  Help
[root@cli ~]# showmount -e 192.168.10.10
Export list for 192.168.10.10:
/nfs *
[root@cli ~]#
```

Mount the NFS shared directory in your system directory (step2)

- # mount 192.168.10.10:/nfs /Lakshman

```
[root@cli ~]# df -h
Filesystem           Size  Used Avail Use% Mounted on
devtmpfs             894M     0  894M   0% /dev
tmpfs                910M     0  910M   0% /dev/shm
tmpfs                910M   11M  900M   2% /run
tmpfs                910M     0  910M   0% /sys/fs/cgroup
/dev/mapper/cl-root   47G  8.7G   39G  19% /
/dev/sda1           1014M  233M  782M  23% /boot
tmpfs                182M  4.0K  182M   1% /run/user/42
tmpfs                182M   28K  182M   1% /run/user/0
192.168.10.10:/nfs    35G  7.9G   27G  23% /lakshman
[root@cli ~]#
```

Samba server

- It is used for sharing the data between cross-platforms, windows to linux & vice-versa.

Lab setup:

- ✓ linux machine (192.168.10.10)
- ✓ windows machine (192.168.10.102)
- ✓ ensure both are pinging.

Packages required:

- # yum install samba-client samba cifs-utils

Add service in firewall:

- # firewall-cmd --permanent --add-service=samba
- #firewall-cmd --reload

Create a group with users & password in it.

- #groupadd grp1
- #useradd -G grp1 user1
- #useradd -G grp1 user2
- #useradd -G grp1 user3
- #grep grp1 /etc/group
- #passwd user1
- #passwd user2
- #passwd user3

Create a directory to be shared using SAMBA

- #mkdir /samba

Link the group (grp1) to the directory

- # chown -R root:grp1 /samba

Configure samba server config file:

- #vim /etc/samba/smb.conf

Edit from the line number: 275.

```
[mysamba]
comment=this is samba shared directory for WIN users
browseable=yes
path=/samba
valid user=@grp1
writable=yes
```

Save & quit

Verify the configuration for “/etc/samba/smb.conf”
#testparm

```
[root@svr Desktop]# testparm
Load smb config files from /etc/samba/smb.conf
Processing section "[homes]"
Processing section "[printers]"
Processing section "[mysamba]"
Unknown parameter encountered: "valid user"
Ignoring unknown parameter "valid user"
Loaded services file OK.
Server role: ROLE_STANDALONE
Press enter to see a dump of your service definitions
```

Hit enter again:

```
[mysamba]
        comment = this is samba shared directory for WIN users
        path = /samba
        read only = No
[root@svr Desktop]#
```

Now add the users in samba access by providing password:

```
[root@svr Desktop]# smbpasswd  -a user1
New SMB password:
Retype new SMB password:
Added user user1.
[root@svr Desktop]# smbpasswd  -a user2
New SMB password:
Retype new SMB password:
Added user user2.
[root@svr Desktop]# smbpasswd  -a user3
New SMB password:
Retype new SMB password:
Added user user3.
[root@svr Desktop]#
```

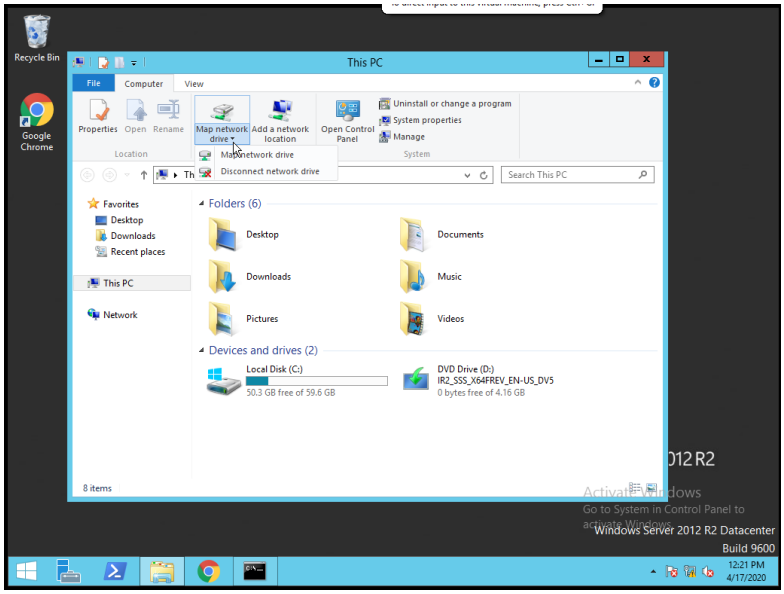
Enable, start & status samba service (smb)

- #systemctl enable smb
- #systemctl start smb
- #systemctl status smb

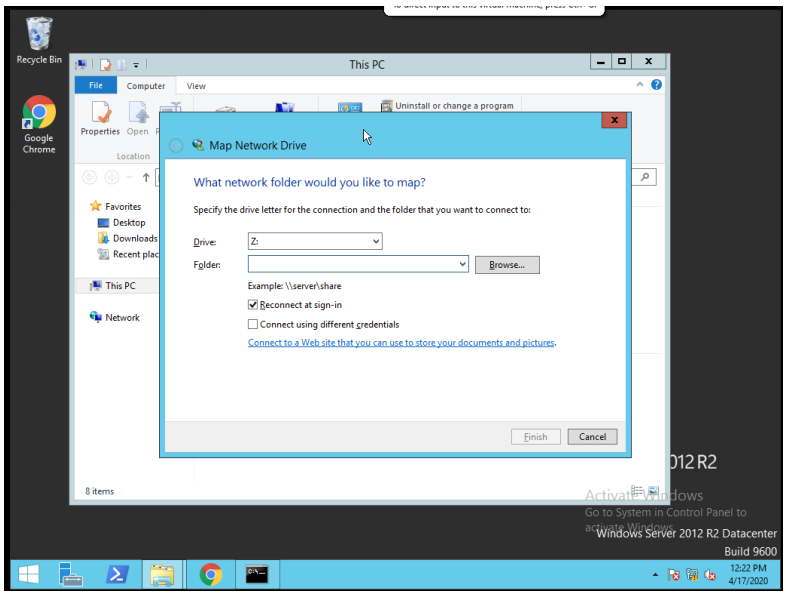
Verify for user1 from linux system:

```
[root@svr Desktop]# smbclient -L localhost -U user1
Enter user1's password:
Domain=[MYGROUP] OS=[Unix] Server=[Samba 4.1.1]

        Sharename       Type      Comment
        ---------       ----      -------
        mysamba         Disk      this is samba shared directory for WIN users
        IPC$            IPC       IPC Service (Samba Server Version 4.1.1)
        user1           Disk      Home Directories
Domain=[MYGROUP] OS=[Unix] Server=[Samba 4.1.1]

        Server               Comment
        ---------            -------

        Workgroup            Master
        ---------            -------
[root@svr Desktop]#
```

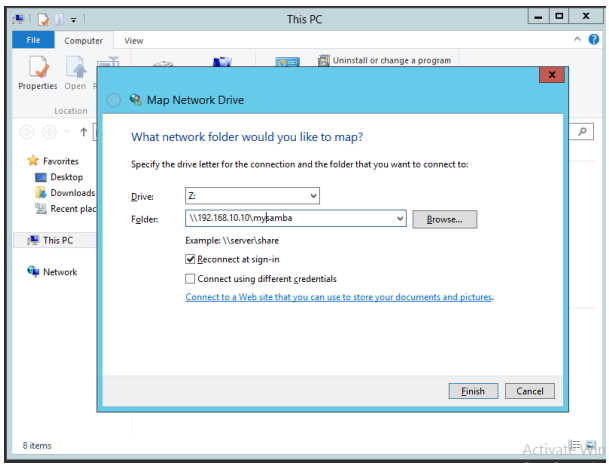
On windows system, open THIS PC, click on computer → Map network drive



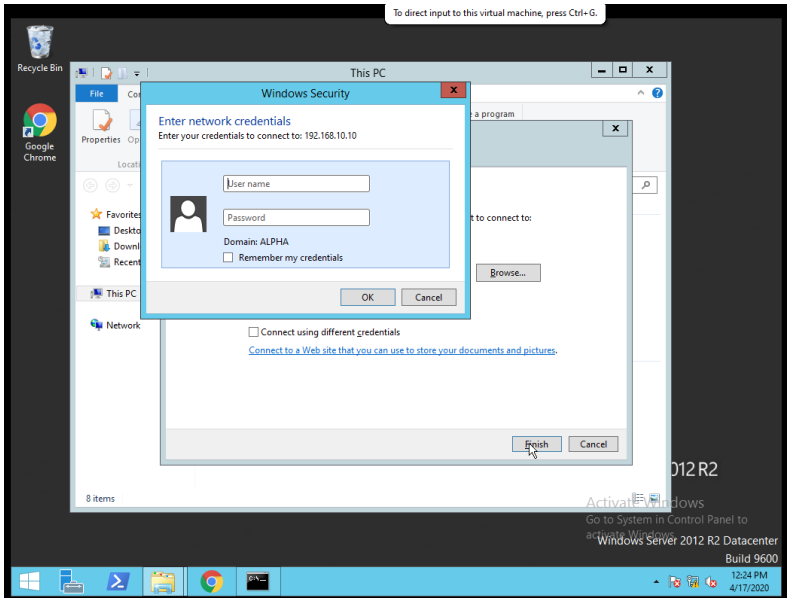
You will get something like:



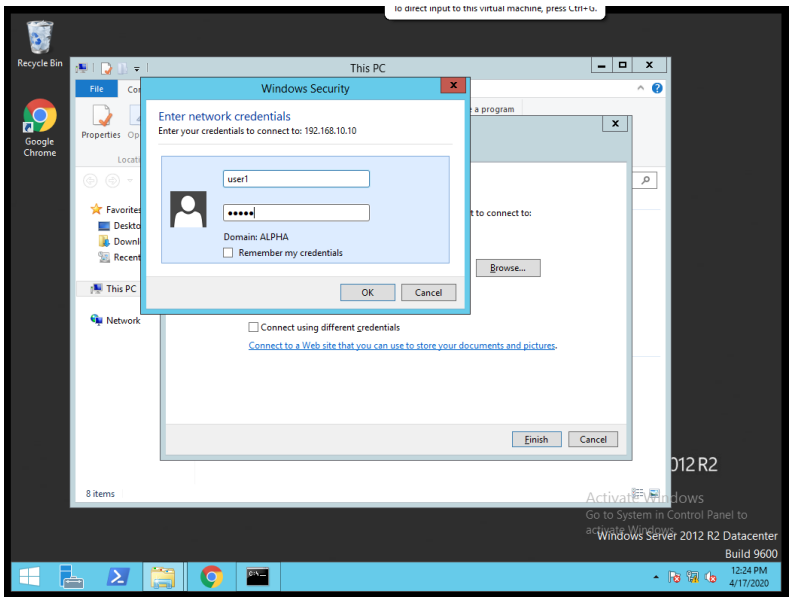
Provide the samba server address with shared folder information:



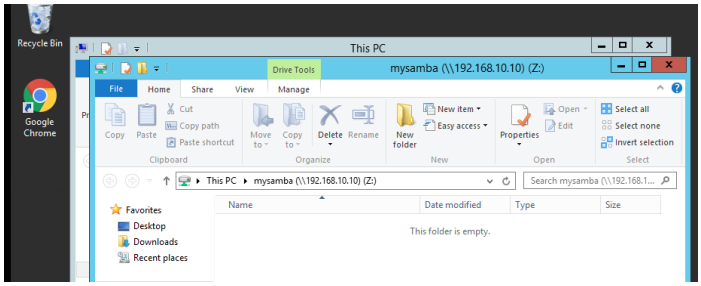
It will prompt the users credentials:



Enter user creds created on Linux server:

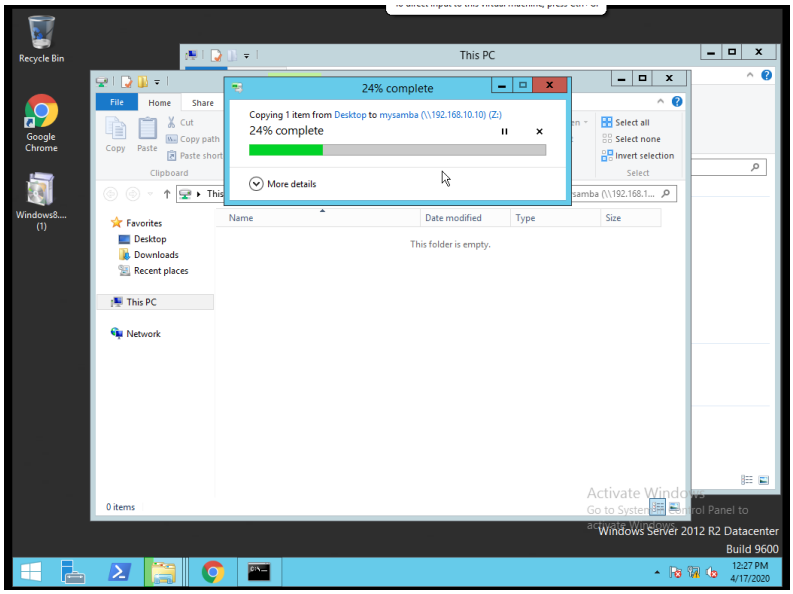


After entering correct creds:

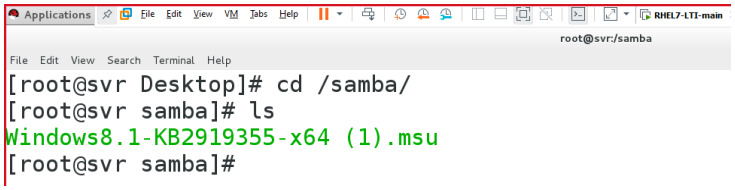


Copy & paste some random files & folders:

If it doesn't allow to copy & paste, change the permission of /samba directory to 777



Switch to Linux VM & verify it the copied data is present:



################SAMBA CONFIG OVER#################

Apache web server:

- Web server in linux
- Similar to IIS web server in windows.
- Used for hosting your website on *intranet* or *internet*.
    - Package required: **httpd**
    - Service required: **httpd**
    - Port number: **80**
    - Config file**: /etc/httpd/conf/httpd.conf**

Install packages:

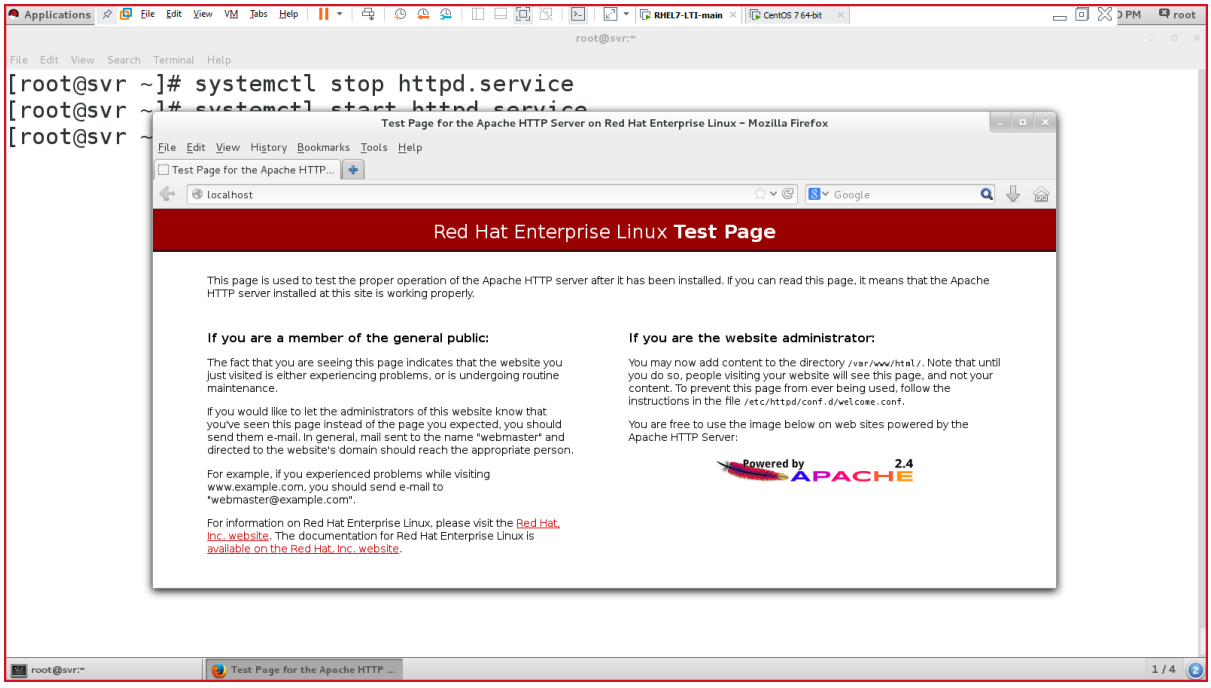
- #yum install -y httpd
  In /etc/passwd → apache (service account)

[scenario -1]: create default web site & access it

Enable, start & status of httpd:

- #systemctl enable httpd.service
- #systemctl start httpd.service
- #systemctl status httpd.service

Open firefox and type http://localhost & verify if the 'test page' is present or not.



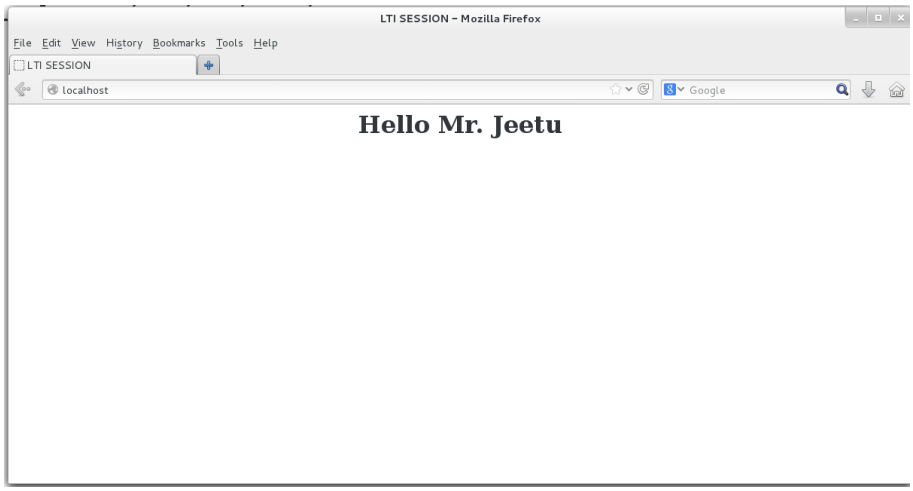
To create your own web page:

#vim /var/www/html/index.html

```
<html>
<head>
<title>LTI SESSION</title>
</head>
<body>
<h1 align='center'>Hello Mr. Jeetu</h1>
</body>
</html>
~
```

save & quit

Go to web browser & refresh the web page.



[scenario -2]: create & access multiple site

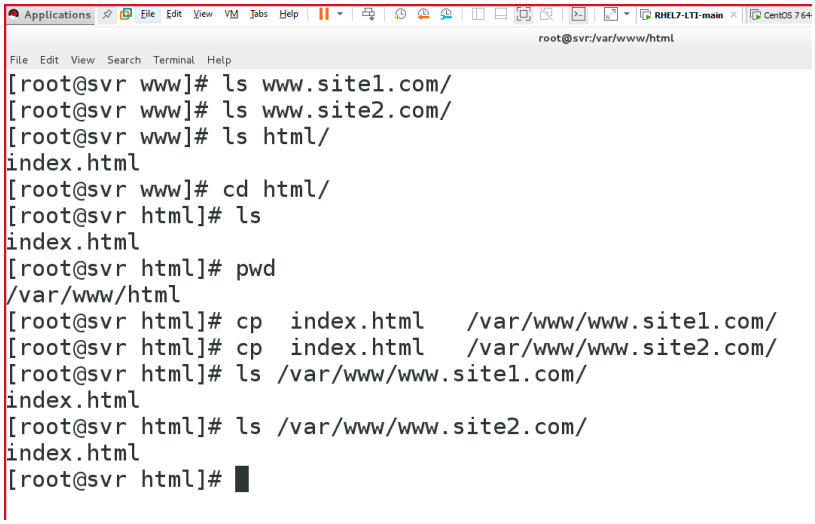
www.site1.com → 192.168.10.10

www.site2.com → 192.168.10.10

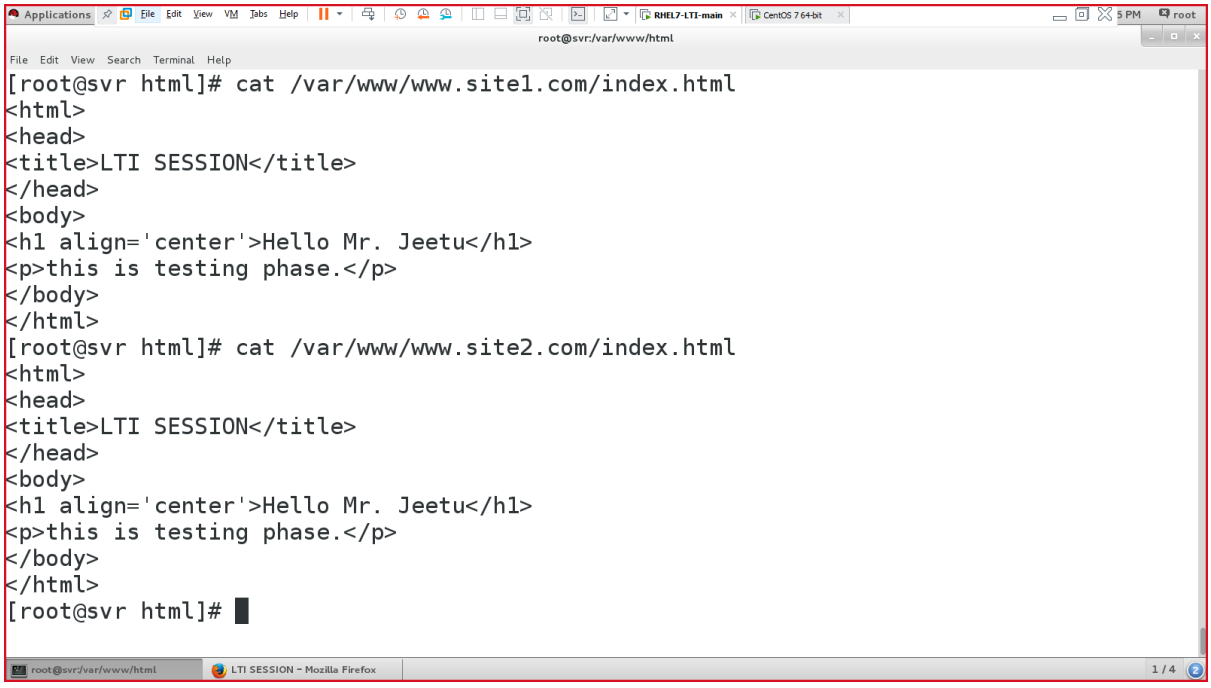
Create 2 folders (in /var/www):

```
[root@svr ~]# cd /var/www/
[root@svr www]# ls
cgi-bin  html
[root@svr www]# mkdir www.site{1,2}.com
[root@svr www]# ls
cgi-bin  html  www.site1.com  www.site2.com
[root@svr www]#
```

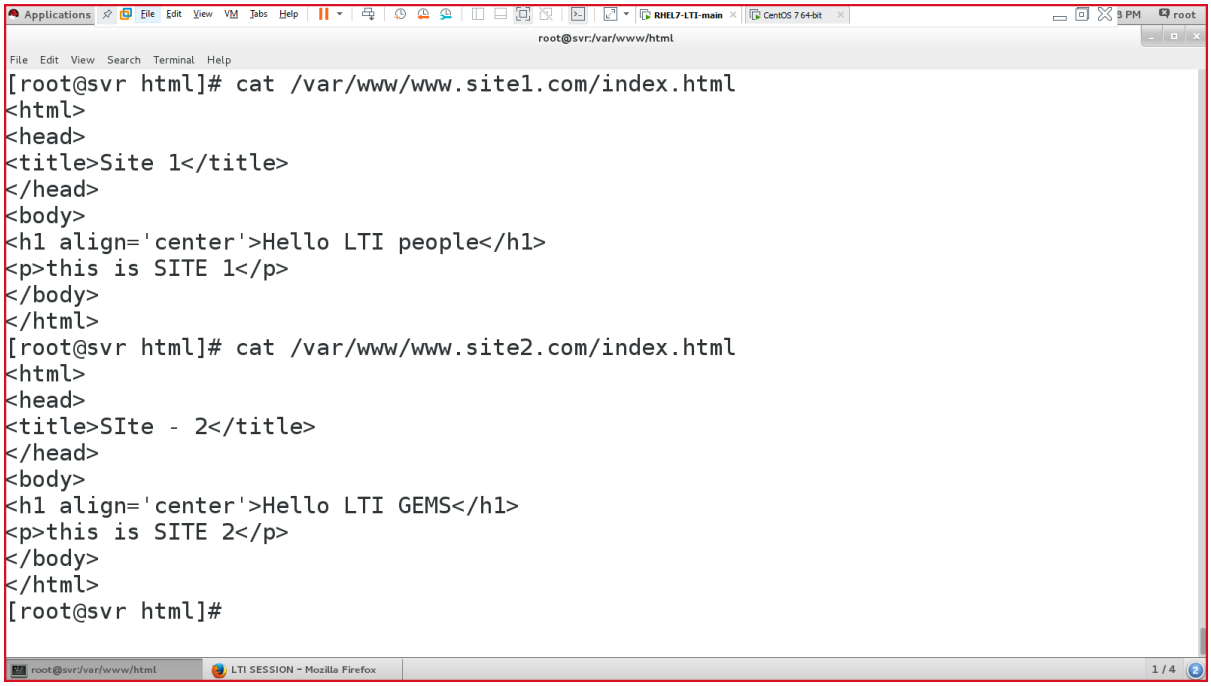
Either create 2 files in the above directories or copy & paste the /var/www/html/index.html file to the above directories



Verify:



Edit the site contents:



Configure the apache config file:

#vim /etc/httpd/conf/httpd.conf

Create a new line below line 42 & add below content

```
#Listen 12.34.56.78:80
Listen 80

<VirtualHost 192.168.10.10:80>
        DocumentRoot /var/www/www.site1.com
        Servername www.site1.com
</VirtualHost>
<VirtualHost 192.168.10.10:80>
        DocumentRoot /var/www/www.site2.com
        Servername www.site2.com
</VirtualHost>
```

save & quit

Make the entries in the /etc/hosts file, to access the site(s) – for name resolution :

```
192.168.10.10   www.site1.com
192.168.10.10   www.site2.com
~
```

:wq!

Enable, start & status httpd service:

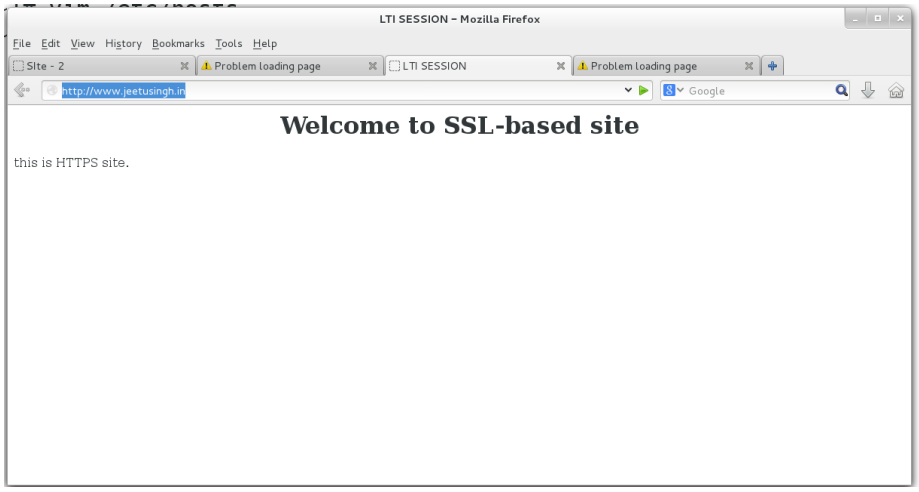
- #systemctl enable httpd
- #systemctl start httpd
- #systemctl status httpd

[scenario -3]: creating HTTPS (using SSL) website

Steps:

- Create a directory in "/var/www/www.jeetusingh.in/"
- Create a web page "/var/www/www.jeetusingh.in/index.html"
- Config httpd file
- Restart the httpd service
- Edit /etc/hosts file

& verify using browser http://www.jeetusingh.in



Configure HTTPS/SSL certificate & attach it to the site (www.jeetusingh.in)

- ✓ Install ssl packages:
  - #yum install -y mod_ssl openssh* openssl*
- ✓ Create SSL certificate:
  - go to certificate location "/etc/pki/tls/certs/"
- ✓ create a key certificate:
  - #make jeetusingh.key & enter the pass phrase
  - #ls

```
[root@svr certs]# ls
ca-bundle.crt        jeetusingh.key  make-dummy-cert  renew-dummy-cert
ca-bundle.trust.crt  localhost.crt   Makefile
[root@svr certs]#
```

- ✓ Create a CSR file
  - #openssl rsa -in jeetusingh.key -out jeetusingh.key
- ✓ Create a CRT file
  - #cp localhost.crt jeetusingh.crt
  - #ls → look for jeetusingh.crt file in the directory.

```
[root@svr certs]# ls
ca-bundle.crt        jeetusingh.crt  localhost.crt    Makefile
ca-bundle.trust.crt  jeetusingh.key  make-dummy-cert  renew-dummy-cert
[root@svr certs]#
```

✓ Create a new “jeetusingh.csr” file

```
[root@svr certs]# make jeetusingh.csr
umask 77 ; \
/usr/bin/openssl req -utf8 -new -key jeetusingh.key -out jeetusingh.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:IN
State or Province Name (full name) []:KA
Locality Name (eg, city) [Default City]:BLR
Organization Name (eg, company) [Default Company Ltd]:IIHT
Organizational Unit Name (eg, section) []:Trainer
Common Name (eg, your name or your server's hostname) []:jeetusingh.in
Email Address []:jitendra.singh@iiht.com

Please enter the following 'extra' attributes
to be sent with your certificate request
```

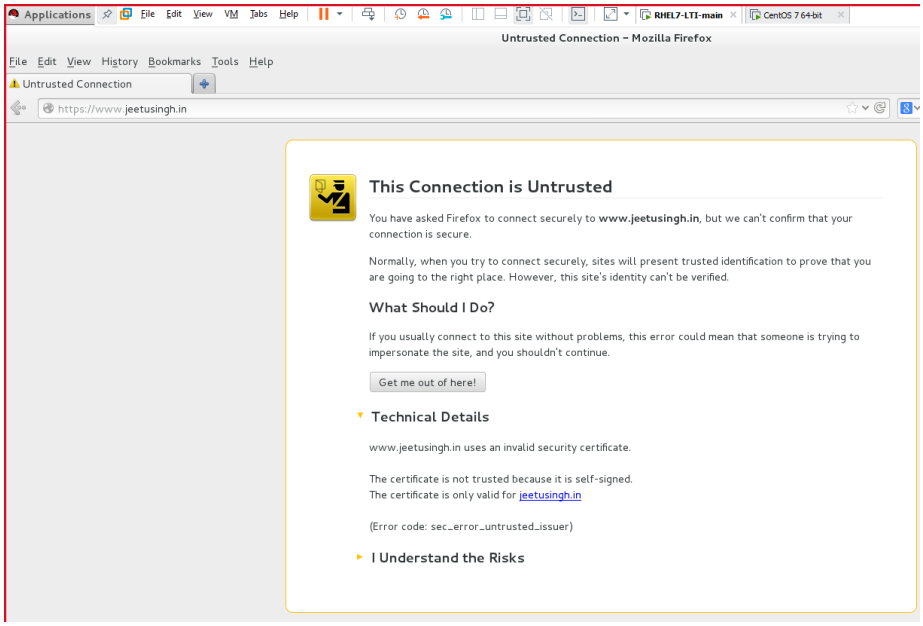
✓ Validate & bind the certificate:
#openssl x509 -in jeetusingh.csr -out jeetusingh.crt -req -signkey jeetusingh.key -days 365

```
root@svr:/etc/pki/tls/certs
File Edit View Search Terminal Help
[root@svr certs]# openssl x509 -in jeetusingh.csr -out jeetusingh.crt -req -signkey j
eetusingh.key -days 365
Signature ok
subject=/C=IN/ST=KA/L=BLR/O=IIHT/OU=Trainer/CN=jeetusingh.in/emailAddress=jitendra.singh
@iiht.com
Getting Private key
[root@svr certs]#
```

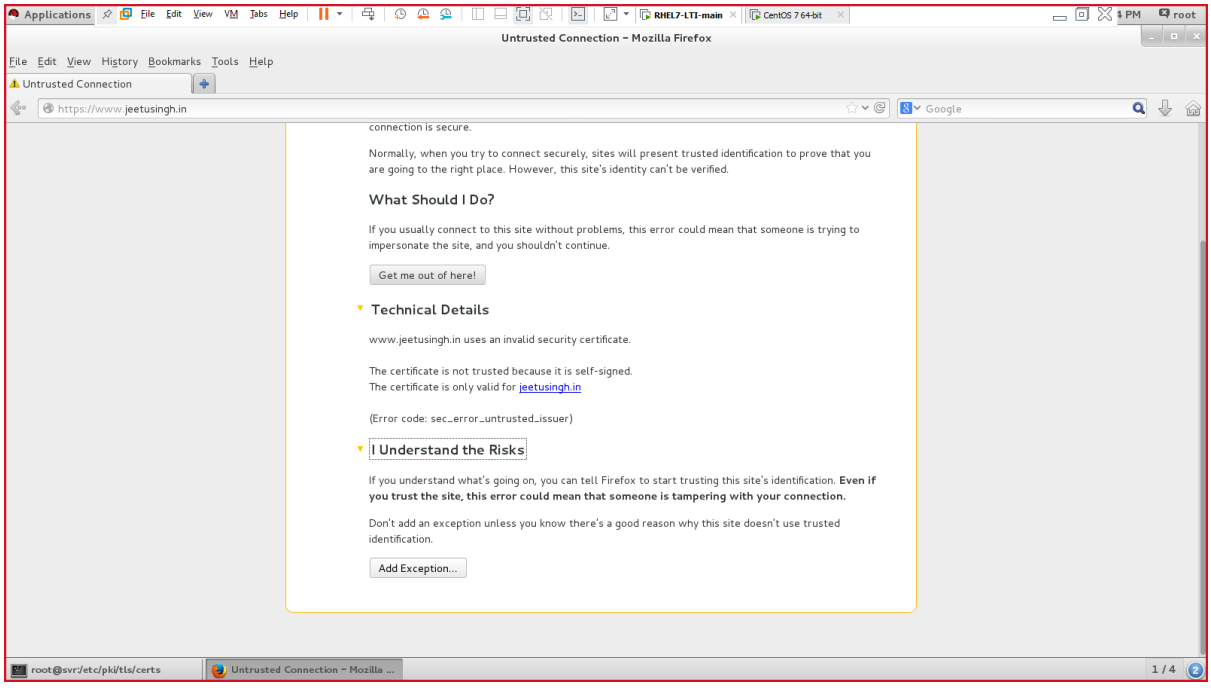
✓ Configure the SSL config file:
- #vim /etc/httpd/conf.d/ssl.conf
- Uncomment line 59 & 60 & edit accordingly
- On line 100,
  SSLCertificateFile /etc/pki/tls/certs/jeetusingh.crt
- On line 107,
  SSLCertificateKeyFile /etc/pki/tls/certs/jeetusingh.key
  :wq!

✓ Restart & status httpd service:
#systemctl restart httpd
#systemctl status httpd

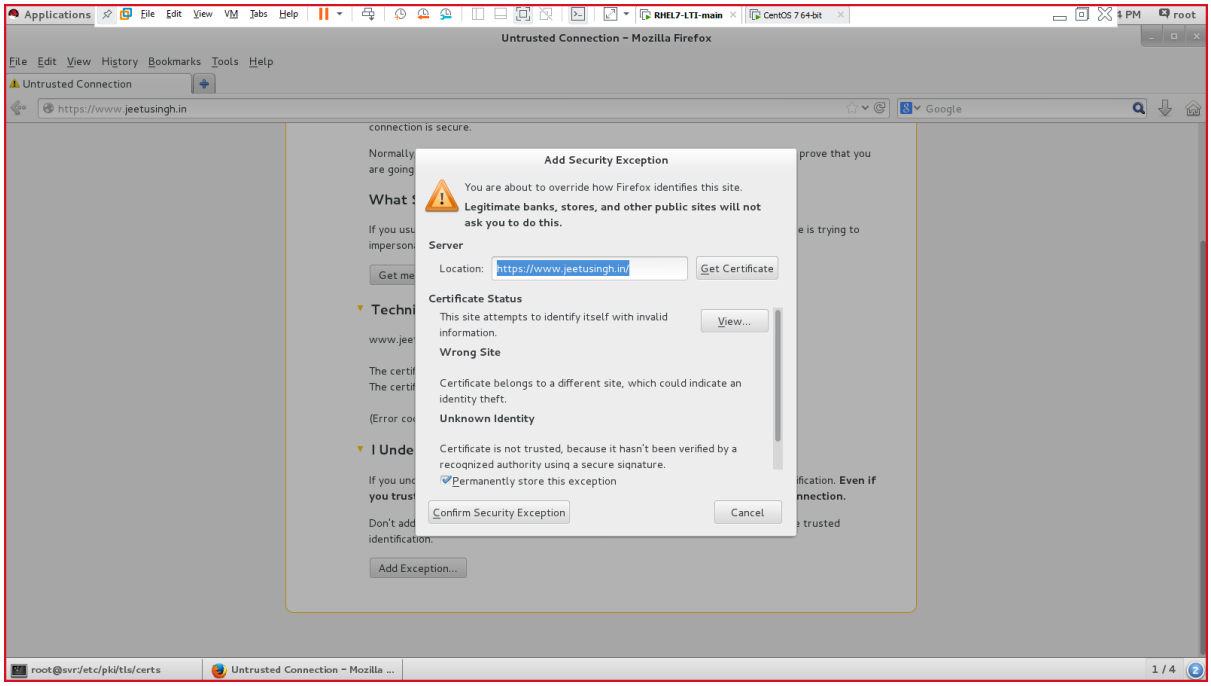
✓ Go to web browser & type https://www.jeetusingh.in



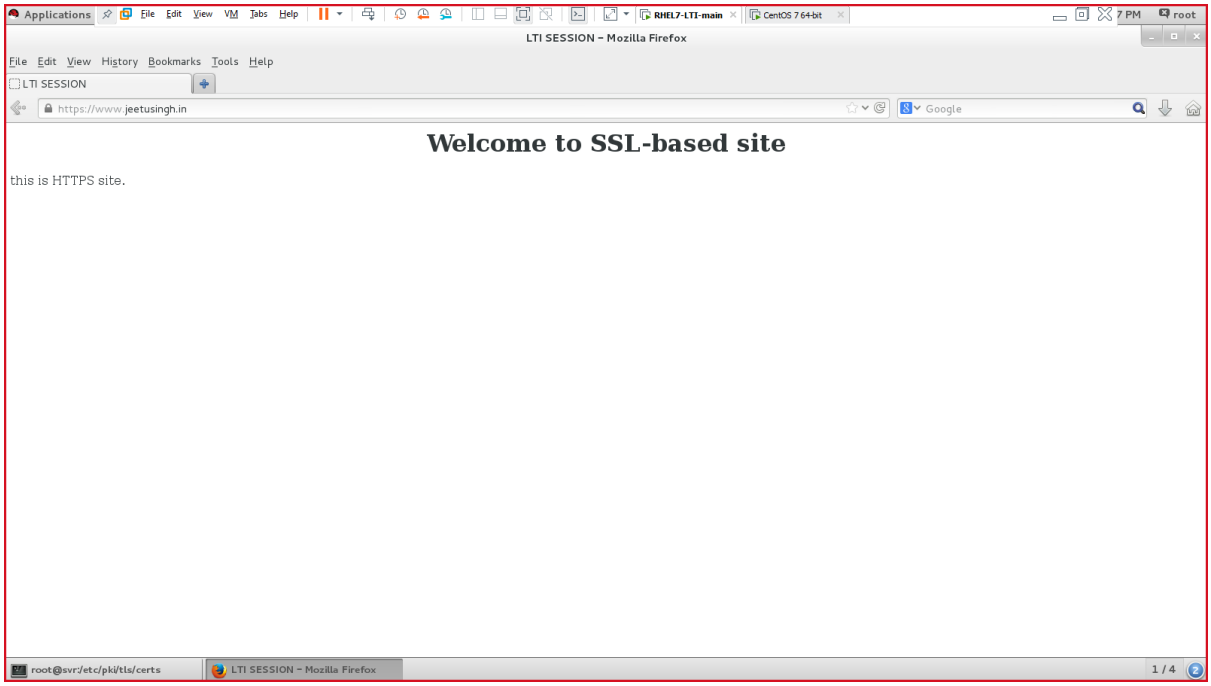
Click on “I Understand the Risks” & click on “Add exception”

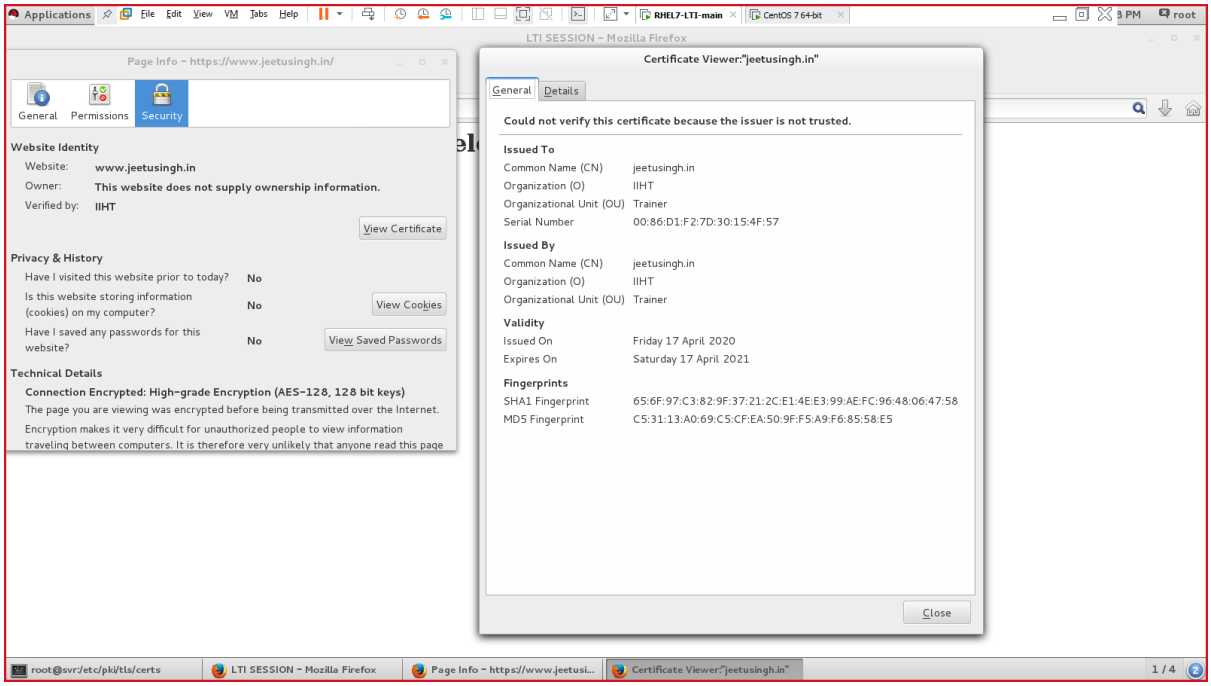


Click on “Confirm security exception”



Access the site on HTTPS network:




Applications
File Edit View VM Tabs Help
RHEL7-LTI-main
CentOS 7 64-bit
3 PM
root
LTI SESSION – Mozilla Firefox
Page Info – https://www.jeetusingh.in/
General
Permissions
Security
Website Identity
Website: www.jeetusingh.in
Owner: This website does not supply ownership information.
Verified by: IIHT
View Certificate
Privacy & History
Have I visited this website prior to today? No
Is this website storing information (cookies) on my computer? No
View Cookies
Have I saved any passwords for this website? No
View Saved Passwords
Technical Details
Connection Encrypted: High-grade Encryption (AES-128, 128 bit keys)
The page you are viewing was encrypted before being transmitted over the Internet.
Encryption makes it very difficult for unauthorized people to view information traveling between computers. It is therefore very unlikely that anyone read this page
Certificate Viewer:"jeetusingh.in"
General
Details
Could not verify this certificate because the issuer is not trusted.
Issued To
Common Name (CN) jeetusingh.in
Organization (O) IIHT
Organizational Unit (OU) Trainer
Serial Number 00:86:D1:F2:7D:30:15:4F:57
Issued By
Common Name (CN) jeetusingh.in
Organization (O) IIHT
Organizational Unit (OU) Trainer
Validity
Issued On Friday 17 April 2020
Expires On Saturday 17 April 2021
Fingerprints
SHA1 Fingerprint 65:6F:97:C3:82:9F:37:21:2C:E1:4E:E3:99:AE:FC:96:48:06:47:58
MD5 Fingerprint C5:31:13:A0:69:C5:CF:EA:50:9F:F5:A9:F6:85:58:E5
Close
root@svr:/etc/pki/tls/certs
LTI SESSION – Mozilla Firefox
Page Info – https://www.jeetusi...
Certificate Viewer:"jeetusingh.in"
1 / 4

Telnet:

- ✓ Remote connection
- ✓ Similar to SSH
- ✓ BUT: it is not SECURE

Lab setup:

- ➢ System 1: telnet server → 192.168.10.10
- ➢ System 2: telnet client → 192.168.10.11

Install telnet packages:

- ➢ #yum install -y telnet telnet-server xinetd

On telnet server:

#vim /etc/xinetd.conf

- :set nu
- Go to line 14, and uncomment it.
- Then, write
- disabled = no

:wq!

Enable, start & status of xinetd & telnet services

- ➢ #systemctl enable xinetd telnet.socket
- ➢ #systemctl start xinetd telnet.socket
- ➢ #systemctl status xinetd telnet.socket

###########telnet server config over######

On telnet client machine,

- ➢ Install telnet services
  #yum install -y telnet*
- ➢ Enable, start & status telnet services
  #systemctl enable telnet.socket
  #systemctl start telnet.socket
  #systemctl status telnet.socket
- ➢ Connect to telnet server via client machine:

```
[root@cli ~]# telnet 192.168.10.10
Trying 192.168.10.10...
Connected to 192.168.10.10.
Escape character is '^]'.

Kernel 3.10.0-123.el7.x86_64 on an x86_64
svr login: rocky
Password:
[rocky@svr ~]$ hostname
svr.alpha.corp
[rocky@svr ~]$ whoami
rocky
[rocky@svr ~]$ exit
logout
Connection closed by foreign host.
[root@cli ~]#
[root@cli ~]#
```

Connecting to telnet using telnet server:

```
File  Edit  View  Search  Terminal  Help
[root@svr ~]# telnet 192.168.10.10
Trying 192.168.10.10...
Connected to 192.168.10.10.
Escape character is '^]'.

Kernel 3.10.0-123.el7.x86_64 on an x86_64
svr login: rocky
Password:
Last login: Sat Apr 18 08:18:23 from ::ffff:192.168.10.11
[rocky@svr ~]$ exit
logout
Connection closed by foreign host.
[root@svr ~]#
```

Connect to a service to check if the service is working.

```
[root@svr ~]# telnet localhost smtp
Trying ::1...
Connected to localhost.
Escape character is '^]'.
220 svr.alpha.corp ESMTP Postfix
```

SQUID server:

- It is a proxy server
- Filtering the traffic on the network
- Blocking the unwanted sites within the network
- Squid server uses port 3128
- Config file: /etc/squid/squid.conf
- Service: squid

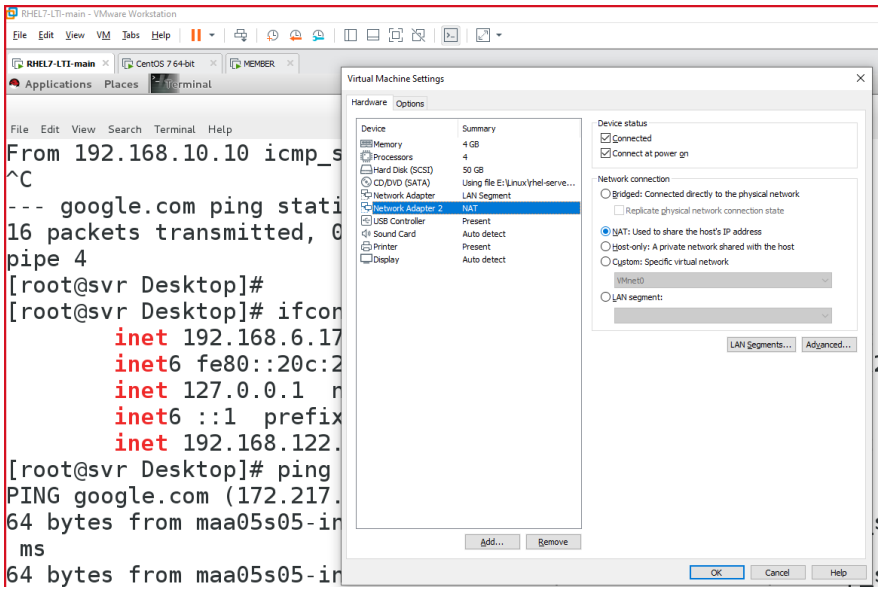
Lab setup:

- One linux based server (squid server) (svr -- internet is present)
- One linux / windows-based client. (cli -- internet is present)

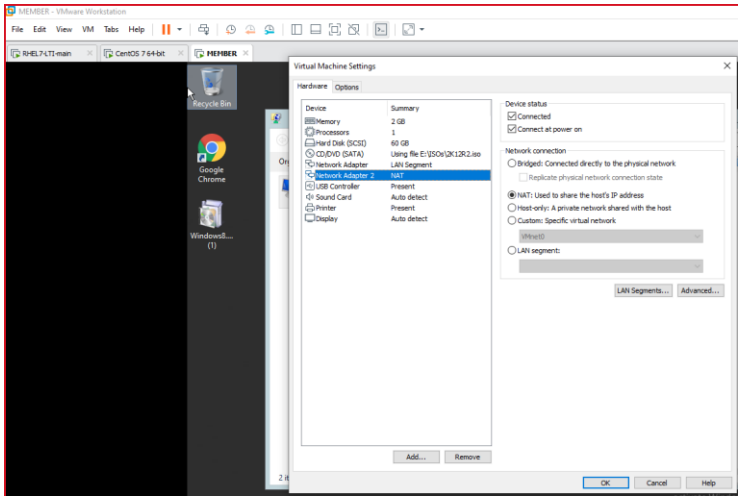
Steps

- Install squid server using yum.
  #yum install -y squid OR
  #rpm -ivh /run/media/root/RHEL…/Packages/squid<hit-tab>.rpm (enter)
- Add additional NIC card in the server VM & client VM with NAT service.

Server machine:



Client machine:



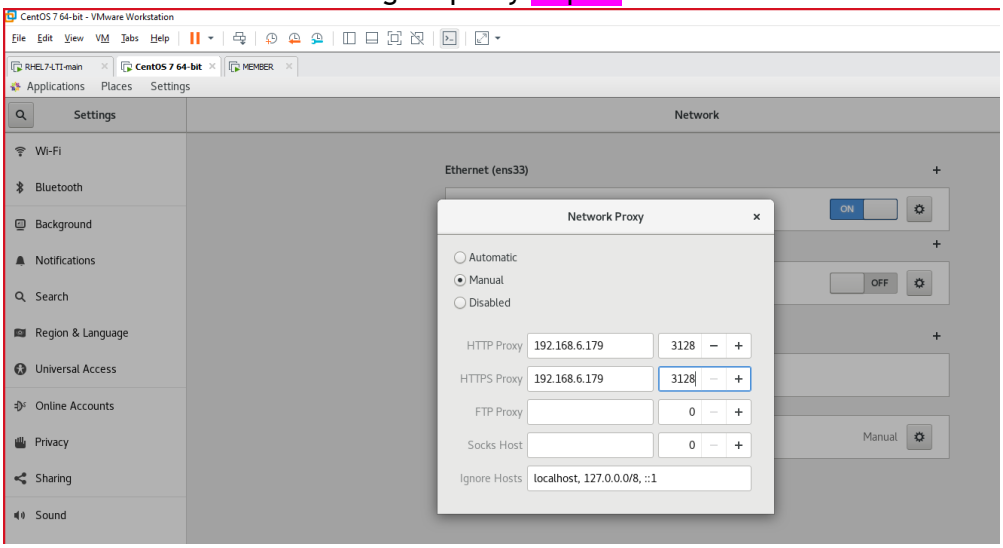
- Disable internal network with the VMs

Install SQUID on the server machine.

- #yum install -y squid

Steps:

- On client, open any webpage (link facebook.com)
- Start the monitoring of the client.
- Go to client machine & configure proxy IP:port on the client.



Enable, start & status of squid service:

- #systemctl enable squid
- #systemctl start squid
- #systemctl status squid

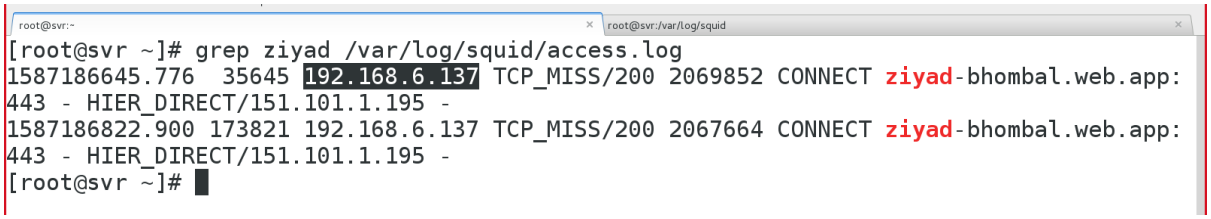
Log file of squid: /var/log/squid/access.log

- #vim /etc/squid/squid.conf
- Set the line numbers (:set nu)
- Add a new entry in the line number 11
  acl localnet src 192.168.6.0/24 → copy the content from line 10 & add ur IP.
- Go to line 38, and append localnet in the line (ex😊)
  http_access allow localhost manager localnet
- Save & quit the file (:wq!)

Run command: tail -f /var/log/squid/access.log

Go to client & open https://ziyad-bhombal.web.app

Switch back to squid server & verify the logs contain keywork “ziyad-bhombal.web.app”.



NOW: To block the unwanted websites using SQUID

- Go to squid config file
- Undo the changes, and add the below content on the top of the file:
  acl badsites url_regex "/etc/squid/badsites"
  http_access deny badsites
  acl pgnetwork src 192.168.6.0/24
  http_access allow pgnetwork
- :wq!

Create the file to block the websites without '-‘ sign:

#cat > /etc/squid/badsites

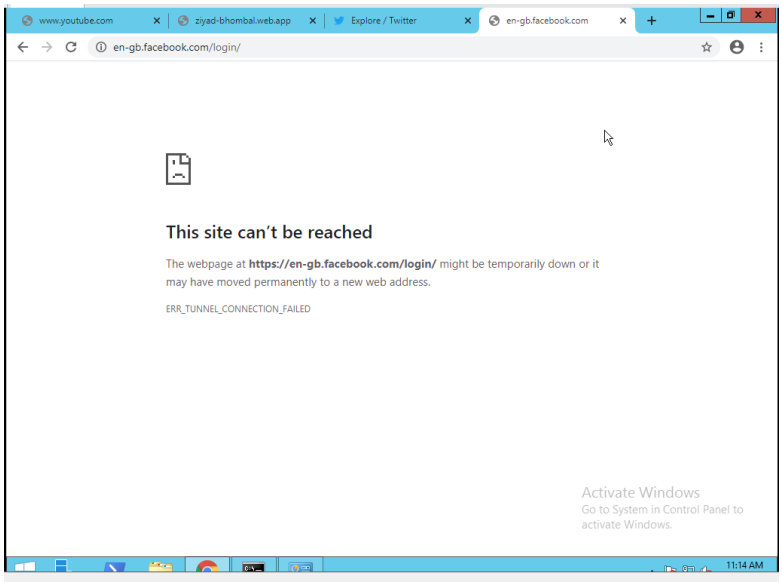
- Facebook.com
- Youtube.com
- ziyad-bhombal.web.app

press ctrl+d

Restart the SQUID service:

- #systemctl restart squid
- #systemctl status squid

Go to client machine to verify the sites are working.



##########END OF SQUID SERVER ###############

DNS server:

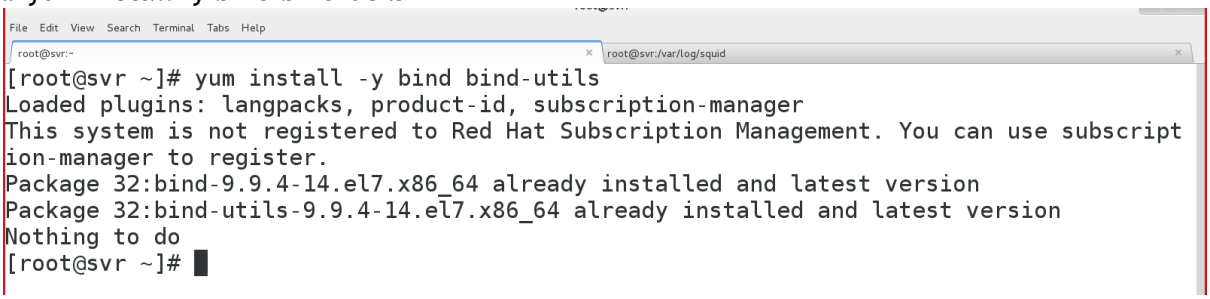
- Resolves IP into name & vice-versa.
- Local system
  - Linux → /etc/hosts
  - Windows → C:\Windows\System32\drivers\etc\hosts
- Port number: 53
- Service: named
- Files:
  - /etc/named.conf (main config file)
  - /var/named/forward.alpha.corp (create a new file)
  - /var/named/reverse.alpha.corp (create a new file)

Lab setup:

- One server
  - DNS server
  - Static IP
  - Add DNS port in firewall or turn off
  - Internet is not required.
  - Hostname must a domain-based name (svr.alpha.corp)
- One/multiple client (windows or linux)
  - DNS client
  - Static/dynamic IP
  - Add DNS port in firewall or turn off
  - Internet is not required.
  - Hostname must a domain-based name (cli.alpha.corp)
  - PING SERVER & CLIENT SYSTEM BY IP ADDRESS
- Disable SQUID server on client & server machine

Install BIND/DNS

- # yum install -y bind bind-utils



- Edit /etc/named.conf file:
  On line 11, add your DNS ip address at the end. Followed by ';'
  listen-on port 53 { 127.0.0.1; 192.168.10.10; }
  on line 17, add 'any;' to the end:
  allow-query { localhost; any; };

```
10 options {
11         listen-on port 53 { 127.0.0.1; 192.168.10.10;  };
12         listen-on-v6 port 53 { ::1; };
13         directory       "/var/named";
14         dump-file       "/var/named/data/cache_dump.db";
15         statistics-file "/var/named/data/named_stats.txt";
16         memstatistics-file "/var/named/data/named_mem_stats.txt"
17         allow-query     { localhost; any; };
18
19         /*
-- INSERT --
```

:wq!

Enable, start & status 'named' service:

- #systemctl enable named
- #systemctl start named
- #systemctl status named → active (Running)

Open & modify /etc/named.conf file again:

Go to line 55, & add below data in it.

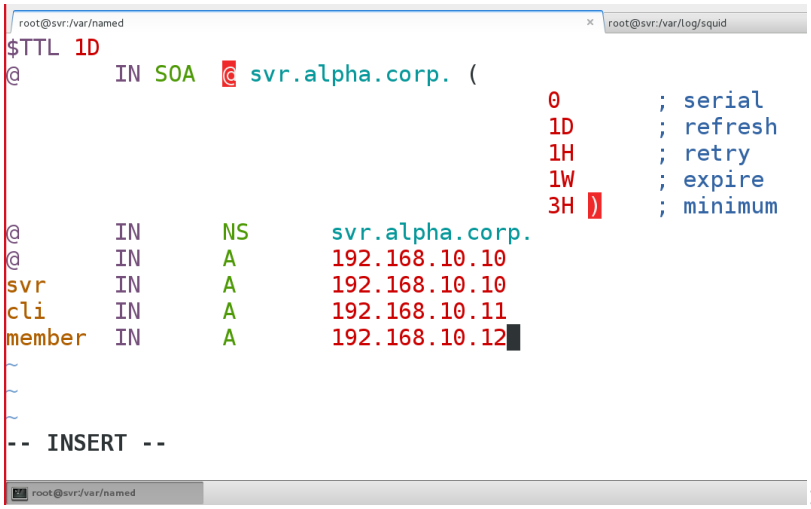
```
zone "alpha.corp" IN {
type master;
file "forward.alpha.corp";
allow-update {none;};
};

zone "10.168.192.in-addr-arpa" IN {
type master;
file "reverse.alpha.corp";
allow-update {none;};
};

include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
-- INSERT --
```

```
zone "alpha.corp" IN {
type master;
file "forward.alpha.corp";
allow-update {none;};
};

zone "10.168.192.in-addr-arpa" IN {
type master;
file "reverse.alpha.corp";
allow-update {none;};
};
```

#cd /var/named; ls → look for the named.localhost file

#cp named.localhost forward.alpha.corp

#vim forward.alpha.corp

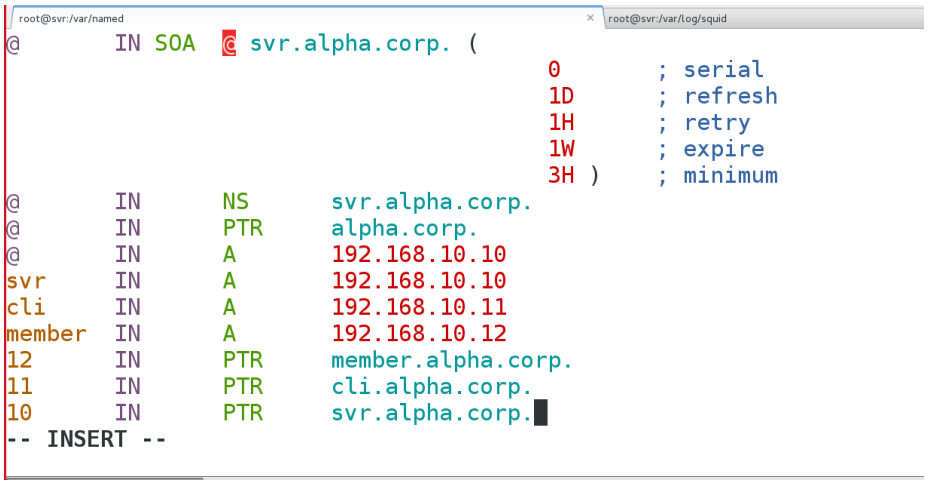


:wq!

```
$TTL 1D
@       IN SOA  @ svr.alpha.corp. (
                                0       ; serial
                                1D      ; refresh
                                1H      ; retry
                                1W      ; expire
                                3H )    ; minimum
@       IN      NS      svr.alpha.corp.
@       IN      A       192.168.10.10
svr     IN      A       192.168.10.10
cli     IN      A       192.168.10.11
member  IN      A       192.168.10.12

save & quit
```

#cp forward.alpha.corp reverse.alpha.corp

#vim reverse.alpha.corp




```
@       IN SOA  @ svr.alpha.corp. (
```

```
                                0       ; serial
                                1D      ; refresh
                                1H      ; retry
                                1W      ; expire
                                3H )    ; minimum
@       IN      NS      svr.alpha.corp.
@       IN      PTR     alpha.corp.
@       IN      A       192.168.10.10
svr     IN      A       192.168.10.10
cli     IN      A       192.168.10.11
member  IN      A       192.168.10.12
12      IN      PTR     member.alpha.corp.
11      IN      PTR     cli.alpha.corp.
10      IN      PTR     svr.alpha.corp.

Save & quit
```

[IMP: Change the ownership of the forward.alpha.corp & reverse.alpha.corp file]

```
root@svr:/var/named                                    root@svr:/var/log/squid
[root@svr named]# chown root:named forward.alpha.corp
[root@svr named]# chown root:named reverse.alpha.corp
[root@svr named]# ll
total 24
drwxr-x---. 7 root  named   56 Apr 14 16:11 chroot
drwxrwx---. 2 named named   22 Apr 18 11:58 data
drwxrwx---. 2 named named   58 Apr 18 11:58 dynamic
-rw-r-----. 1 root  named  241 Apr 18 12:22 forward.alpha.corp
-rw-r-----. 1 root  named 2076 Jan 28  2013 named.ca
-rw-r-----. 1 root  named  152 Dec 15  2009 named.empty
-rw-r-----. 1 root  named  152 Jun 21  2007 named.localhost
-rw-r-----. 1 root  named  168 Dec 15  2009 named.loopback
-rw-r-----. 1 root  named  343 Apr 18 12:27 reverse.alpha.corp
drwxrwx---. 2 named named    6 Jan 29  2014 slaves
[root@svr named]#
```

Verify the named.conf

#named-checkconf -z /etc/named.conf

```
[root@svr named]# named-checkconf -z /etc/named.conf
zone alpha.corp/IN: loaded serial 0
zone 10.168.192.in-addr-arpa/IN: loaded serial 0
zone localhost.localdomain/IN: loaded serial 0
zone localhost/IN: loaded serial 0
zone 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ip6.arpa/IN: loade
d serial 0
zone 1.0.0.127.in-addr.arpa/IN: loaded serial 0
zone 0.in-addr.arpa/IN: loaded serial 0
[root@svr named]#
```

Output of the command must give → "…loaded serial 0"

Verify forward.alpha.corp

named-checkzone forward forward.alpha.corp

```
root@svr:/var/named                                    root@svr:/var/log/squid
[root@svr named]# named-checkzone  forward  forward.alpha.corp
zone forward/IN: loaded serial 0
OK
[root@svr named]#
```

Output → “loaded serial 0”

Verify reverse.alpha.corp

named-checkzone reverse reverse.alpha.corp

```
root@svr:/var/named                                    root@svr:/var/log/squid
[root@svr named]# named-checkzone  reverse reverse.alpha.corp
zone reverse/IN: loaded serial 0
OK
[root@svr named]#
```

Output → “loaded serial 0”

Enable, start, restart & status of named

- #systemctl enable named
- #systemctl restart named
- #systemctl status named

```
root@svr:/var/named                                    root@svr:/var/log/squid
[root@svr named]# systemctl enable named
[root@svr named]# systemctl restart named
[root@svr named]# systemctl status named
named.service - Berkeley Internet Name Domain (DNS)
   Loaded: loaded (/usr/lib/systemd/system/named.service; enabled)
   Active: active (running) since Sat 2020-04-18 12:37:27 IST; 5s ago
  Process: 23190 ExecStop=/bin/sh -c /usr/sbin/rndc stop > /dev/null 2>&1 || /bin/kill
-TERM $MAINPID (code=exited, status=0/SUCCESS)
  Process: 23202 ExecStart=/usr/sbin/named -u named $OPTIONS (code=exited, status=0/SUC
CESS)
  Process: 23200 ExecStartPre=/usr/sbin/named-checkconf -z /etc/named.conf (code=exited
, status=0/SUCCESS)
 Main PID: 23204 (named)
   CGroup: /system.slice/named.service
           └─23204 /usr/sbin/named -u named
```

From server machine:

- #ping svr.alpha.corp → pinging
- #ping cli.alpha.corp → pinging

Switch to client machine:

vim /etc/resolv.conf

```
# Generated by NetworkManager
search alpha.corp
nameserver 192.168.10.10
```

:wq!

Restart & status the network & NetworkManager services

#systemctl restart network

#systemctl restart NetworkManager

#systemctl status network

#systemctl status networkmanager

```
File  Edit  View  Search  Terminal  Help
[root@cli ~]# ping svr.alpha.corp
PING svr.alpha.corp (192.168.10.10) 56(84) bytes of data.
64 bytes from 192.168.10.10 (192.168.10.10): icmp_seq=1 ttl=64 time=0.210 ms
64 bytes from 192.168.10.10 (192.168.10.10): icmp_seq=2 ttl=64 time=0.451 ms
64 bytes from 192.168.10.10 (192.168.10.10): icmp_seq=3 ttl=64 time=0.629 ms
```

To verify the DNS name:

#dig <domain-name> or #dig alpha.corp

```
[root@cli ~]# dig alpha.corp

; <<>> DiG 9.11.4-P2-RedHat-9.11.4-9.P2.el7 <<>> alpha.corp
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54206
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;alpha.corp.                    IN      A

;; ANSWER SECTION:
alpha.corp.             86400   IN      A       192.168.10.10

;; AUTHORITY SECTION:
root@svr:~   Settings                                                    1 / 4
To direct input to this VM, click inside or press Ctrl+G.
                                                                ENG  12:53 PM  18-04-2020
```

#########END OF DNS/BIND server configuration ############

Mail server:

- Linux: **for sending** (Postfix)**, for receiving** (dovecot), GUI-web (squirrel mail or thunderbird)

Components of a mail server:

- MTA
  a. Mail transfer agent
  b. Send/receive email to & from other systems using SMTP
- MUA
  a. Mail user agent
  b. Email client
  c. Outlook, thunderbird, lotus, Squirrel mail
- MDA
  a. Mail delivery agent
  b. Delivery of the mail

For configuring mail server:

- Postfix
  - For sending an email
  - SMTP protocol
  - Already in DVD
- Dovecot
  - For receiving emails
  - POP3/IMAP protocol
  - Already in DVD
- Squirrel mail / ThunderBird
  - MUA
  - Download it from internet (URL: https://squirrelmail.org/download.php)

Lab setup:

- Mail server: svr.alpha.corp
- Domain name: alpha.corp
- IP address: 192.168.10.10/24
- DNS is required

On server (svr.alpha.corp)

- In the DNS forward lookup zone file (MX), add below command at line 8
  #svr IN MX 10 svr.alpha.corp.

```
$TTL 1D
@       IN SOA  @ svr.alpha.corp. (
                                        0       ; serial
                                        1D      ; refresh
                                        1H      ; retry
                                        1W      ; expire
                                        3H )    ; minimum
svr     IN      MX      10      svr.alpha.corp.
@       IN      NS              svr.alpha.corp.
@       IN      A               192.168.10.10
svr     IN      A               192.168.10.10
cli     IN      A               192.168.10.11
member  IN      A               192.168.10.12
```

:wq!

Restart & status of 'named' service:

- #systemctl restart named
- #systemctl status named → active running, proceed further

Above setup will only if DNS is present. Else edit /etc/hosts file to proceed further.

POSTFIX server config starts

- Install required packages (in DVD)
  #yum install -y postfix
- Configure postfix configuration file:
  #vim /etc/postfix/main.cf
  line75 (uncomment & write this)→ myhostname = svr.alpha.corp
  line83 (uncomment & write this)→ mydomain = alpha.corp
  line99 *uncomment*
  line113 *uncomment*
  line116 *comment it.*
  line164 (appendat last) mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
  line264 mynetworks = 192.168.10.0/24, 127.0.0.0/8
  line419 (uncomment) home_mailbox = Maildir/
  :wq!
- Enable, start & status postfix:
  #systemctl enable postfix
  #systemctl start postfix
  # systemctl status postfix

Verify the postfix configuration:

#systemctl enable telnet.socket
#systemctl start telnet.socket
#systemctl status telnet.socket → active (listening)

Communicate using telnet service:

```
[root@svr ~]# telnet localhost smtp
Trying ::1...
Connected to localhost.
Escape character is '^]'.
220 svr.alpha.corp ESMTP Postfix
ehlo localhost
250-svr.alpha.corp
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
```

Sending email from THOR to LOKI:

```
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
mail from:<thor>
250 2.1.0 Ok
rcpt to:<loki>
250 2.1.5 Ok
data
354 End data with <CR><LF>.<CR><LF>
Hello Loki,

How's Asguard?

Hulk sends his LOVE.

TR
THOR, Odin's Son
.
250 2.0.0 Ok: queued as 8265822FE076
quit
221 2.0.0 Bye
Connection closed by foreign host.
[root@svr ~]#
```

Switch to user's home directory:

```
[root@svr ~]# su - loki
[loki@svr ~]$ ls
Maildir
[loki@svr ~]$ ls Maildir/
cur  new  tmp
[loki@svr ~]$ ls Maildir/new/
1587203344.Vfd00I20063e3M497879.svr.alpha.corp
[loki@svr ~]$ cat Maildir/new/
[loki@svr ~]$ cat Maildir/new/1587203344.Vfd00I20063e3M497879.svr.alpha.corp
Return-Path: <thor@alpha.corp>
X-Original-To: loki
Delivered-To: loki@alpha.corp
Received: from localhost (localhost [IPv6:::1])
        by svr.alpha.corp (Postfix) with ESMTP id 8265822FE076
        for <loki>; Sat, 18 Apr 2020 15:16:44 +0530 (IST)
Message-Id: <20200418094721.8265822FE076@svr.alpha.corp>
Date: Sat, 18 Apr 2020 15:16:44 +0530 (IST)
From: thor@alpha.corp

Hello Loki,

How's Asguard?

Hulk sends his LOVE.

TR
THOR, Odin's Son
[loki@svr ~]$
```

POSTFIX CONFIGURATION IS OVER

CONFIGURING DOVECOT

- Install dovecot server
  #yum install -y dovecot
- To verify
  #rpm -q dovecot

Edit the config file for dovecot server.

- Edit main file:
  # vim /etc/dovecot/dovecot.conf
  Uncomment line 24,
  protocols = imap pop3 lmtp
  :wq!
- Edit supportive files in dovecot:
  a. # vim /etc/dovecot/conf.d/10-mail.conf
  Uncomment line 24, to download the mails
  mail_location = maildir:~/Maildir
  b. #vim /etc/dovecot/conf.d/10-auth.conf
  Uncomment line 10
  disable_plaintext_auth = yes
  on line 100, append 'login'
  auth_mechanisms = plain login
  :wq!
  c. #vim /etc/dovecot/conf.d/10-master.conf
  Line 91 & 92, uncomment & add "postfix" to user and group:
  user = postfix
  group = postfix
  :wq!

Enable, start & status of dovecot service:

- #systemctl enable dovecot
- #systemctl start dovecot
- #systemctl status dovecot → active (running)

Verify dovecot configuration:

telnet localhost pop3

```
[root@centos7 ~]# telnet localhost pop3
Trying ::1...
Connected to localhost.
Escape character is '^]'.
+OK Dovecot ready.
user loki          → login to user "loki"
+OK
pass loki          → password for user loki
+OK Logged in.
list               → list the mail present
+OK 1 messages:
1 389
.
retr 1             → retrieve <mail-number>
```

```
+OK 389 octets
Return-Path: thor@alpha.corp
X-Original-To: loki
Delivered-To: loki@alpha.corp
Received: from localhost (localhost [IPv6:::1])
        by svr.alpha.corp (Postfix) with ESMTP id 021A322FE076
        for <loki>; Mon, 20 Apr 2020 00:48:39 +0530 (IST)
Message-Id: <20200419191847.021A322FE076@svr.alpha.corp>
Date: Mon, 20 Apr 2020 00:48:39 +0530 (IST)
From: thor@alpha.corp

Hahaha                  → mail body
Hahahahah               → mail body
.
quit                     → quit the console
+OK Logging out.
Connection closed by foreign host.
[root@centos7 ~]#
```

Example:

```
[root@centos7 ~]#
[root@centos7 ~]# telnet localhost pop3
Trying ::1...
Connected to localhost.
Escape character is '^]'.
+OK Dovecot ready.
user loki
+OK
pass loki
+OK Logged in.
list
+OK 1 messages:
1 389
.
1
-ERR Unknown command: 1
retr 1
+OK 389 octets
Return-Path: <thor@alpha.corp>
```

```
retr 1
+OK 389 octets
Return-Path: <thor@alpha.corp>
X-Original-To: loki
Delivered-To: loki@alpha.corp
Received: from localhost (localhost [IPv6:::1])
        by svr.alpha.corp (Postfix) with ESMTP id 021A322FE076
        for <loki>; Mon, 20 Apr 2020 00:48:39 +0530 (IST)
Message-Id: <20200419191847.021A322FE076@svr.alpha.corp>
Date: Mon, 20 Apr 2020 00:48:39 +0530 (IST)
From: thor@alpha.corp

hahaha
hahahahah
.
quit
+OK Logging out.
Connection closed by foreign host.
[root@centos7 ~]#
```

Squirrel mail to send, receive and read mail in linux system:

- Install squirrelmail, apache (httpd) & PHP packages.
  #yum install -y httpd
  #yum install -y php

- Install epel in centos to download & install squirrel mail:
  #yum -y install epel-release
- Install squirrel mail:
  #yum install -y squirrelmail
- Switch to squirrelmail config directory
  #cd /usr/share/squirrelmail/config
  #ls → conf.pl

```
[root@centos7 ~]# cd /usr/share/squirrelmail/config/
[root@centos7 config]# ls
config_default.php  config_local.php  config.php  conf.pl  index.php
[root@centos7 config]#
```

To execute the conf.pl file,
#./conf.pl
In the main menu, change below settings:

- 1. Organization Preferences
- 2. Server Settings
- 7. Message of the Day (MOTD)

Save & quit the console.

Edit httpd file for squirrel mail:

- # vim /etc/httpd/conf/httpd.conf
  Go to the bottom of the file & append below content:

```
Alias /webmail /usr/share/squirrelmail
<Directory /usr/share/squirrelmail>
                Options indexes FollowSymLinks
                RewriteEngine   On
                AllowOverride   All
                DirectoryIndex index.php
                Order   allow,deny
                Allow from all
</Directory>
```

```
#EnableMMAP off
EnableSendfile on

Alias /webmail /usr/share/squirrelmail
<Directory /usr/share/squirrelmail>
        Options indexes FollowSymLinks
        RewriteEngine   On
        AllowOverride   All
        DirectoryIndex index.php
        Order   allow,deny
        Allow from all
</Directory>
```

:wq!

Restart "httpd" & "dovecot" service & view the status

Launch firefox web browser & type below path

- http://<IP-Address>/webmail

we installed THUNDERBIRD for sending and receiving emails:

steps:

- thunderbird-68.7.0.tar.bz2
- uncompress it:
    - #bzip2 thunderbird-68.7.0.tar.bz2
- output -> thunderbird-68.7.0.tar
- extract the abv:
    - #tar -xvf thunderbird-68.7.0.tar
- output -> thunderbird--> directory

Single system env:
---
name: svr.lti.com
domain: lti.com
DNS is running
install
- postfix
- dovecot
- squirrelmail or thunderbird
users on same system
- user1
- user2

==========MAIL SERVER CONFIG OVER==========

Recommendation for LAB setup:

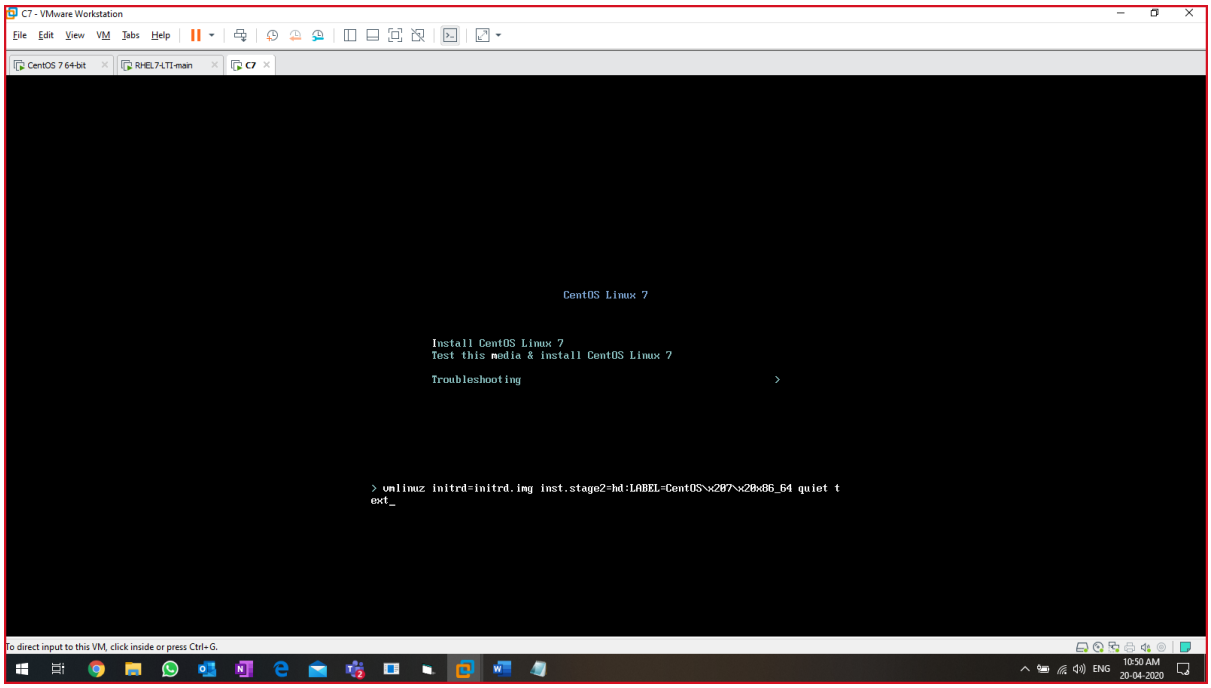
sys-1 (2 LAN cards- 1 LAN segment, 1 NAT)

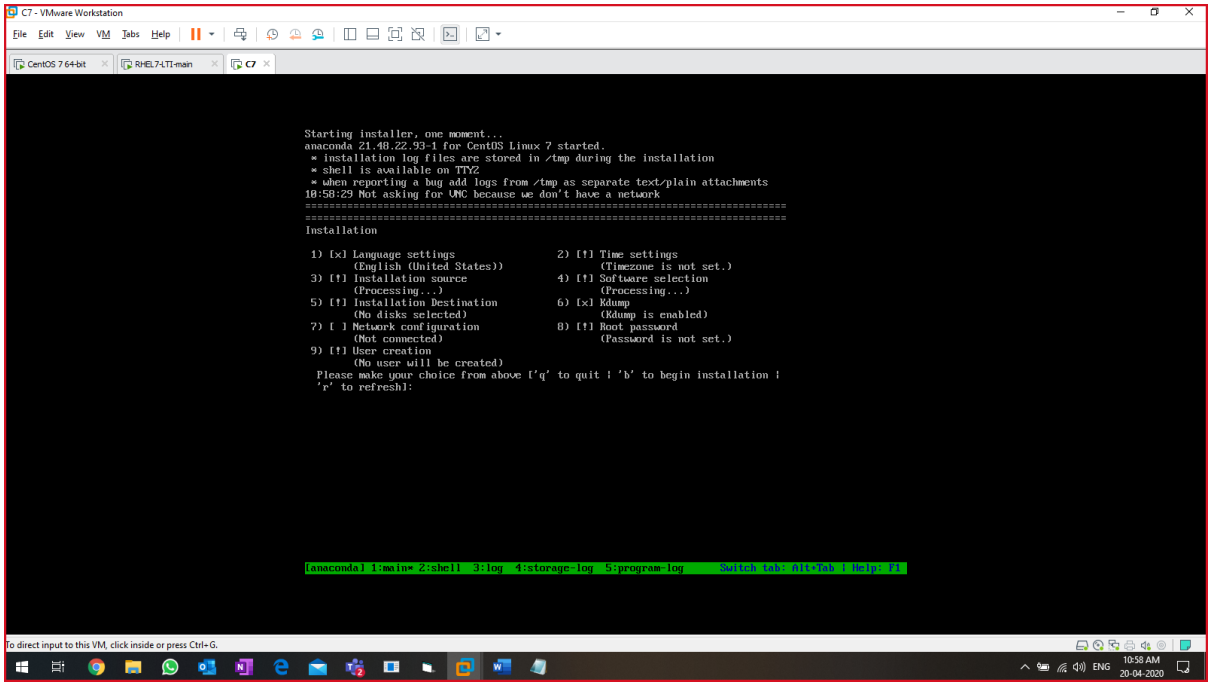
- GUI/text installation
- file, user, service mgmt
- RPM / YUM (full and shortcut)
- partitioning (attach 3GB disks)
    - Simple
    - LVM
    - RAID
- Networking
- Firewall
- NTP + internet
- FTP
- SSH
- telnet
- SCP
- rsync
- apache web server
- DNS
- MAIL server (postfix, dovecot, (thunderbird or squirrel mail)

sys-2 (2 LAN cards- 1 LAN segment, 1 NAT)

- DHCP
- SCP (win to linux with NAT)
- rsync
- NFS (l-l)
- SAMBA (linux-windows)
- Squid proxy

Text based installation:





URL: http://www.geekpills.com/linux_tutorial/rhel7centos7-text-installation

Convert GUI to CLI permanently

- # systemctl set-default multi-user.target & reboot

Convert CLI to GUI permanently

- # systemctl set-default graphical.target & reboot

FSCK:

- File system consistency check
- Use cases
    - System failure at boot time
    - File System is corrupted.
    - External devices

Scan HDD without reboot:

- List all available device:
    - #ls -l /sys/class/scsi_host/
- search for the device having keyword “mpt” in it & check the “host<value>”
    - #grep mpt /sys/class/scsi_host/host??/proc_name
- reset the values for the host<value> file
    - #echo “- - -“ > /sys/class/scsi_host/host32/scan
- Check using the commands
    - “disks” GUI
    - #lsblk

FSCK = RHEL 6,

Xfs_repair = RHEL 7

At boot time, file check:

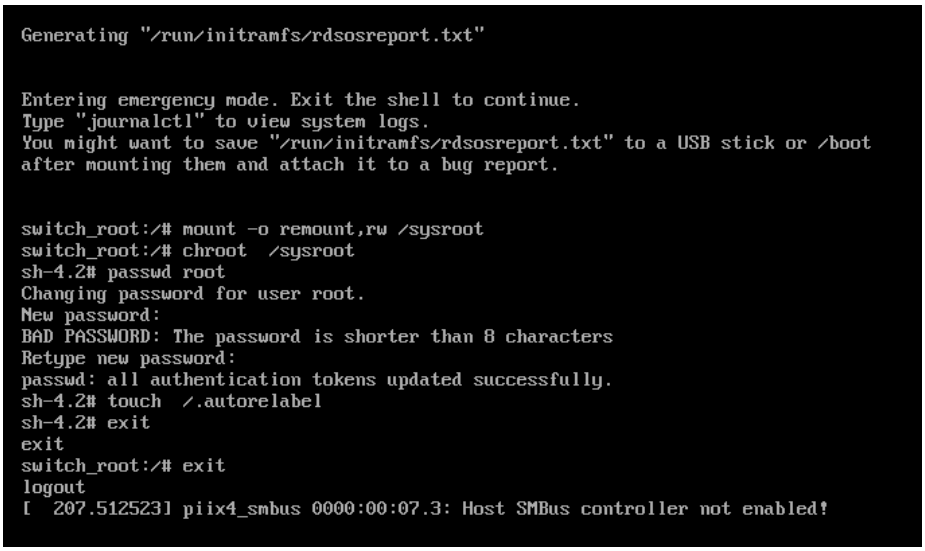
- 0 = dont scan at boot time
- 1 = scan at boot time -> /
- 2 = scan at boot time -> /boot, /usr, /home

Redirection:

- Redirecting the output in a certain file/location
- 3 std
    - Standard input → ‘0’ → keyboard, file…
        - # cat < lti-history.txt
    - Standard output → ‘1’
        - #cat >> lti-history.txt
    - Standard error → ‘2’
        - #ld 2> error.txt
- To send success messages/outputs in one file & errors in another file:
    - # ls -l f1 f2 lti-history.txt pkgs.txt d1 d2 1>success.txt 2>error.txt
- Send the output to null zone
    - #ld 1> success.txt 2>/dev/null

Reset root password in RHEL 7:

- Reboot the linux system
- Stop the booting process at GRUB screen.
- Select the linux kernel line & press “e”.
- Search for the line that starts with “Linux 16” and on the same line append “rd.break”
- Press CTRL+X



Get system information:

- #cat /proc/meminfo → list memory information
- #free → list memory & swap
- #free -k → list memory info in KBs
- #free -m → list memory info in MBs
- #free -g → list memory info in GBs
- #cat /proc/cpuinfo → list CPU info (file)
- #lscpu → list CPU info (command)

[Task: increase the SWAP space to twice the current SWAP space size]

```
[root@cli Desktop]# free -m
              total        used        free      shared     buffers      cached
Mem:           1826        1040         786          10           0         407
-/+ buffers/cache:         631        1194
Swap:          4999           0        4999
[root@cli Desktop]#
```

Powershell

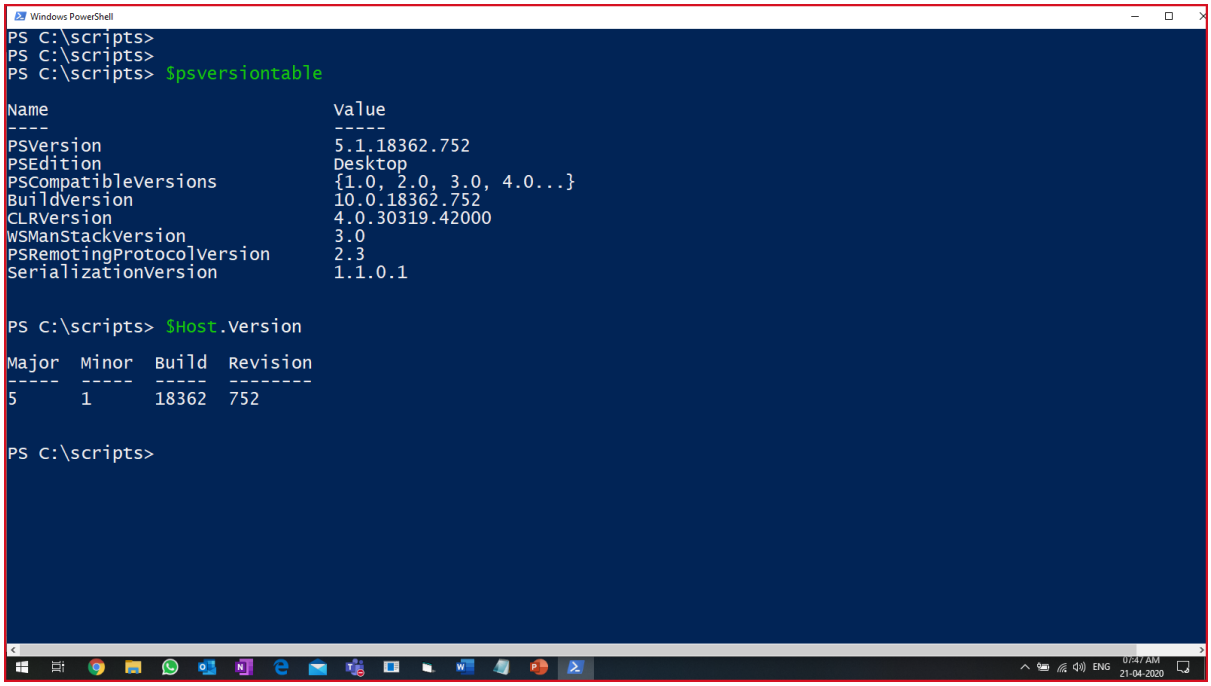
Lab setup:

- One domain controller – **DC**
- Two members – **svr**, **member**

Powershell:

- Command-line utility
- **Administrative tasks**
    - Manage local system
    - Manage remote system
        - Domain-joined systems
        - Non-domain joined systems
    - Report of users, drives, spaces…
- Microsoft product, which does not have any GLOBAL EXAMINATION.
    - 10961 – basics of powershell
    - 10962 – advanced powershell
- CMD prompt commands can be executed on powershell.

Powershell version:



Powershell commands:

- Cmdlets
- Verb-noun

To update PS:

- Update-help -force

To check date:

- Get-date or
- date

to check the list of cmdlets, present in PS:

- get-command or
- gcm

to count the number of cmdlets:

- get-command | measure
- gcm | measure

to create a new alias

- new-alias -name <new-name-u-want> -value <original-cmd>
- Ex: new-alias -name d -value get-date

To create a directory in PS:

- mkdir <directory-name-with-path> or
- new-item -type directory -path <path-with-name-of-directory>
- Ex: New-Item -Type Directory -Path C:\lti-ps\testDir

To change the directory:

- cd <full-path-to-the-location> //Linux or CMS
- Set-Location C:\lti-ps\ //powershell

To list cmdlets with a specific keyword ('item')

- get-help *item* → search for key in start, middle or end of the cmdlet
- Get-Command -Noun item → search for the key END's with the 'item'

List top 10 processes which are consuming higher CPU:

Listing service:

- Get-Service -Name "T*" | Where-Object Status -eq "running"
- Get-Service | Where-Object {($_.Status -eq 'running') -and ($_.Name -like 'T*')}

```
#list running processes & send it to a text file
Get-Service
| Where-Object Status -EQ "running" `
| Select-Object DisplayName `
| Out-File running-servvvv.txt

#display the file
notepad.exe running-servvvv.txt

#[Method-1:] to list services starting with "T" & status is running
Get-Service -Name "T*" | Where-Object Status -eq "running"

#[Method-2:]to list services starting with "T" & status is running
Get-Service | Where-Object {($_.Status -eq 'running') -and ($_.Name -like 'T*')}
`
| select Name

#to convert it into different outputs
Get-Process | Out-File pro1.txt
.\pro1.txt

#to convert it into CSV format
Get-Process | ConvertTo-Csv | Out-File pro1.csv
.\pro1.csv
```

```
#to convert it into csv + list 10 lines
Get-Process | select -First 10 `
|Select-Object processname, cpu `
| ConvertTo-Csv -NoTypeInformation `
| Out-File pro1.csv
.\pro1.csv

#to convert it into HTML append the data using "-append" parameter
Get-Process `
| select -First 10 `
|Select-Object processname, cpu `
| ConvertTo-Html | Out-File pro1.html -Append
.\pro1.html
```

Execution policy:

- Run a script in powershell
- Do u have permissions to run a script.
- Locally, remotely
- Policy:
    - **Restricted**
        - Locally, remotely NO script is allowed to get executed.
    - **Unrestricted**
        - Locally, remotely ALL script is allowed to get executed.
    - **All-signed**
        - Locally, remotely ALL script must be digitally signed.
    - **Remote-signed**
        - Locally no digital signature but remote script must be digitally signed.
- Cmdlets:
    - Get-executionpolicy
    - Set-executionpolicy <Policy-name> → admin mode only

To create a self-sign certificate:

a. Create certificate using below cmdlet:

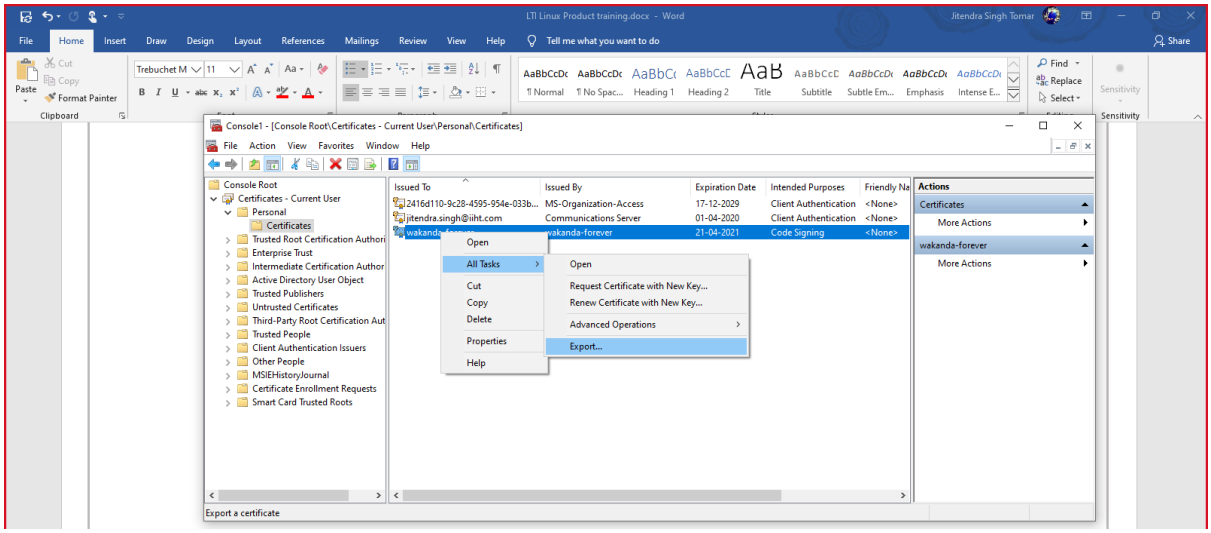
```
New-SelfSignedCertificate -CertStoreLocation cert:\currentuser\my `
-Subject "CN=wakanda-forever" `
-KeyAlgorithm RSA `
-KeyLength 2048 `
-Provider "Microsoft Enhanced RSA and AES Cryptographic Provider" `
-KeyExportPolicy Exportable `
-KeyUsage DigitalSignature `
-Type CodeSigningCert
```

To verify, either of these can be executed.

- ls Cert:\CurrentUser\My\

- Get-ChildItem Cert:\CurrentUser\My\

b. Export the certificate.
    - Open MMC using run prompt
    - Add certificate snap-in
    - Export the certificate:

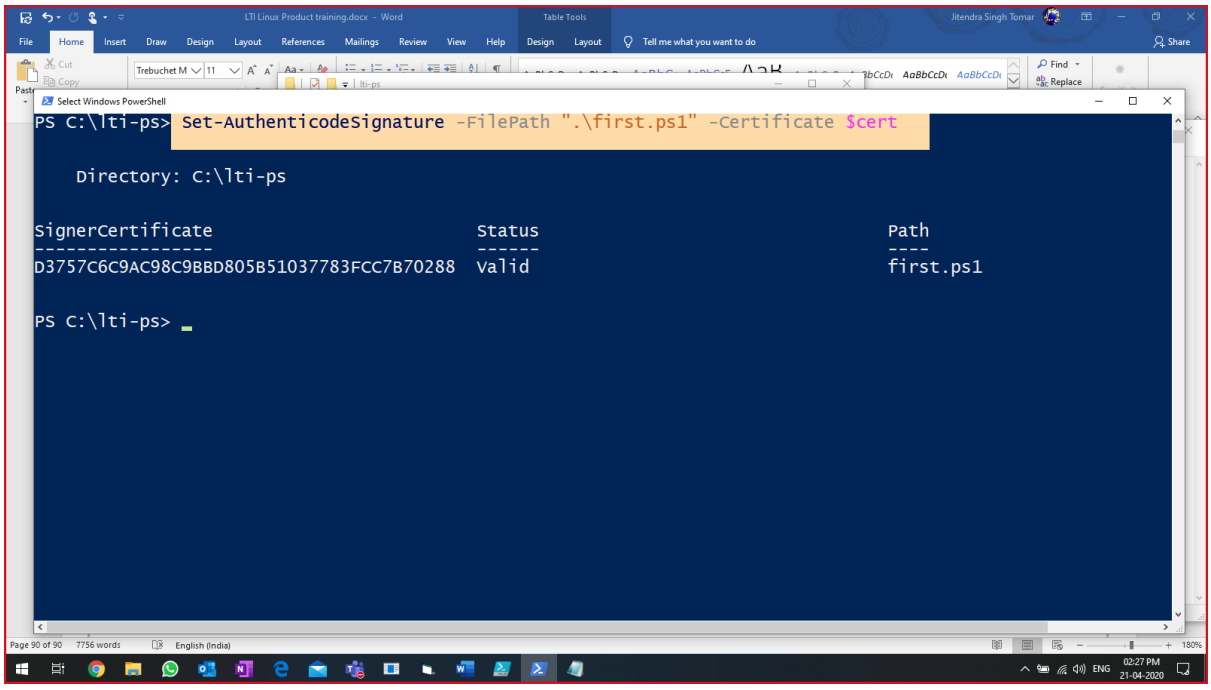


c. Install/Importing the exported certificate.
   - Double click on the certificate
   - Install it in TRUSTED ROOT CERTIFICATE AUTHORITY.

d. Digitally sign the script.

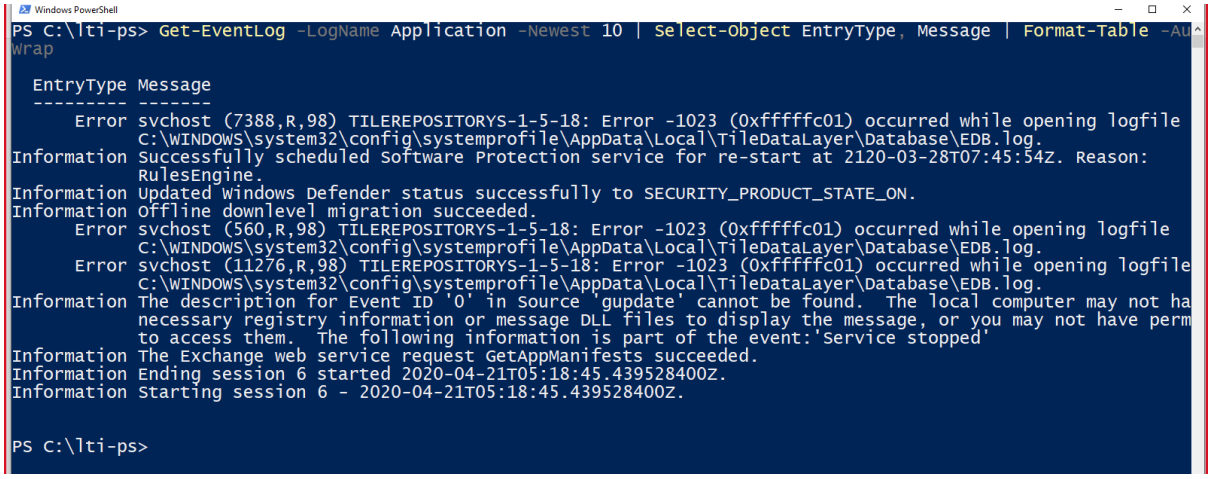
```
PS C:\lti-ps> $cert = Get-ChildItem
Cert:\CurrentUser\My\D3757C6C9AC98C9BBD805B51037783FCC7B70288

PS C:\lti-ps> $cert
PSParentPath: Microsoft.PowerShell.Security\Certificate::CurrentUser\My

          Thumbprint                                Subject
          ----------                                -------
D3757C6C9AC98C9BBD805B51037783FCC7B70288  CN=wakanda-forever
```

e. Bind the certificate



To read 10 newest application event logs in local system:

PS C:\lti-ps> Get-EventLog -LogName Application -Newest 10 | Select-Object EntryType, Message | Format-Table -AutoSize -Wrap



Basic commands:

```
pwd
Set-Location C:\lti-ps
Get-ChildItem
Move-Item -Path .\1st-get-date.ps1 -Destination .\testDir
Copy-Item -Path C:\lti-ps\*.ps1  -Destination C:\lti-ps\testDir
```

Task:

a. Create a script, which when clicked opens Outlook, notepad, Google Chrome.
b. List the logical disks information in GBs

WMI objects

- CIM
  - Common information model.
  - A standard given by distributed mgmt task force (DMTF)
  - Provides a common definition of managing network, systems, application, service etc…
- WMI
  - Windows management instrumentation
  - WMI is Microsoft implementation of CIM.
  - CIM + hardware + software
  - Everything is based on CLASSES
  - Cmdlet: get-wmiobject

Cmdlets:

- Get-WmiObject -Class win32_operatingsystem
- Get-WmiObject -Class win32_bios
- Get-WmiObject -Class win32_battery
- Get-WmiObject -Class win32_logicaldisk → task (b)

To get top 10 highest CPU utilization services:

```
Get-Process `
| select -Unique `
| Select-Object Processname, CPU `
| Sort-Object CPU -Descending `
| select -First 10
```

Mathematical operations:

```
[math]::Sqrt(25)
[math]::PI
[math]::Round(18.26)
[math]::Floor()
[math]::Ceiling()
```

Hash table:

@{key=value; key1=value1;key2=value2…}

@{label= “CPU(in2digits)” ; exp= “”} OR @{l= “CPU(in2digits)” ; e= “”}

To get top 10 processes in round off format:

```
Get-Process `
| select -Unique `
| Select-Object Processname,@{l="CPU";e={[math]::round( ($_.CPU),2) }} `
| Sort-Object CPU -Descending `
| select -First 10
```

Microsoft documentation: https://docs.microsoft.com/en-us/powershell/scripting/getting-started/getting-started-with-windows-powershell?view=powershell-5.1

Task: display below values:

Name : "OS-name"
IP address : <IP>
C Drive size : <val> Total in GBs
C Drive size : <val> free in GBs
domain name : <domain-name>

Operators:

- = → '-eq'
- != → '-ne'
- '>' → '-gt'
- '<' → '-lt'
- '>=' → '-ge'
- '<=' → '-le'

Task: to fetch the name of a service from user, display the status (running or stopped) & then ask user to start the service if it is stopped:

```
# to list the status of a service,
# and ask user to start the service
cls

#fetch the service name from user
$name = Read-Host "Enter service name"

#store the service info in a variable
$svc = Get-Service -Name $name

#store service NAME in a variable
$n = $svc.name

#store service status in a variable
$s = $svc.Status

#display all together
write-host "Service name $n is $s"

if($s -eq "stopped"){
    $ans = read-host "Do u want to start the service (Y | N)"
    if(($ans -eq "y") -or ($ans -eq "yes")){
        Get-Service -Name $name | start-service
    }else{
        Write-Host "Fine, if u dont want to start the service"
    }
}
else{
       Write-Host "service already running"
}
```

Interactive script to start a service & display its final result:

```
# to list the status of a service,
# and ask user to start the service
cls

#fetch the service name from user
$name = Read-Host "Enter service name"

#store the service info in a variable
$svc = Get-Service -Name $name

#store service NAME in a variable
$n = $svc.name

#store service status in a variable
$s = $svc.Status

#display all together
write-host "Service name $n is $s"

if($s -eq "stopped"){
    $ans = read-host "Do u want to start the service (Y | N)"
    if(($ans -eq "y") -or ($ans -eq "yes")){
        Get-Service -Name $name | start-service
        $cs = Get-Service -Name $name
        if($cs.Status -eq "running"){
           Write-Host "Service is running"
        }else{
           Write-Host "still not running"
        }
    }else{
        Write-Host "Fine, if u dont want to start the service"
    }
}
else{
       Write-Host "service already running"
}
```

Create a “Microsoft.PowerShell_profile.ps1” file, which gets loaded every time you start the terminal.

New-Item -ItemType File $PROFILE -Force → PS console

- “New-item” = create something (file or directory)
- “-itemtype file” = creating a text file
- “$PROFILE” = below value is stored in this variable [C:\Users\Jeetu\Documents\WindowsPowerShell\Microsoft.PowerShell_profile.ps1]
- “-Force” = create FORCEFULLY.

For PS ISE

```
➢  $profile
➢  C:\Users\Jeetu\Documents\WindowsPowerShell\Microsoft.PowerShellISE_profile.ps1
```

If the above file does not exist, then create it:

```
➢  New-Item -ItemType File $profile -Force
```

Syntax:

```
Function <function-name>
{
…
…
}
<function-name>
```

To get the service info using functions:

```
#declare the function
function get-svcstatus{
cls
$n = Read-Host "enter service name"
Get-Service -Name $n
}

#call the function
get-svcstatus
```

To display the status of a service using function & passing service name as parameter:

```
function get-svcstatus{
param($svc)
$ans = Get-Service -Name $svc
Write-Host "Service $svc is: "$ans.status
}

get-svcstatus -svc mpssvc
```

Create function and load it in the memory, when PS console is triggered/opened:

Create a script & store it in any place [*C:\lti-ps\testDir\function-1.ps1*]

```
function get-svcstatus{
    param($svc)
    $ans = Get-Service -Name $svc
    Write-Host "Service $svc is: "$ans.status
}
```

Call the function from the prompt by passing the parameter:

```
get-svcstatus -svc bits
```

if you want, the same script must be called every time you start the PS console:

- Notepad.exe $PROFILE
  Import-Module C:\lti-ps\testDir\function-1.ps1

To get logical disk information:

```
function get-diskinfo{
cls
Get-WmiObject -Class win32_logicaldisk `
| Select-Object DeviceID, DriveType,`
FreeSpace,
Size
}
get-diskinfo
```

Converting & displaying the disk info in GBs format:

```
function get-diskinfo{
cls
Get-WmiObject -Class win32_logicaldisk `
| Select-Object DeviceID, DriveType,`
@{l="Size";e={[math]::Round($_.size/1GB,2)}}, `
@{l="FreeSpace";e={[math]::Round($_.size/1GB,2)}}
}
get-diskinfo
```

Get the info of other devices in the network:

[method-1]

```
function get-diskinfo{
cls
Get-WmiObject -Class win32_logicaldisk -ComputerName cli `
| Select-Object DeviceID, DriveType,`
@{l="Size";e={[math]::Round($_.size/1GB,2)}}, `
@{l="FreeSpace";e={[math]::Round($_.size/1GB,2)}}
}
get-diskinfo
```

[method-2]

```
function get-diskinfo{
cls
Get-WmiObject -Class win32_logicaldisk -ComputerName $newcomp `
| Select-Object DeviceID, DriveType,`
@{l="Size";e={[math]::Round($_.size/1GB,2)}}, `
@{l="FreeSpace";e={[math]::Round($_.size/1GB,2)}}
}

#create a text file & write computernames in it
#notepad.exe comp.txt

#fetch the values in a variable
$newcomp = Get-Content .\comp.txt

#call the function
get-diskinfo
```

To get the logs for remote system:

```
Get-EventLog -LogName Application -Newest 5 -ComputerName dc `
| Select-Object EntryType, Message | ft -AutoSize -Wrap
```

To get the output in the colour mode:

```
$name = Get-Service -Name winrm
cls
if($name.Status -eq "Running"){
    Write-Host $name.name" is running" -ForegroundColor Black -BackgroundColor
Green
}else{
    Write-Host $name.name" is stopped" -ForegroundColor white -BackgroundColor
Red
}
```

TASK: display ALL SERVICES (running in green & stopped in red)

Sending variable values using pipeline in advance functions:

```
function get-eventdata{
[cmdletbinding()]
    param(
    [Parameter(Mandatory=$true,
    ValueFromPipeline=$true
    )]
    [string]$com
    )
    cls
    #Write-Host "$com function executed"
Get-EventLog -LogName Application -ComputerName $com -Newest 5
}
$com | get-eventdata
```

Active directory mgmt using PowerShell: