

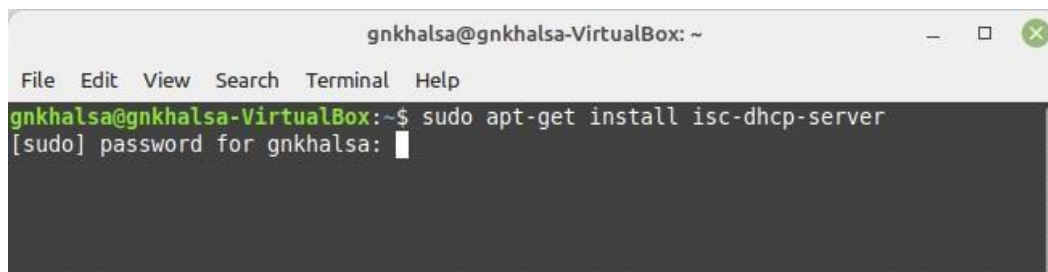
## Practical 1

### Installing DHCP server in Ubuntu 16.04

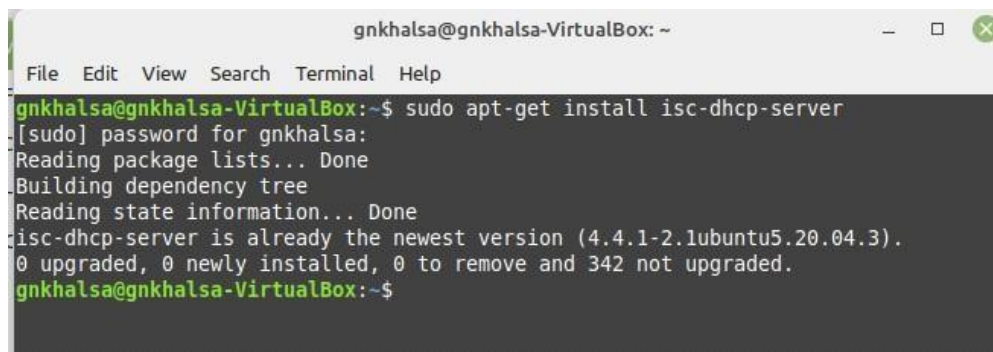
Steps to configure DHCP Server, Configure NFS Server to share directories on your network, Configure NFS Client (Ubuntu and Windows OS)

**Step1:** In your Ubuntu 16.04, open up a terminal and input the following command to install dhcp server.

**sudo apt-get install isc-dhcp-server**



```
gnkhalsa@gnkhalsa-VirtualBox: ~  
File Edit View Search Terminal Help  
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo apt-get install isc-dhcp-server  
[sudo] password for gnkhalsa:
```



```
gnkhalsa@gnkhalsa-VirtualBox: ~  
File Edit View Search Terminal Help  
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo apt-get install isc-dhcp-server  
[sudo] password for gnkhalsa:  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
isc-dhcp-server is already the newest version (4.4.1-2.1ubuntu5.20.04.3).  
0 upgraded, 0 newly installed, 0 to remove and 342 not upgraded.  
gnkhalsa@gnkhalsa-VirtualBox:~$
```

**Step2:** Once the installation has been done, make sure the network settings of your virtual machine are set to bridged network.

**Step3:** In the terminal, type `inconfig` to verify as to whether an ip address has been assigned to your virtual machine.

```
gnkhalsa@gnkhalsa-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::e1eb:1c6c:f15c:9bea prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:03:27:c3 txqueuelen 1000 (Ethernet)
    RX packets 534 bytes 559995 (559.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 326 bytes 31164 (31.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 182 bytes 16992 (16.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 182 bytes 16992 (16.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

gnkhalsa@gnkhalsa-VirtualBox:~$
```

**Step4:** Next we need to configure our installed dhcp server to it serve ip address to connecting clients. Follows the following configuration.

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo nano /etc/dhcp/dhcpd.conf
```

**Step5:** Look for the section which says "A slightly different configuration for internal subnet".

```
# A slightly different configuration for an internal subnet.
#subnet 10.5.5.0 netmask 255.255.255.224 {
#   range 10.5.5.26 10.5.5.30;
#   option domain-name-servers ns1.internal.example.org;
#   option domain-name "internal.example.org";
#   option subnet-mask 255.255.255.224;
#   option routers 10.5.5.1;
#   option broadcast-address 10.5.5.31;
#   default-lease-time 600;
#   max-lease-time 7200;
#}

# Hosts which require special configuration options can be listed in
# host statements. If no address is specified, the address will be
# allocated dynamically (if possible), but the host-specific information
# will still come from the host declaration.
```

**Step6:** Now start by config the subnet line. Set the first ip address to the start of your network range.(The ip address you received in the output of ifconfig be use dto calculated it. Here the ip address was 192.168.0.10 and hence network ip is 192.168.0.0).

**Step7:** Set the net mask to 255.255.255.0 (This can be done by pressing ctrl+\ which will open a replace prompt when you will have to type the original net mask which here was 255.255.255.224 and press enter then it will ask what it should be replaced with, then type 255.255.255.0, then press enter).

```
[ Replaced 4 occurrences ]
^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit      ^R Read File  ^\ Replace    ^U Paste Text ^T To Spell   ^_ Go To Line
```

**Step8:** In range set a range of ip address you would like server to serve. Here it is set to server 20 addresses ranging from 10.5.5.10 to 10.5.5.30 (For changing the range do the same by replacing the whole range section by typing "range 10.5.5.10 10.5.5.30" in replace it with prompt)

**Note:** The range for me was 10.5.5.26 10.5.5.30, for your pc/system it might be different so change accordingly.

```
# A slightly different configuration for an internal subnet.
#subnet 10.5.5.0 netmask 255.255.255.0 {
#   range 10.5.5.26 10.5.5.30;
#   option domain-name-servers ns1.internal.example.org;
#   option domain-name "internal.example.org";
#   option subnet-mask 255.255.255.0;
#   option routers 10.5.5.1;
#   option broadcast-address 10.5.5.31;
#   default-lease-time 600;
#   max-lease-time 7200;
#}
```

**Step9:** Configure the routes line to be the default gateway.

**Step10:** Save the file by exiting it "ctrl+X", then in the prompt enter y and press enter or just press enter.

**Step11:** Now we have installed and configured our dhcp server. Let's start our dhcp server by using the following command.

**sudo /etc/init.d/isc-dhcp-server start**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo /etc/init.d/isc-dhcp-server start
[sudo] password for gnkhalsa:
Starting isc-dhcp-server (via systemctl): isc-dhcp-server.service.
gnkhalsa@gnkhalsa-VirtualBox:~$
```

**Step12:** To cross verify that help the ip address is actually served from the dhcp server go back to Ubuntu where the dhcp server has been configured and type in **cat /var/lib/dhcp/dhcpd.leases**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ cat /var/lib/dhcp/dhcpd.leases
# The format of this file is documented in the dhcpd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.4.1

# authoring-byte-order entry is generated, DO NOT DELETE
authoring-byte-order little-endian;

gnkhalsa@gnkhalsa-VirtualBox:~$
```

## Practical 2

**Aim :-** Configure NFS Server to share the directories on your Network

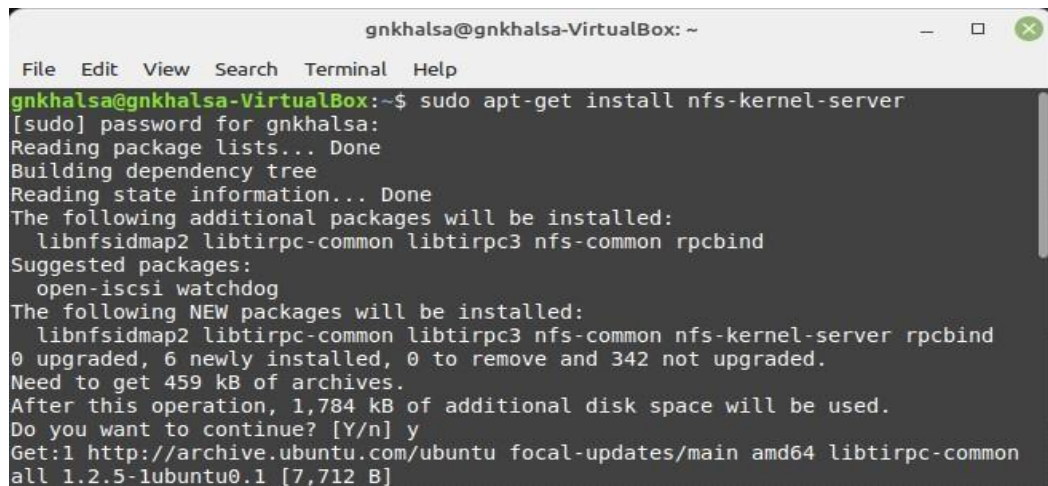
Configure NFS Server on your machine.

We skipped it because it's not necessary and the first 3 commands are same as of the execution on Ubuntu machine.

Configure NFS Server on your Ubuntu.

**Step 1:** Install the NFS server on your Ubuntu 16.04 using the following command.

**sudo apt-get install nfs-kernel-server**

A terminal window titled 'gnkhalsa@gnkhalsa-VirtualBox: ~' showing the command 'sudo apt-get install nfs-kernel-server'. The output shows the package lists being read, the dependency tree being built, and the state information being read. It lists additional packages to be installed: libnfsidmap2, libtirpc-common, libtirpc3, nfs-common, and rpcbind. It also lists suggested packages: open-iscsi and watchdog. The terminal shows that 6 new packages will be installed, requiring 459 kB of archives and 1,784 kB of additional disk space. The user is prompted to continue, and the installation proceeds from the Ubuntu archive.

**Step 2:** Once the installation is done we have to change the domain name of our domain which will be as same as of our root machine.

to change the domain name we will use the following command to get into that file.

**sudo nano /etc/idmapd.conf**

A terminal window showing the command 'sudo nano /etc/idmapd.conf' being executed.

After pressing enter you will uncomment the line which says

#Domain=localdomain and edit the domain name.

Here is how you edit this vi/vim file.

Press 'i' so that you could go into insert mode after that if you use A,B,C,D will get inserted but to avoid that we will press delete after pressing the up arrow, then locate to the line #Domain = localdomain, uncomment it by using del and also change the name by deleting the default name (which has to be done by delete) and then rename it to 'srv.,world'

After renaming the domain press 'esc' to get back into command line form and to save and exit press ":" then type "wq!" and press enter to exit. (:+ wq! + enter)

Be careful while you do these steps because one mistake and you might corrupt your idmapd.conf file.

```

gnkhalsa@gnkhalsa-VirtualBox: ~
File Edit View Search Terminal Help
GNU nano 4.8 /etc/idmapd.conf Modified

[General]
Verbosity = 0
Pipefs-Directory = /run/rpc_pipefs
# set your own domain here, if it differs from FQDN minus hostname
# Domain = srv.world

[Mapping]
Nobody-User = nobody
Nobody-Group = nogroup

[ Replaced 1 occurrence ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go To Line

```

**Step 3:** Now open the exports file to check if it's there or not. It is basically a list for file systems which might be exported via our NFS server.

**sudo nano /etc/exports**

```

gnkhalsa@gnkhalsa-VirtualBox:~$ sudo vi /etc/exports
gnkhalsa@gnkhalsa-VirtualBox: ~
File Edit View Search Terminal Help
# /etc/exports: the access control list for filesystems which may be exported
# to NFS clients. See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4 gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
"/etc/exports" 10 lines, 389 characters

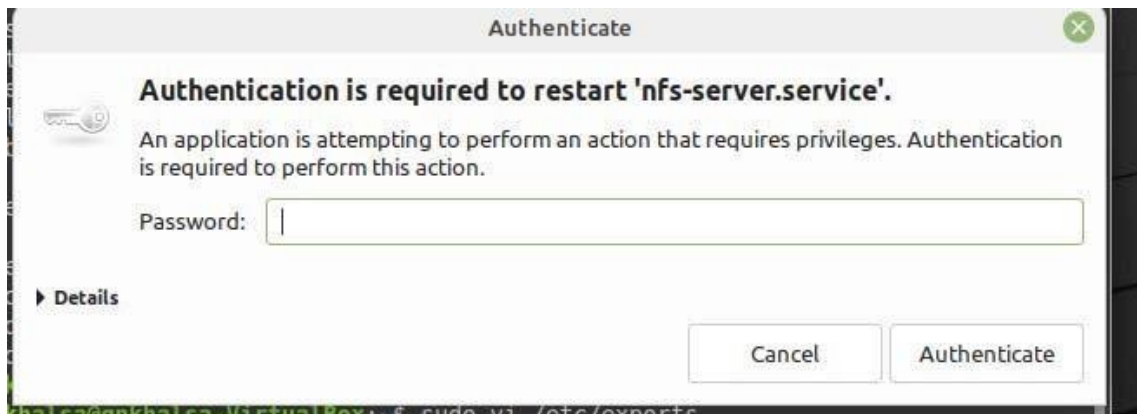
```

**Step 4:** Now restart the NFS server you've installed in your Ubuntu at the start of this practical using the following command .

**systemctl restart nfs-server**



```
gnkhalsa@gnkhalsa-VirtualBox:~$ systemctl restart nfs-server
```



**Step 5:** In this step we will mount our nfs server into our home directory using the following command.

```
mount -t nfsdlp.srv.world:/home/home
```

```
gnkhalsa@gnkhalsa-VirtualBox:~$ mount -t nfsdlp.srv.world:/home/home
```

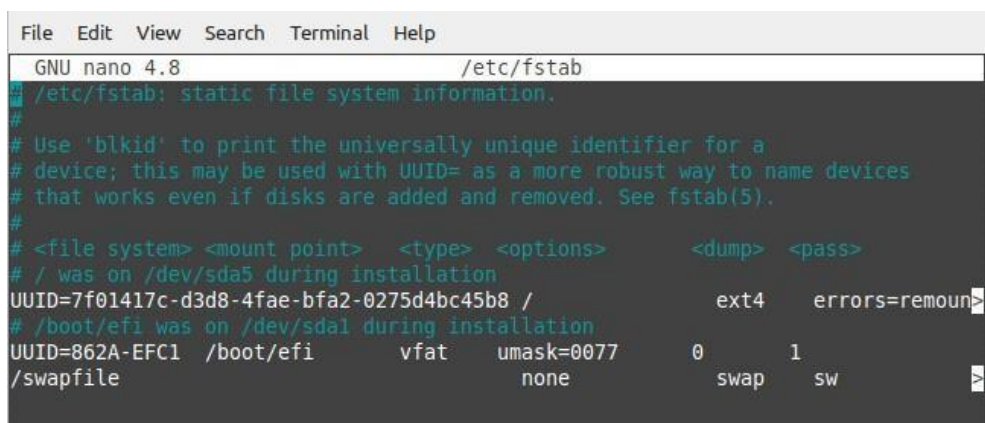
**Step 6:** In this step we will display the information mounted by our NFS server, the 'df' command is used to display the information about total space and available space on a file system.

```
gnkhalsa@gnkhalsa-VirtualBox:~$ df -hT
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
udev	devtmpfs	1.9G	0	1.9G	0%	/dev
tmpfs	tmpfs	394M	1.2M	393M	1%	/run
/dev/sda5	ext4	78G	8.7G	66G	12%	/
tmpfs	tmpfs	2.0G	0	2.0G	0%	/dev/shm
tmpfs	tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
tmpfs	tmpfs	2.0G	0	2.0G	0%	/sys/fs/cgroup
/dev/sda1	vfat	511M	4.0K	511M	1%	/boot/efi
tmpfs	tmpfs	394M	32K	394M	1%	/run/user/1000
/dev/sr0	iso9660	59M	59M	0	100%	/media/gnkhalsa/VBox_GAs_6.1.34

**Step 7:** In this step we will configure the mounting on fstab to mount it when the system boot.

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo nano /etc/fstab
```



**Step 8:** Now we will install our auto mounting fstab using the following command.

**sudo apt-get install autofs**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo apt-get install autofs
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  autofs
0 upgraded, 1 newly installed, 0 to remove and 342 not upgraded.
Need to get 515 kB of archives.
After this operation, 3,455 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 autofs amd64 5.1.6-2ubuntu0.1 [515 kB]
Fetched 515 kB in 0s (1,487 kB/s)
Selecting previously unselected package autofs.
(Reading database ... 276037 files and directories currently installed.)
Preparing to unpack .../autofs_5.1.6-2ubuntu0.1_amd64.deb ...
Unpacking autofs (5.1.6-2ubuntu0.1) ...
Setting up autofs (5.1.6-2ubuntu0.1) ...
```

**Step 9:** After getting done with the installation we have to open and check whether the auto.master file was created or not. This will be done by using the following command.

**sudo nano /etc/auto.master**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo nano /etc/auto.master
```

**Step 10:** After checking the auto.master file we will need to create a directory(file) in which be adding mounted data.

**sudo mkdir /mntdir**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo mkdir /mntdir
```

After creating this directory we can find this directory in files,computer section.

**Step 11:** Now after making the directory we will restart the auto mounting fstab, using the following command.

**systemctl restart autofs**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ systemctl restart autofs
```

**Step 12:** Now we have to mount the data, before doing that we will need to change the current directory we are in to the directory we made for data mounting purpose (mntdir). For changing directory we will use "cd"

**cd/mntdir**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ cd /mntdir
gnkhalsa@gnkhalsa-VirtualBox:/mntdir$ ll
```

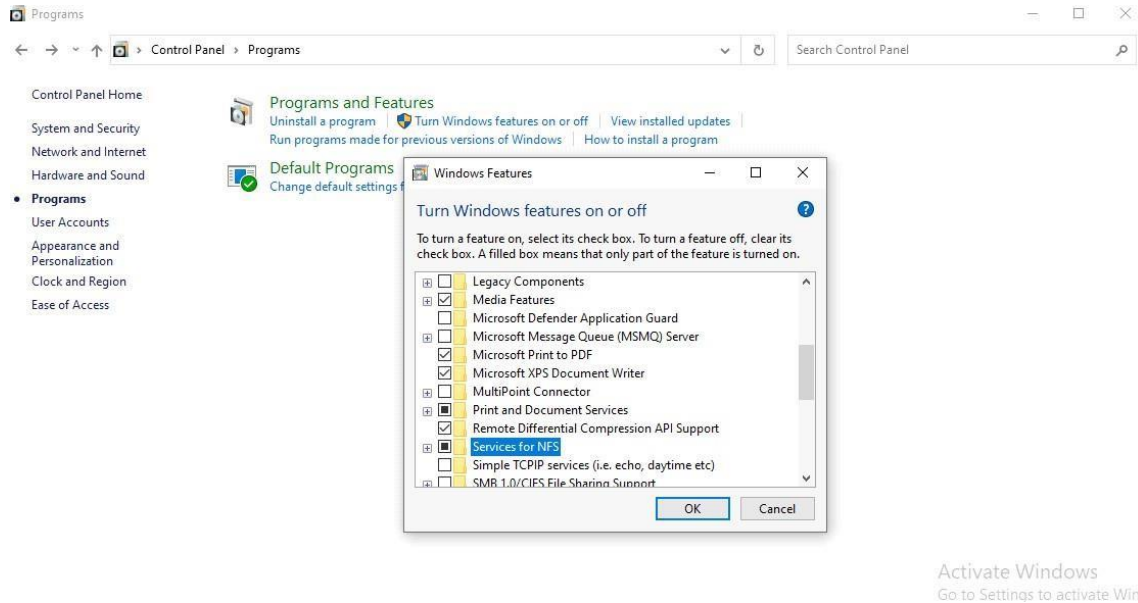
**Step 13:** We have to mount the data into mntdir which we was mounted we installed the NFS and auto fstab in the mounts directory we will use grep command. it can be done as follows.

`cat /proc/mounts | grep mntdir`

```
gnkhalsa@gnkhalsa-VirtualBox:/mntdir$ cat /proc/mounts | grep mntdir
```

**Step 14-** Make sure you start your systems NFS services, which can be done by going into the Control Panel>Programs>Turn Windows features on or off.

There you will see Services for NFS just tick them both individually and then press ok.





## Practical 3

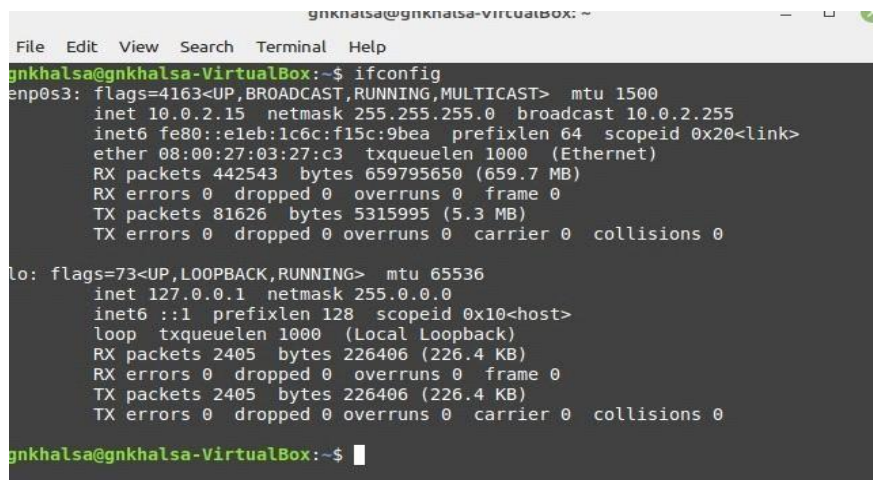
**Aim:-** Initial settings .Add a user. Network Settings, change to static IP address, Disable IPv6 if not needed, Configure Services, during the list of services which are running, stop and turn OFF auto-start setting for a service if you don't need it, Sudo settings.

**Step 1:** Add a User

**Step 2:** Change to static IP address

To assign a static ip address to a Ubuntu system, you need to edit the etc/network/interfaces file.

First you'll need to find out your system's static ip address which can be done by the ifconfig command.



```
gnkhalsa@gnkhalsa-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::e1eb:1c6c:f15c:9bea prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:03:27:c3 txqueuelen 1000 (Ethernet)
    RX packets 442543 bytes 659795650 (659.7 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 81626 bytes 5315995 (5.3 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 2405 bytes 226406 (226.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2405 bytes 226406 (226.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

gnkhalsa@gnkhalsa-VirtualBox:~$
```

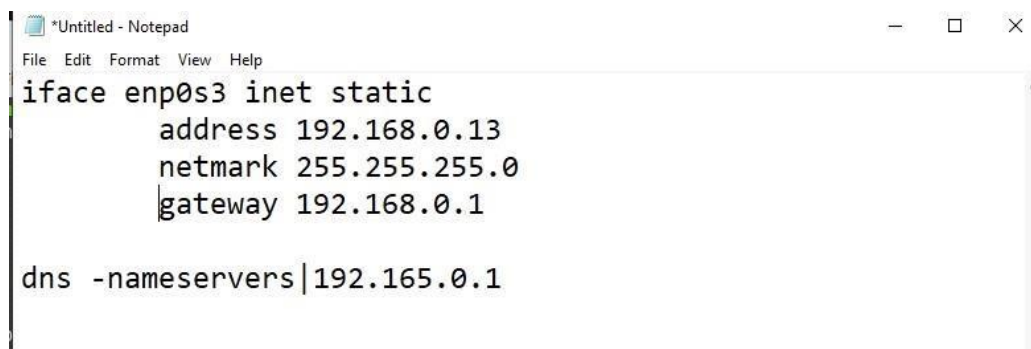
Now copy down the ip address and netmask from the output of ifconfig command into a notepad, to find out the gateway ip address type the following command.

**ip r | grep default**



```
gnkhalsa@gnkhalsa-VirtualBox:~$ ip r | grep default
default via 10.0.2.2 dev enp0s3 proto dhcp metric 100
```

Copy them as follows in a notepad.



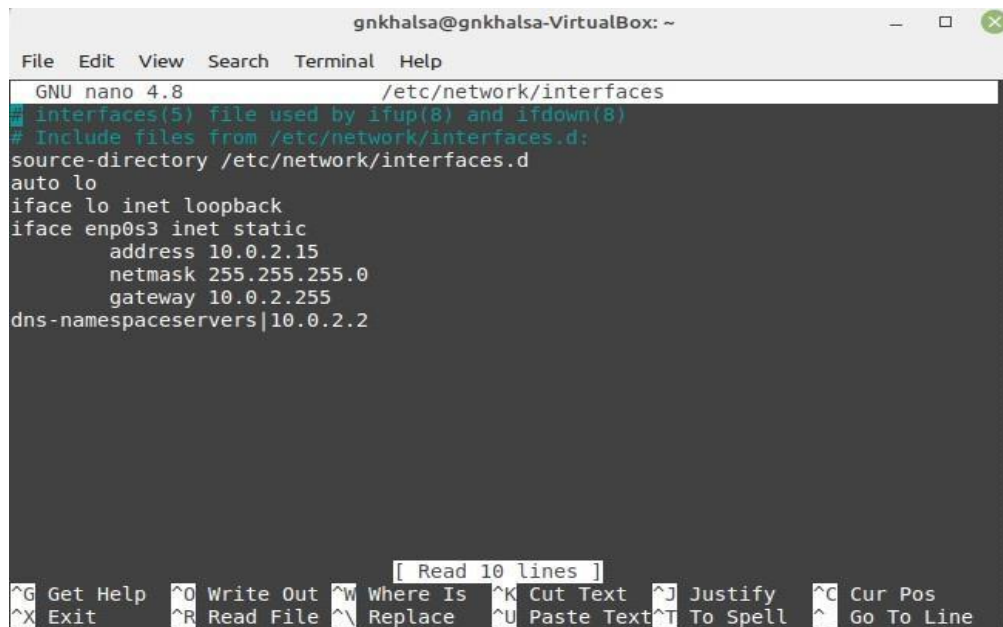
```
*Untitled - Notepad
File Edit Format View Help
iface enp0s3 inet static
    address 192.168.0.13
    netmask 255.255.255.0
    gateway 192.168.0.1

dns -nameservers|192.165.0.1
```

**Step 3:** Now editing the static IP address, for that you'll need the following command.  
**sudo nano /etc/network/interfaces**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo nano /etc/network/interfaces
[sudo] password for gnkhalsa:
```

Write down the data about static ip you copied in notepad into the interfaces file as follows.



```
gnkhalsa@gnkhalsa-VirtualBox: ~
File Edit View Search Terminal Help
GNU nano 4.8 /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d
auto lo
iface lo inet loopback
iface enp0s3 inet static
    address 10.0.2.15
    netmask 255.255.255.0
    gateway 10.0.2.255
dns-nameservers 10.0.2.2
```

Exit the file using 'ctrl+x' and then save the file.



**Step 4:** Now after editing your static IP you should reboot your machine.

**Step 5:** Now we will disable the IPv6 if not needed. That can be done using the following command.  
**sudo /bin/su -c "echo 'net.ipv6.conf.all.disable\_ipv6=1'>>/etc/sysctl.conf"**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo/bin/su-c "echo 'net.ipv6.conf.all.disable_ipv6=1'>> /etc/sysctl.conf"
```

**Step 6:** We will check the status of IPv6 using the following command.  
**sudo sysctl -p**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo sysctl -p
net.ipv6.conf.all.disable_ipv6 = 1
```

**Step 7:** Now to show the list of multicast address we will use the following command. **ipmaddr** .

```
gnkhalsa@gnkhalsa-VirtualBox:~$ ipmaddr
1:      lo
        inet  224.0.0.251
        inet  224.0.0.1
        inet6 ff02::1
        inet6 ff01::1
2:      enp0s3
        inet6 ff02::1
        inet6 ff01::1
```

**Step 8:** Now we will display the list of services which are running and have been stopped. Which will be done using the following command.

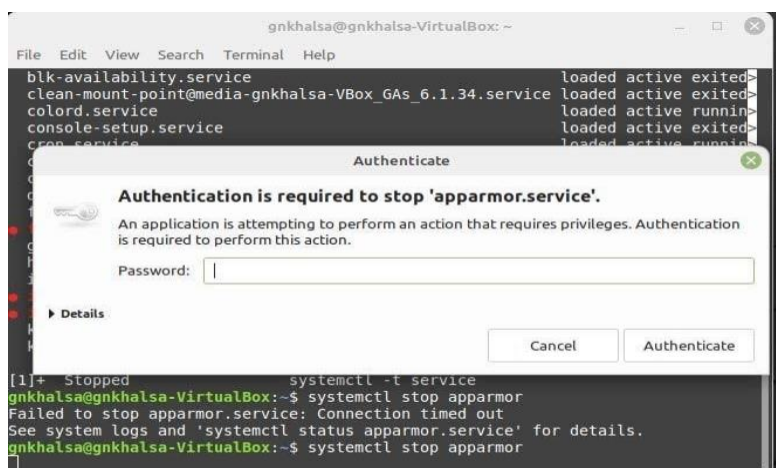
**systemctl -t service**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ systemctl -t service
UNIT                                LOAD    ACTIVE SUB    >
accounts-daemon.service            loaded active runnin>
acpid.service                      loaded active runnin>
alsa-restore.service              loaded active exited>
autofs.service                    loaded active runnin>
avahi-daemon.service              loaded active runnin>
blk-availability.service           loaded active exited>
clean-mount-point@media-gnkhalsa-VBox_GAs_6.1.34.service loaded active exited>
colord.service                    loaded active runnin>
console-setup.service             loaded active exited>
cron.service                      loaded active runnin>
cups-browsed.service              loaded active runnin>
cups.service                      loaded active runnin>
dbus.service                     loaded active runnin>
finalrd.service                  loaded active exited>
getty@tty1.service               loaded active runnin>
hddtemp.service                  loaded active exited>
ifupdown-pre.service             loaded active exited>
isc-dhcp-server.service           loaded failed failed>
isc-dhcp-server6.service          loaded failed failed>
kerneloops.service              loaded active runnin>
keyboard-setup.service           loaded active exited>
```

**Step 9:** Now we will stop the apparmor service from the list of services which can be done by using the following command.

**systemctl stop apparmor**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ systemctl stop apparmor
```

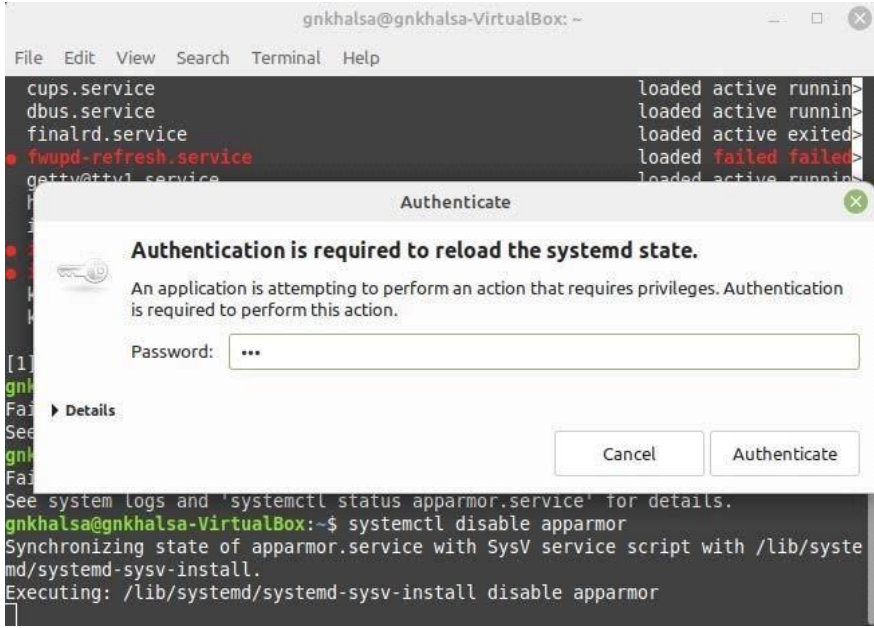


```
[1]+  Stopped                  systemctl -t service
gnkhalsa@gnkhalsa-VirtualBox:~$ systemctl stop apparmor
Failed to stop apparmor.service: Connection timed out
See system logs and 'systemctl status apparmor.service' for details.
gnkhalsa@gnkhalsa-VirtualBox:~$ systemctl stop apparmor
```

**Step 10:** After stopping the apparmor service we will disable it using the following command.

**systemctl disable apparmor**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo systemctl disable apparmor
```



The terminal window shows the following output for `systemctl status apparmor.service`:

Service	Loaded	Active	Running
cups.service	loaded	active	running
dbus.service	loaded	active	running
finalrd.service	loaded	active	exited
fwupd-refresh.service	loaded	failed	failed
getty@tty1.service	loaded	active	running

An authentication dialog box is displayed with the following text:

**Authentication is required to reload the systemd state.**

An application is attempting to perform an action that requires privileges. Authentication is required to perform this action.

Password:

Buttons: Cancel, Authenticate

The terminal output for `systemctl disable apparmor` is as follows:

```
gnkhalsa@gnkhalsa-VirtualBox:~$ systemctl disable apparmor
Synchronizing state of apparmor.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable apparmor
```

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo systemctl disable apparmor
Synchronizing state of apparmor.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable apparmor
Removed /etc/systemd/system/sysinit.target.wants/apparmor.service.
gnkhalsa@gnkhalsa-VirtualBox:~$
```



## Practical No. 4

**Aim:** To configure NTP Server and configure NTP client Network timing.

**Steps to do so.....**

**Step 1:** Step 1 would be installing ntp server on our machine which can be done by the following command. (Also make sure your machine's software are up to date).

**sudo apt-get install ntp**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo apt install ntp
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo apt install ntp
[sudo] password for gnkhalsa:
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  ntp-doc
The following packages will be REMOVED:
  systemd-timesyncd
The following NEW packages will be installed:
  ntp
0 upgraded, 1 newly installed, 1 to remove and 9 not upgraded.
Need to get 0 B/657 kB of archives.
After this operation, 1,770 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
dpkg: systemd-timesyncd: dependency problems, but removing anyway as you request
ed:
systemd depends on systemd-timesyncd | time-daemon; however:
Package systemd-timesyncd is to be removed.
Package time-daemon is not installed.
Package systemd-timesyncd which provides time-daemon is to be removed.
Package ntp which provides time-daemon is not installed.
systemd depends on systemd-timesyncd | time-daemon; however:
Package systemd-timesyncd is to be removed.
```

**Step 2:** To configure ntp the ntp file is stored at '/etc/ntp.conf' and can be modified with any editor. The command for that would be as follows.

**sudo nano /etc/ntp.conf**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo nano /etc/ntp.conf
GNU nano 4.8 /etc/ntp.conf
/etc/ntp.conf, configuration for ntpd; see ntp.conf(5) for help

driftfile /var/lib/ntp/ntp.drift

# Leap seconds definition provided by tzdata
leapfile /usr/share/zoneinfo/leap-seconds.list

# Enable this if you want statistics to be logged.
#statsdir /var/log/ntpstats/

statistics loopstats peerstats clockstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable

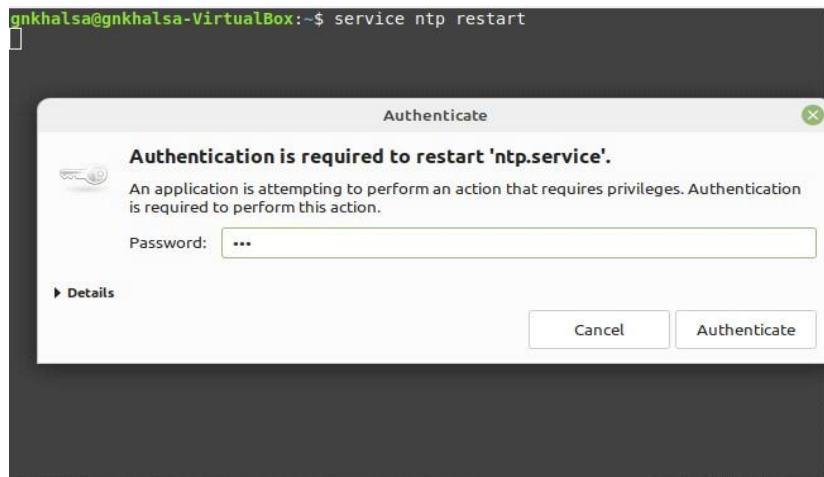
# Specify one or more NTP servers.

# Use servers from the NTP Pool Project. Approved by Ubuntu Technical Board
# on 2011-02-08 (LP: #104525). See http://www.pool.ntp.org/join.html for
# more information.

[ Read 60 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^_ Replace ^U Paste Text ^T To Spell ^_ Go To Line
```

**Step 3:** After the configuration you will need to restart the ntp service which can be done by the 'service ntp restart' command.





**Step 4:** Now we will show the NTP servers running on the host.

**ntpq -pn**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ ntpq -pn
      remote           refid      st t when poll reach  delay  offset jitter
=====
0.ubuntu.pool.n .POOL.    16 p   - 64    0   0.000   0.000   0.000
1.ubuntu.pool.n .POOL.    16 p   - 64    0   0.000   0.000   0.000
2.ubuntu.pool.n .POOL.    16 p   - 64    0   0.000   0.000   0.000
3.ubuntu.pool.n .POOL.    16 p   - 64    0   0.000   0.000   0.000
ntp.ubuntu.com .POOL.    16 p   - 64    0   0.000   0.000   0.000
gnkhalsa@gnkhalsa-VirtualBox:~$
```

**Step 5:** Now we will check the time synchronization on our client (Ubuntu) machine for that you will need to install the ntpdate package first which can be done by following command.

**sudo apt-get install ntpdate**

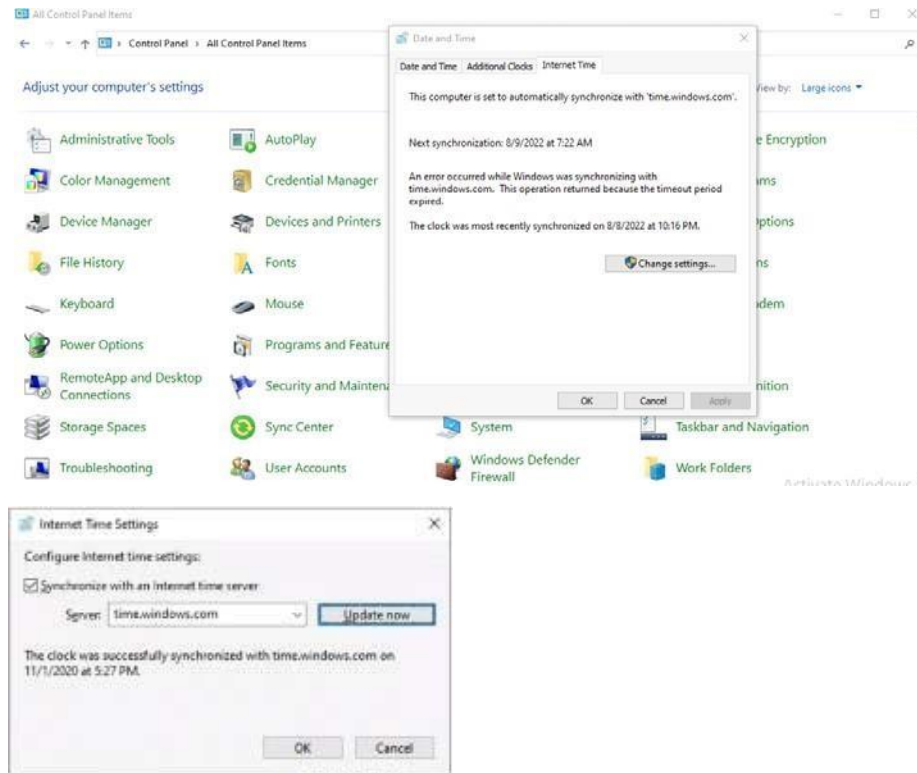
```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo apt-get install ntpdate
[sudo] password for gnkhalsa:
Reading package lists... Done
Building dependency tree
Reading state information... Done
ntpdate is already the newest version (1:4.2.8p12+dfsg-3ubuntu4.20.04.1).
0 upgraded, 0 newly installed, 0 to remove and 9 not upgraded.
gnkhalsa@gnkhalsa-VirtualBox:~$
```

**Step 6:** To check the time synchronization on our client machine use the following command.

**ntpdate ntp1.jst.mfeed.ad.jp**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo ntpdate ntp1.jst.mfeed.ad.jp
9 Aug 02:25:17 ntpdate[74269]: the NTP socket is in use, exiting
```

**Step 7:** Now we will do synchronization on our Windows Client, for that you will need to go to control panel and then open “Date and timing” setting, move to the tab “Internet Time” and click change settings and then update it.



## Practical 5

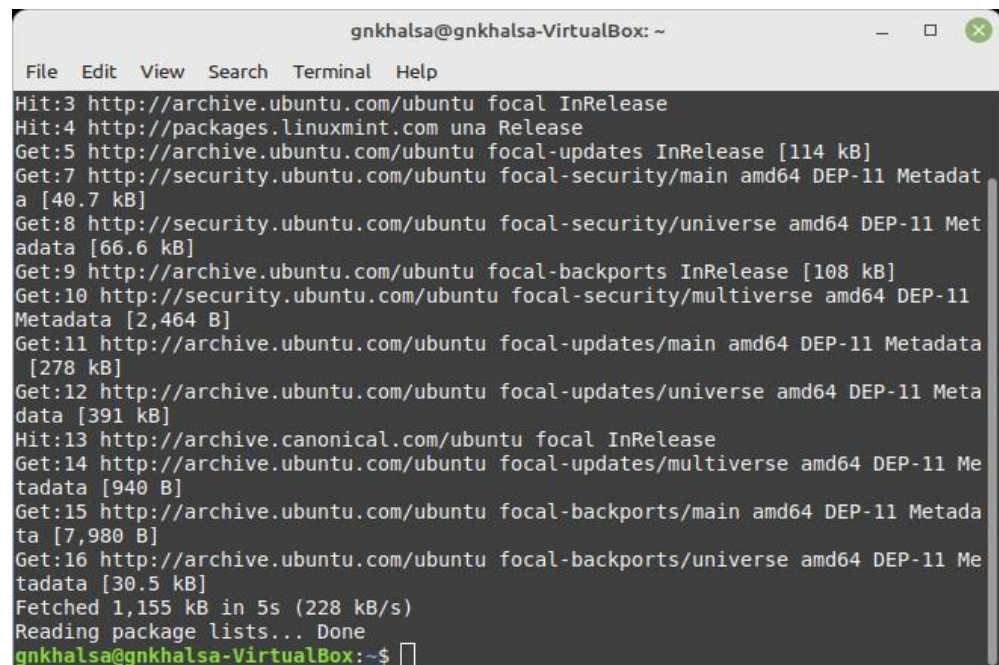
**Aim:** To Install and configure DNS server in Ubuntu.

**Step 1:** Make sure your Ubuntu server is up-to-date, if not then use the update and upgrade command.

`sudo apt-get`

`update sudo apt-get`

`upgrade`



```
gnkhalsa@gnkhalsa-VirtualBox: ~  
File Edit View Search Terminal Help  
Hit:3 http://archive.ubuntu.com/ubuntu focal InRelease  
Hit:4 http://packages.linuxmint.com una Release  
Get:5 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]  
Get:7 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [40.7 kB]  
Get:8 http://security.ubuntu.com/ubuntu focal-security/universe amd64 DEP-11 Metadata [66.6 kB]  
Get:9 http://archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]  
Get:10 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 DEP-11 Metadata [2,464 B]  
Get:11 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [278 kB]  
Get:12 http://archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [391 kB]  
Hit:13 http://archive.canonical.com/ubuntu focal InRelease  
Get:14 http://archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 DEP-11 Metadata [940 B]  
Get:15 http://archive.ubuntu.com/ubuntu focal-backports/main amd64 DEP-11 Metadata [7,980 B]  
Get:16 http://archive.ubuntu.com/ubuntu focal-backports/universe amd64 DEP-11 Metadata [30.5 kB]  
Fetched 1,155 kB in 5s (228 kB/s)  
Reading package lists... Done  
gnkhalsa@gnkhalsa-VirtualBox:~$
```

**Step 2:** After updating the system, run the following command to install BIND9 packages which are used to setup DNS server.

**sudo apt-get install bind9 bind9utils bind9**

```

gnkhalsa@gnkhalsa-VirtualBox: ~
File Edit View Search Terminal Help
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
done.
Setting up libnemo-desktop4:amd64 (5.2.1+una) ...
Processing triggers for ntp (1:4.2.8p12+dfsg-3ubuntu4.20.04.1) ...
Processing triggers for dbus (1.12.16-2ubuntu2.2) ...
Processing triggers for shared-mime-info (1.15-1) ...
Processing triggers for mintsystem (8.5.3) ...
Setting up gir1.2-cinnamondesktop-3.0 (5.2.1+una) ...
Setting up libmuffin0 (5.2.1+una) ...
Processing triggers for install-info (6.7.0.dfsg.2-5) ...
Setting up nemo (5.2.4+una) ...
Setting up muffin (5.2.1+una) ...
Processing triggers for fontconfig (2.13.1-2ubuntu3) ...
Setting up cinnamon-screensaver (5.2.1+una) ...
Setting up muffin-dbg (5.2.1+una) ...
Processing triggers for desktop-file-utils (0.24+linuxmint1) ...
Setting up nemo-dbg (5.2.4+una) ...
Setting up libnemo-desktop-dbg:amd64 (5.2.1+una) ...
Setting up gir1.2-meta-muffin-0.0 (5.2.1+una) ...
Processing triggers for initramfs-tools (0.136ubuntu6.7) ...
update-initramfs: Generating /boot/initrd.img-5.4.0-91-generic

```

**Step 3:** Configuring the caching name server, caching name server saves the DNS query results locally for a particular period of time. To configure caching name server, edit/etc/bind/named.conf.options file. Which will be done by the following command.

**sudo nano /etc/bind/named.conf.options**

Uncomment the following lines and then, add then,add your ISP or Google public DNS server IP addresses.

Save and close the file.

```

gnkhalsa@gnkhalsa-VirtualBox: ~
File Edit View Search Terminal Help
GNU nano 4.8 /etc/bind/named.conf.options Modified
options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk. See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    forwarders {
        0.0.0.0;
    };

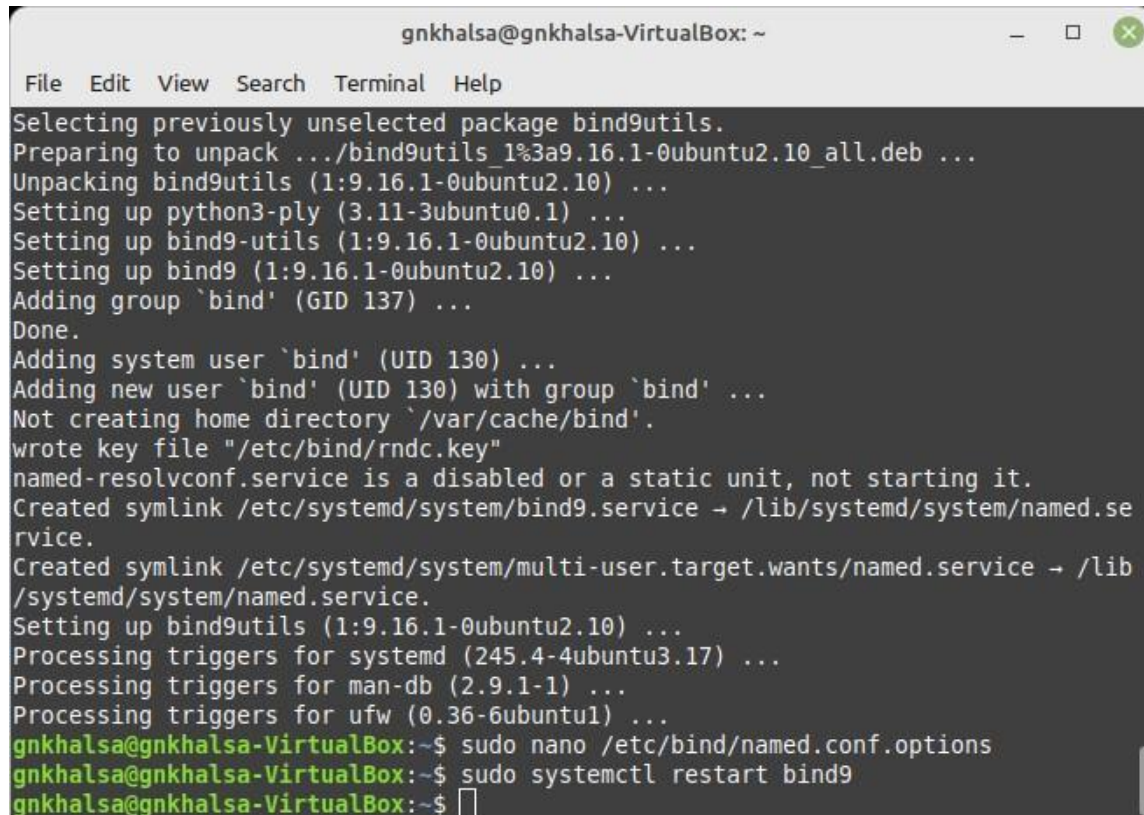
    //=====
    // If BIND logs error messages about the root key being expired,
    // you will need to update your keys. See https://www.isc.org/bind-keys
    //=====
File Name to Write: /etc/bind/named.conf.options
^G Get Help      M-D DOS Format  M-A Append      M-B Backup File
^C Cancel        M-M Mac Format  M-P Prepend     ^T To Files

```

**Step 4:** Now restart the bind9 service to take effect the changes. This will be done by the following command.

**`sudo systemctl restart bind9`**

We have successfully installed the caching name server.



```
gnkhalsa@gnkhalsa-VirtualBox: ~  
File Edit View Search Terminal Help  
Selecting previously unselected package bind9utils.  
Preparing to unpack .../bind9utils_1%3a9.16.1-0ubuntu2.10_all.deb ...  
Unpacking bind9utils (1:9.16.1-0ubuntu2.10) ...  
Setting up python3-ply (3.11-3ubuntu0.1) ...  
Setting up bind9-utils (1:9.16.1-0ubuntu2.10) ...  
Setting up bind9 (1:9.16.1-0ubuntu2.10) ...  
Adding group `bind' (GID 137) ...  
Done.  
Adding system user `bind' (UID 130) ...  
Adding new user `bind' (UID 130) with group `bind' ...  
Not creating home directory `/var/cache/bind'.  
wrote key file "/etc/bind/rndc.key"  
named-resolvconf.service is a disabled or a static unit, not starting it.  
Created symlink /etc/systemd/system/bind9.service → /lib/systemd/system/named.service.  
Created symlink /etc/systemd/system/multi-user.target.wants/named.service → /lib/systemd/system/named.service.  
Setting up bind9utils (1:9.16.1-0ubuntu2.10) ...  
Processing triggers for systemd (245.4-4ubuntu3.17) ...  
Processing triggers for man-db (2.9.1-1) ...  
Processing triggers for ufw (0.36-6ubuntu1) ...  
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo nano /etc/bind/named.conf.options  
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo systemctl restart bind9  
gnkhalsa@gnkhalsa-VirtualBox:~$
```



## Practical 6

**Aim :-** Install Samba to share folders or files between Windows and Linux.

### **STEPS'S TO INSTALL SAMBA AND SHARE FOLDER'S ANF FILE'S:**

**Step1 :** This is the basic step and should be performed every time, that is to update system repositories.

**sudo apt-get**

**update sudo apt-get**

**upgrade**

**Step2:** Now after updating system repositories next step is to install samba using command

**sudo apt-get install samba**

```
File Edit View Search Terminal Help
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo apt-get install samba
[sudo] password for gnkhalsa:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ibverbs-providers libcephfs2 libibverbs1 libldb2 librados2 librdmacm1
  libsmbclient libwbclient0 python3-dnspython python3-ldb python3-markdown
  python3-pygments python3-samba samba-common samba-common-bin
  samba-dsdb-modules samba-libs samba-vfs-modules smbclient tdb-tools
Suggested packages:
  python-markdown-doc python-pygments-doc ttf-bitstream-vera ctdb ldb-tools
  smbldap-tools winbind heimdal-clients
Recommended packages:
  python3-gpg
The following NEW packages will be installed:
  ibverbs-providers libcephfs2 libibverbs1 librados2 librdmacm1
  python3-dnspython python3-markdown python3-pygments samba samba-dsdb-modules
  samba-vfs-modules tdb-tools
The following packages will be upgraded:
  libldb2 libsmbclient libwbclient0 python3-ldb python3-samba samba-common
  samba-common-bin samba-libs smbclient
9 upgraded, 12 newly installed, 0 to remove and 345 not upgraded.
Need to get 16.1 MB of archives.
```

**Step 3:** Now you need to configure the shared folder so first thing you want to do is to edit the configuration file.

**sudo nano /etc/samba/smb.conf**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo nano /etc/samba/smb.conf
```

Now first things you want to check in this configuration file is: Go to Global Settings section in that file and check whether workgroup=WORKGROUP, here WORKGROUP is name of your workgroup windows.

```

# enough to be mentioned here
#
# NOTE: Whenever you modify this file you should run the command
# "testparm" to check that you have not made any basic syntactic
# errors.

===== Global Settings =====

[global]

## Browsing/Identification ###
# Change this to the workgroup/NT-domain name your Samba server will part of
workgroup = WORKGROUP

# server string is the equivalent of the NT Description field
server string = %h server (Samba, Ubuntu)

#### Networking ####

^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^N Replace    ^U Paste Text ^T To Spell   ^_ Go To Line

```

Next is go to authentication part of same configuration file and check whether security=user if this line is present no need to change, if this line is missing then type security=user and save file.

```

##### Authentication #####

# Server role. Defines in which mode Samba will operate. Possible
# values are "standalone server", "member server", "classic primary
# domain controller", "classic backup domain controller", "active
# directory domain controller".
#
# Most people will want "standalone server" or "member server".
# Running as "active directory domain controller" will require first
# running "samba-tool domain provision" to wipe databases and create a
# new domain.
server role = standalone server
security = user

obey pam restrictions = yes

# This boolean parameter controls whether Samba attempts to sync the Unix
# password with the SMB password when the encrypted SMB password in the
# passdb is changed.
unix password sync = yes

^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^N Replace    ^U Paste Text ^T To Spell   ^_ Go To Line

```

**Step4:** Now we will make the directory and check if it's created or not.

- a) `sudo mkdir -p /srv/samba/share` b) `cd /srv` c) `ls`  
d) `ls samba`

```

gnkhalsa@gnkhalsa-VirtualBox:~$ sudo mkdir -p /srv/samba/share
gnkhalsa@gnkhalsa-VirtualBox:~$ cd /srv
gnkhalsa@gnkhalsa-VirtualBox:/srv$ ls
samba
gnkhalsa@gnkhalsa-VirtualBox:/srv$ ls samba
share
gnkhalsa@gnkhalsa-VirtualBox:/srv$

```

**Step5:** Once you have made the directories you need to change ownership so to do that we execute the following command. `sudo chown nobody.nogroup /srv/samba/share`

```
gnkhalsa@gnkhalsa-VirtualBox:/srv$ sudo chown nobody.nogroup /srv/samba/share
```

**Step6:** Now we have to list the files present in the directory we created, so we will use the list long command but before that we will move into the directory.

a) `cd/srv/samba` b) `ll`

```
gnkhalsa@gnkhalsa-VirtualBox:/srv$ cd /srv/samba
gnkhalsa@gnkhalsa-VirtualBox:/srv/samba$ ll
total 12
drwxr-xr-x 3 root root 4096 Aug 10 22:10 ./
drwxr-xr-x 3 root root 4096 Aug 10 22:10 ../
drwxr-xr-x 2 nobody nogroup 4096 Aug 10 22:10 share/
```

**Step7:** Now we will configure the nmbd file for samba, for that we will use the following command.

`sudo nano /etc/init/nmbd.conf`

```
gnkhalsa@gnkhalsa-VirtualBox:/srv/samba$ sudo nano /etc/init/nmbd.conf
```

Comment out all the lines in pre-start script except one line.

```
GNU nano 2.5.3      File: /etc/init/nmbd.conf      Modified
description "NetBIOS name server"
author      "Steve Langasek <steve.langasek@ubuntu.com>"
start on (local-filesystems and net-device-up IFACE!=lo)
stop on runlevel [12345]
expect fork
respawn
pre-start script
# [ -f /etc/samba/smb.conf ] || { stop; exit 0; }
# install -o root -g root -m 755 -d /var/run/samba
# NMBD_DISABLED="testparm -s --parameter-name='disable netbios' 2>/dev/n$
# [ "$NMBD_DISABLED" = xYes ] && { stop; exit 0; }
# exit 0
end script
exec nmbd -D
^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify
^X Exit          ^R Read File    ^_ Replace      ^U Uncut Text   ^T To Spell
```

**Step 8:** Now we will restart our nmbd and smb services using commands.

`sudo service smb`

`restart sudo service nmbd`

`restart`

```
gnkhalsa@gnkhalsa-VirtualBox:/srv/samba$ sudo service smb restart
```

**Step 9:** Now create a file using touch command in /srv/samba/share named test.txt and we will check if our file is created or not.

```
gnkhalsa@gnkhalsa-VirtualBox:/srv/samba/share$ sudo touch /srv/samba/share/test,txt
```

```
gnkhalsa@gnkhalsa-VirtualBox:/srv/samba/share$ ll /srv/samba/share
total 8
drwxr-xr-x 2 nobody nogroup 4096 Aug 10 23:33 ./
drwxr-xr-x 3 root    root    4096 Aug 10 22:10 ../
-rw-r--r-- 1 root    root      0 Aug 10 23:33 test,txt
gnkhalsa@gnkhalsa-VirtualBox:/srv/samba/share$
```

## Practical 7

**Aim:-** To install SSH ( Secure Shell) server on you Ubuntu machine and configure it.

**Step1:** It is always necessary to keep your machine up-to-date before you install any type of server or service in it.

**sudo apt-get update**

**sudo apt-get upgrade**

**Step 2:** Now we have to install the SSH server, so we use the following command.

**sudo apt-get -y install openssh-server**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo apt-get -y install openssh-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere ssh-askpass
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 4 newly installed, 0 to remove and 9 not upgraded.
Need to get 688 kB of archives.
After this operation, 6,010 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu focal/main amd64 ncurses-term all 6.2-0ubuntu2 [249 kB]
Get:2 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-sftp-server amd64 1:8.2p1-4ubuntu0.5 [51.5 kB]
Get:3 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-server amd64 1:8.2p1-4ubuntu0.5 [377 kB]
Get:4 http://archive.ubuntu.com/ubuntu focal/main amd64 ssh-import-id all 5.10-0ubuntu1 [10.0 kB]
Fetched 688 kB in 2s (281 kB/s)
Preconfiguring packages ...
Selecting previously unselected package ncurses-term.
```

**Step3:** Now we have to configure the file SSH server which can be done by editing the 'sshd\_config' file for that we have to use the following command.

**sudo nano /etc/ssh/sshd\_config**

Here we have to edit the file and make a change in the authentication block "Permitrootlogin yes"



```

GNU nano 4.8 /etc/ssh/sshd_config
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

LoginGraceTime 2m
PermitRootLogin yes
StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

```

<sup>^</sup>G Get Help   <sup>^</sup>O Write Out   <sup>^</sup>W Where Is   <sup>^</sup>K Cut Text   <sup>^</sup>J Justify   <sup>^</sup>C Cur Pos  
<sup>^</sup>X Exit   <sup>^</sup>R Read File   <sup>^</sup>\ Replace   <sup>^</sup>U Paste Text   <sup>^</sup>T To Spell   <sup>^</sup>\_ Go To Line

**Step4:** Now we have to restart the ssh server which can be done by following command..

**sudo systemctl restart ssh**

```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo systemctl restart ssh
```

**Step5:** In this step we install openssh-client server on our machine by using the following command.

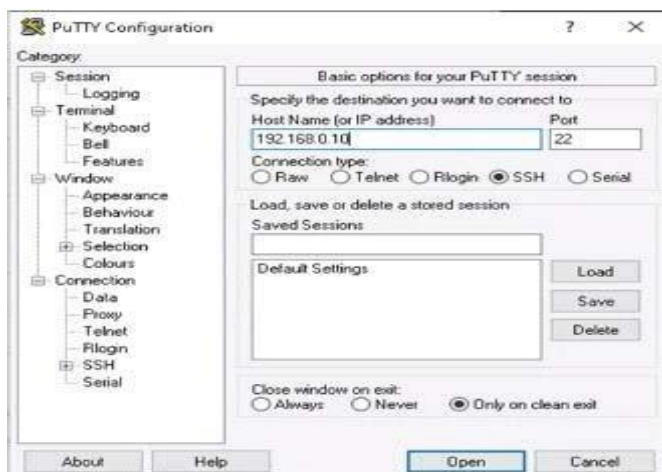
```
gnkhalsa@gnkhalsa-VirtualBox:~$ sudo apt-get -y install openssh-client
```

```

gnkhalsa@gnkhalsa-VirtualBox:~$ sudo apt-get -y install openssh-client
Reading package lists... Done
Building dependency tree
Reading state information... Done
openssh-client is already the newest version (1:8.2p1-4ubuntu0.5).
0 upgraded, 0 newly installed, 0 to remove and 9 not upgraded.

```

**Step6:** After installing all the necessary servers on your ubuntu machine Install Putty on your Windows client and open it put your ubuntu machine's ip address which can be know using ifconfig command.



## Practical 8

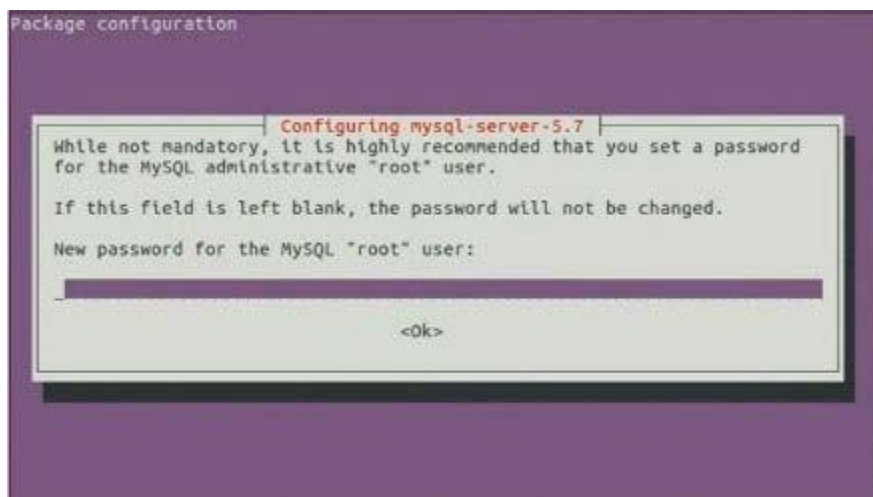
**Aim:** To install MySQL server to configure database server and install phpMyAdmin to operate MySQL on web browser from clients.

**Step 1:** First step would be to install MySQL server and configure it.

**sudo apt-get -y install mysql-server-5.7**

```
roar@roar-VirtualBox:~$ sudo apt-get -y install mysql-server-5.7
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  snapd-login-service
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  libhtml-template-perl mysql-client-5.7 mysql-client-core-5.7 mysql-common
  mysql-server-core-5.7
Suggested packages:
  libipc-sharedcache-perl mailx tinyc
The following NEW packages will be installed:
  libhtml-template-perl mysql-client-5.7 mysql-client-core-5.7 mysql-common
  mysql-server-5.7 mysql-server-core-5.7
0 upgraded, 6 newly installed, 0 to remove and 1 not upgraded.
Need to get 18.1 MB of archives.
After this operation, 152 MB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu xenial-updates/main i386 mysql-common
all 5.7.32-0ubuntu0.16.04.1 [14.8 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu xenial-updates/main i386 mysql-client-
core-5.7 i386 5.7.32-0ubuntu0.16.04.1 [6,365 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu xenial-updates/main i386 mysql-client-
5.7 i386 5.7.32-0ubuntu0.16.04.1 [1,466 kB]
```

During installation it will ask you set the root users password which you will be using when you will be accessing the database. So set the password accordingly.



**Step 2:** Once the MySQL server installation is done you can connect to it by opening it.

**mysql -u root -p**

```
roar@roar-VirtualBox:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.7.32-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

**Step 3:** Now we will display the user info and also the ip for our host by using the following command.  
`select user,host from mysql.user;`

```
mysql> select user,host from mysql.user;
+-----+-----+
| user          | host          |
+-----+-----+
| debian-sys-maint | localhost    |
| mysql.session  | localhost    |
| mysql.sys      | localhost    |
| root           | localhost    |
+-----+-----+
4 rows in set (0.00 sec)

mysql> exit
Bye
```

`SHOW VARIABLES WHERE Variable_name='port';`

```
mysql> SHOW VARIABLES WHERE Variable_name='port';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| port          | 3306  |
+-----+-----+
1 row in set (0.05 sec)

mysql>
```

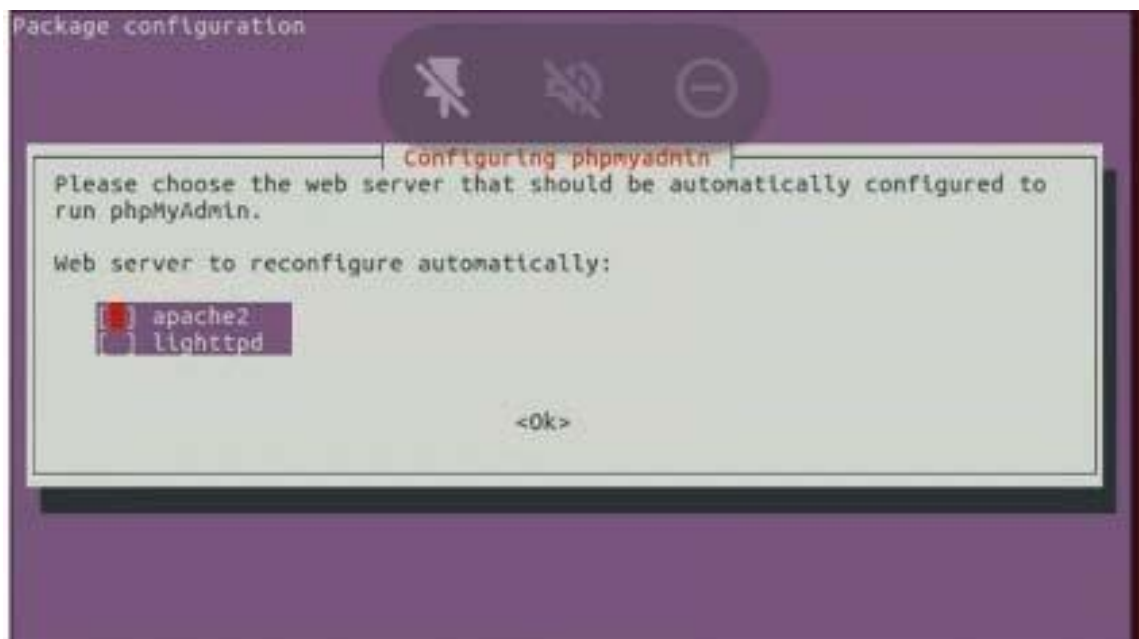
Enter the mysql prompt using 'exit' command.

**Step 4:** Now let's install phpMyAdmin so that we can operate MySQL on web browser from clients.  
`sudo apt-get -y install phpmyadmin`

```
roar@roar-VirtualBox:~$ sudo apt-get -y install phpmyadmin
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  snapd-login-service
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils dbconfig-common
  dbconfig-mysql javascript-common libapache2-mod-php libapache2-mod-php7.0
  libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libjs-jquery
  libjs-sphinxdoc libjs-underscore liblua5.1-0 libmcrypt4 php-common php-gd
  php-gettext php-mbstring php-mcrypt php-mysql php-pear php-phpseclib
  php-tcpdf php-xml php7.0-cli php7.0-common php7.0-gd php7.0-json
  php7.0-mbstring php7.0-mcrypt php7.0-mysql php7.0-opcache php7.0-readline
  php7.0-xml
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom mysql-server
  | mariadb-server libmcrypt-dev mcrypt php-libsodium php-gmp php-inagick
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils dbconfig-common
  dbconfig-mysql javascript-common libapache2-mod-php libapache2-mod-php7.0
  libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libjs-jquery
```

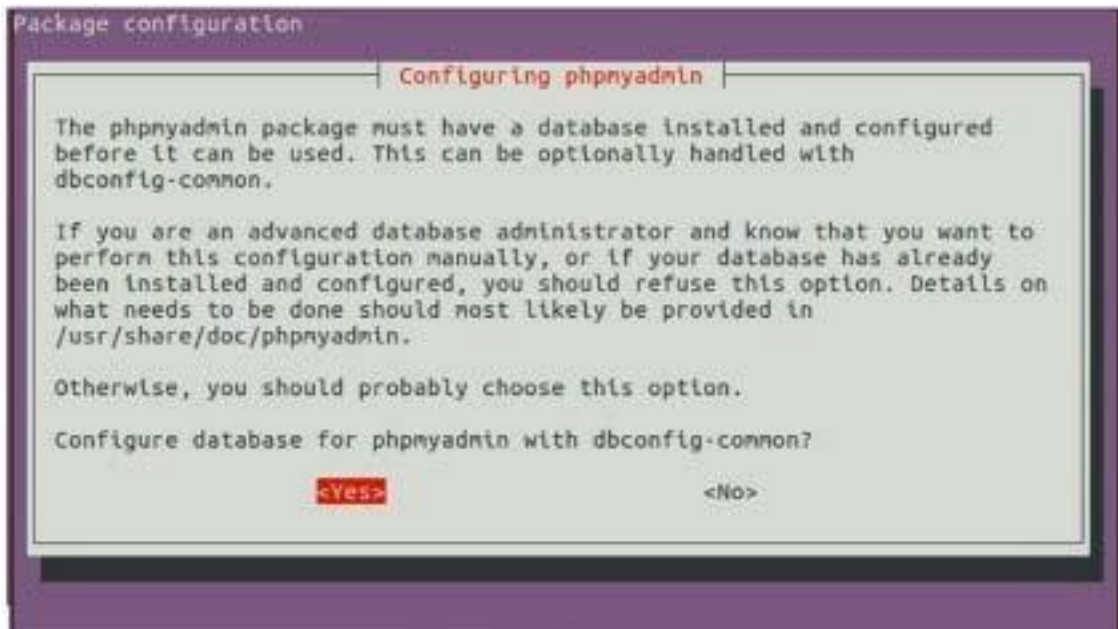
During installation you'll be asked which web server you want to configure to run phpMyAdmin.

Select "apache2".



After that you'll be asked that if you want dbconfig-common for your phpMyAdmin, select yes and proceed.





Now set the password for phpmyadmin.



**Step 5:** Now we have to edit the apache file and set the required ip for our server.

**`sudo nano /etc/phpmyadmin/apache.conf`**

```
roar@roar-VirtualBox:~$ sudo vi /etc/phpmyadmin/apache.conf
```

Once you open the apache.conf file to get the line numbers type the command “: set number”



```
roar@roar-VirtualBox: -
1 # phpMyAdmin default Apache configuration
2
3 Alias /phpmyadmin /usr/share/phpmyadmin
4
5 <Directory /usr/share/phpmyadmin>
6     Options FollowSymLinks
7     DirectoryIndex index.php
8
9     <IfModule mod_php.c>
10         <IfModule mod_mime.c>
11             AddType application/x-httpd-php .php
12         </IfModule>
13         <FilesMatch ".+\.php$">
14             SetHandler application/x-httpd-php
15         </FilesMatch>
16
17         php_flag magic_quotes_gpc Off
18         php_flag track_vars On
19         php_flag register_globals Off
20         php_admin_flag allow_url_fopen On
21         php_value include_path .
22         php_admin_value upload_tmp_dir /var/lib/phpmyadmin/tmp
23
24 @
:set number
```

Now add the required ip near the 8<sup>th</sup> line "Require ip 127.0.0.1 10.0.0.0/24".

After adding this,save the file and exit.

```
# phpMyAdmin default Apache configuration
Alias /phpmyadmin /usr/share/phpmyadmin
<Directory /usr/share/phpmyadmin>
    Options FollowSymLinks
    DirectoryIndex index.php
    Require ip 127.0.0.1 10.0.0.0/24
    <IfModule mod_php.c>
        <IfModule mod_mime.c>
            AddType application/x-httpd-php .php
        </IfModule>
        <FilesMatch ".+\.php$">
            SetHandler application/x-httpd-php
        </FilesMatch>
    </IfModule>

    php_flag magic_quotes_gpc Off
    php_flag track_vars On
    php_flag register_globals Off
    php_admin_flag allow_url_fopen On
    php_value include_path .
    php_admin_value upload_tmp_dir /var/lib/phpmyadmin/tmp
25
:wq!
```

**Step 6:** Now restart the apache server which can be done using the following command.

**`sudo /etc/init.d/apache2 restart`**

```
roar@roar-VirtualBox:~$ sudo /etc/init.d/apache2 restart  
[ ok ] Restarting apache2 (via systemctl): apache2.service.
```

←-----ERROR-----→

**Step 7:** Open the browser on your ubuntu machine and type the following in the search bar.

<http://localhost:3036/phpmyadmin/>

**Step 8:** If the page opens type your root users name and password.

\_\_\_\_\_