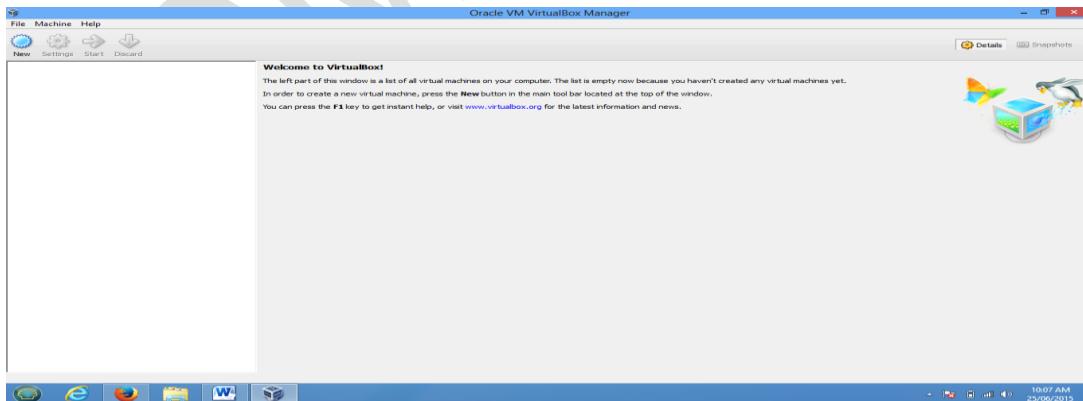


LAB 1**Creating and running virtual machines on Hosted Hypervisor like Virtual Box and KVM****1) Hosted Virtualization on Oracle Virtual Box Hypervisor**

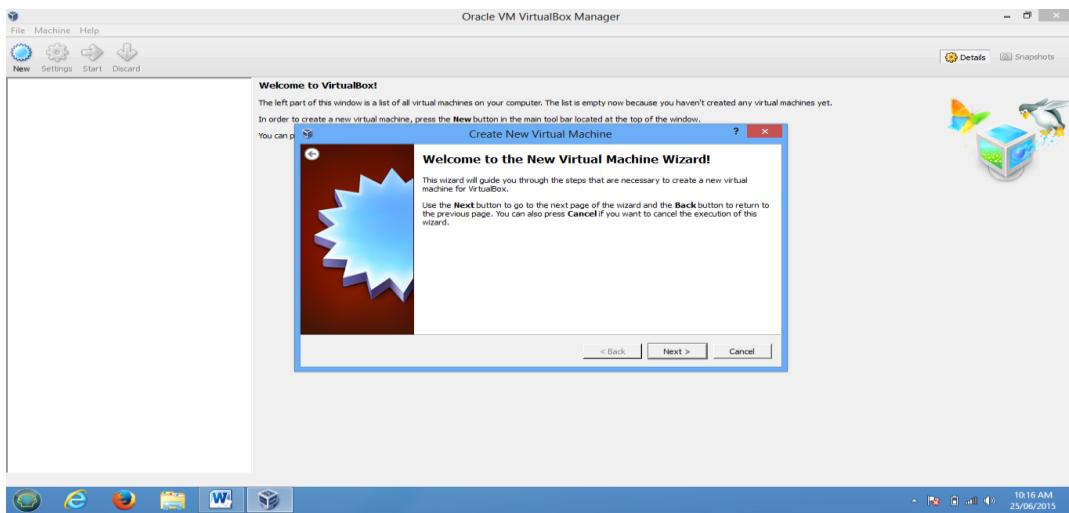
Step 1: Download Oracle Virtual box from <https://www.virtualbox.org/wiki/Downloads>



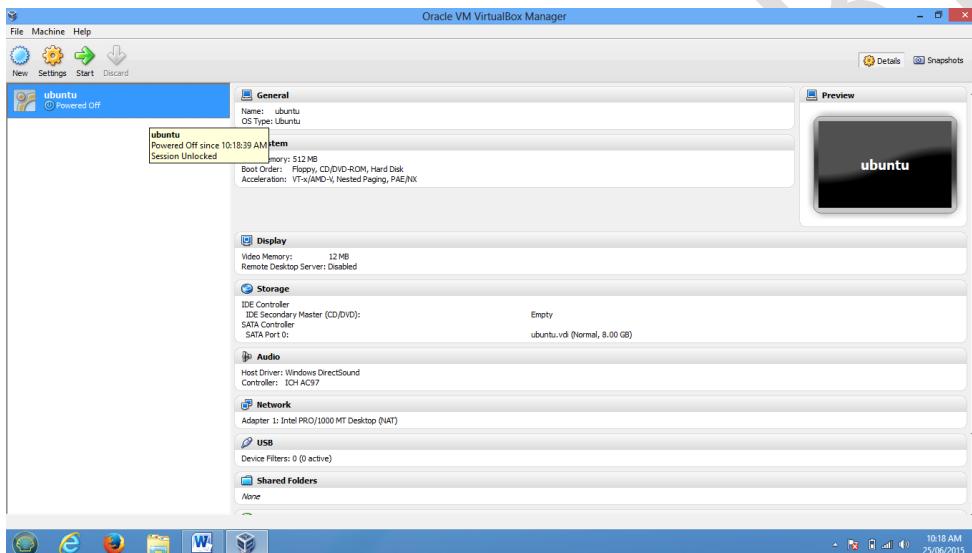
Step 2: Install it in Windows, Once the installation has done open it.



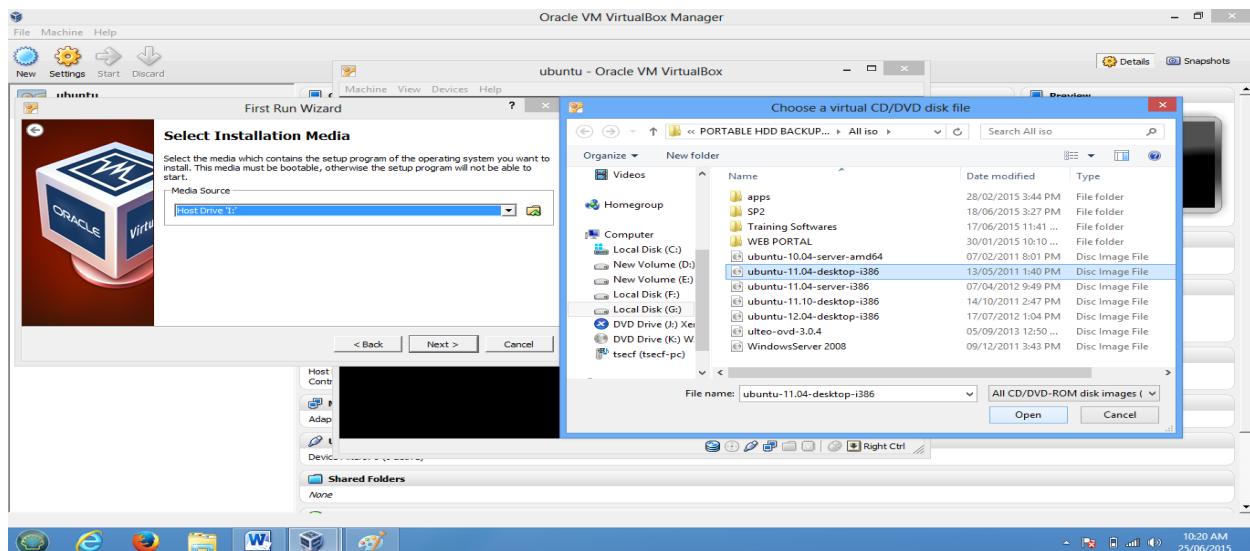
Step 3:-Create Virtual Machine by clicking on New



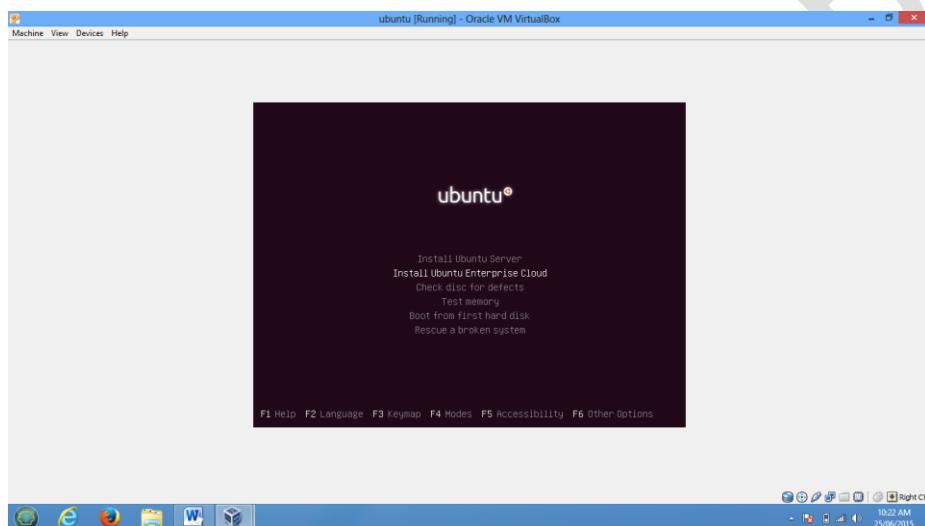
Step 4:- Specify RAM Size, HDD Size, and Network Configuration and Finish the wizard



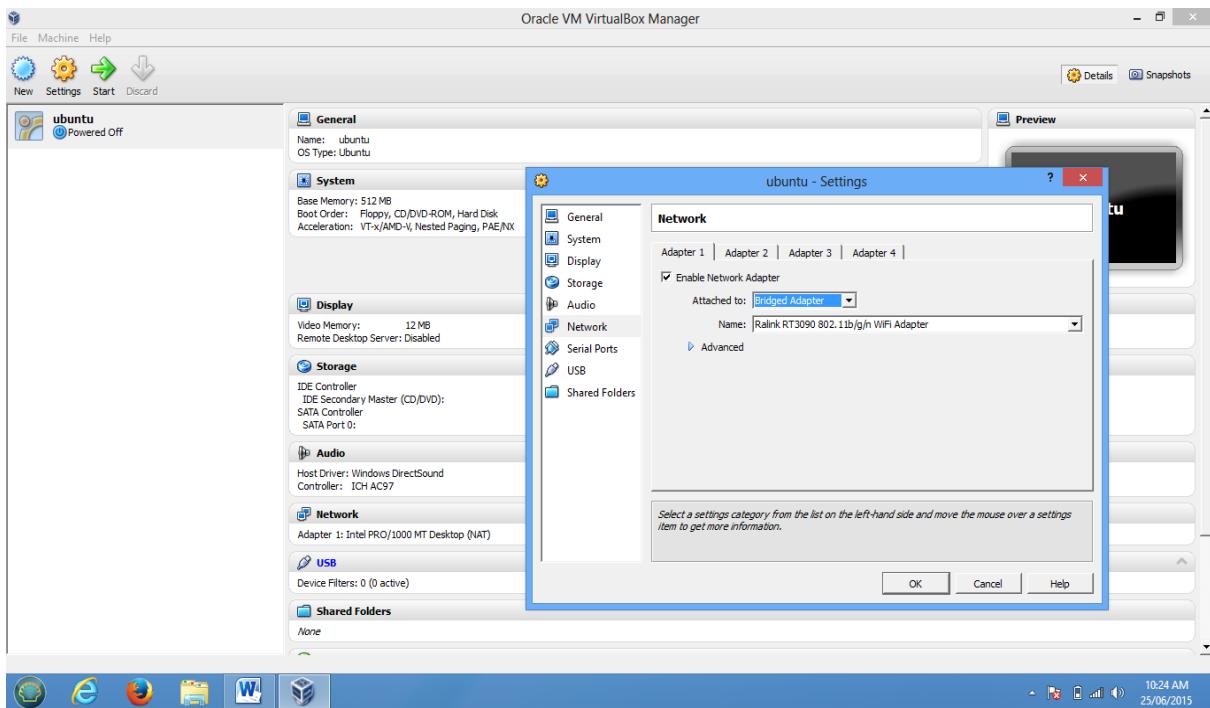
Step 4:- To Select the media for installation Click on start and browse for iso file



Step 5: Complete the Installation and use it.



Step 6: To Connect OS to the network change network Mode to Bridge Adaptor



2) Hosted Virtualization on KVM Hypervisor

The Steps to Create and run Virtual machines in KVM are as follows

1) Check whether CPU has hardware virtualization support.

KVM only works if your CPU has hardware virtualization support – either Intel VT-x or AMD-V. To determine whether your CPU includes these features, run the following command:

```
#sudo grep -c "svm|vmx" /proc/cpuinfo
```

```
root@ubuntu:/home/tsec# sudo grep -c "svm\|vmx" /proc/cpuinfo
3
root@ubuntu:/home/tsec#
```

A 0 indicates that your CPU doesn't support hardware virtualization, while a 1 or more indicates that it does.

2) Install KVM and supporting packages.

Virt-Manager is a graphical application for managing your virtual machines. You can use the kvm command directly, but libvirt and Virt-Manager simplify the process.

```
#sudo apt-get install qemu-kvm libvirt-bin bridge-utils virt-manager
```

```
root@ubuntu:/home/tsec# apt-get install qemu-kvm libvirt-bin bridge-utils virt-manager
```

3) Create User.

Only the root user and users in the libvирtd group have permission to use KVM virtual machines. Run the following command to add your user account to the libvирtd group:

```
#sudo adduser tsec  
#sudo adduser tsec libvирtd
```

After running this command, **log out** and **log back in as tsec**

```
root@ubuntu:/home/tsec# adduser tsec  
root@ubuntu:/home/tsec# adduser tsec libvирtd
```

4) Check whether everything is working correctly.

Run following command after logging back in as **tsec** and you should see an empty list of virtual machines.

This indicates that everything is working correctly.

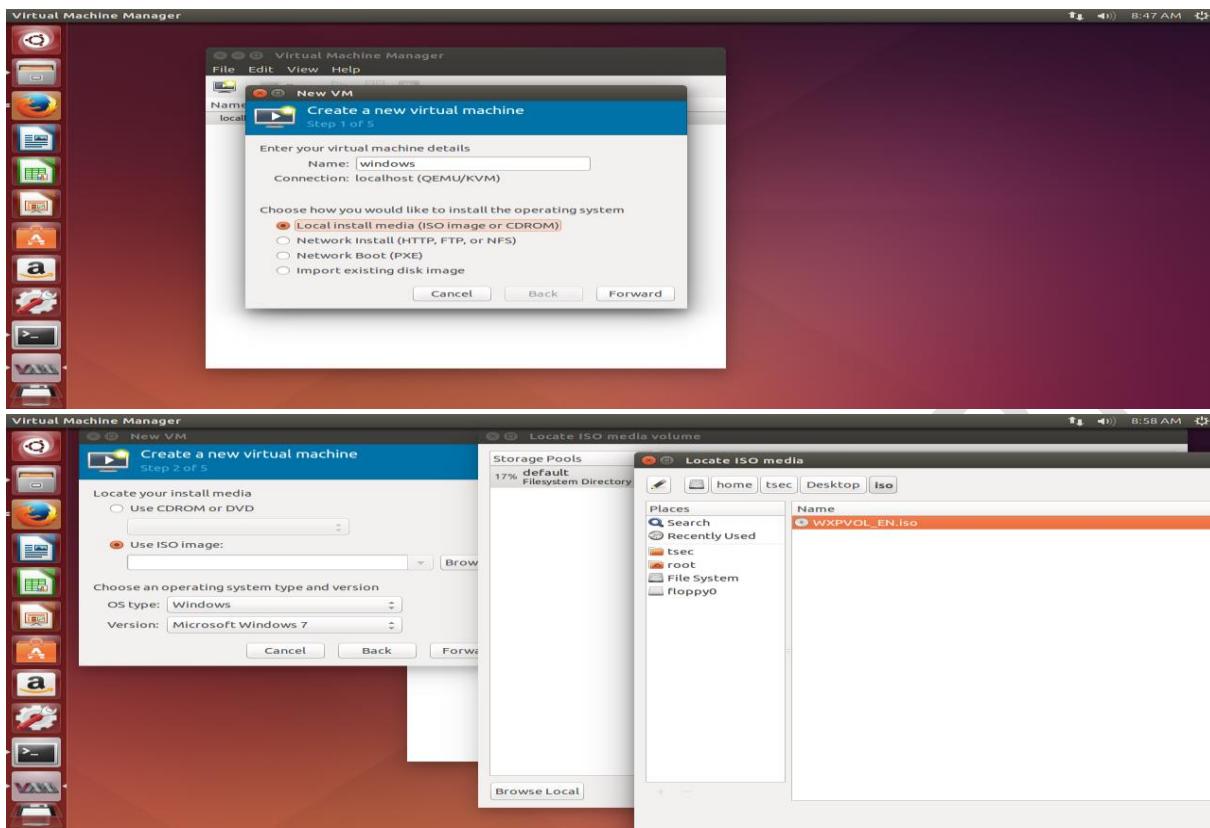
```
#virsh -c qemu:///system list
```

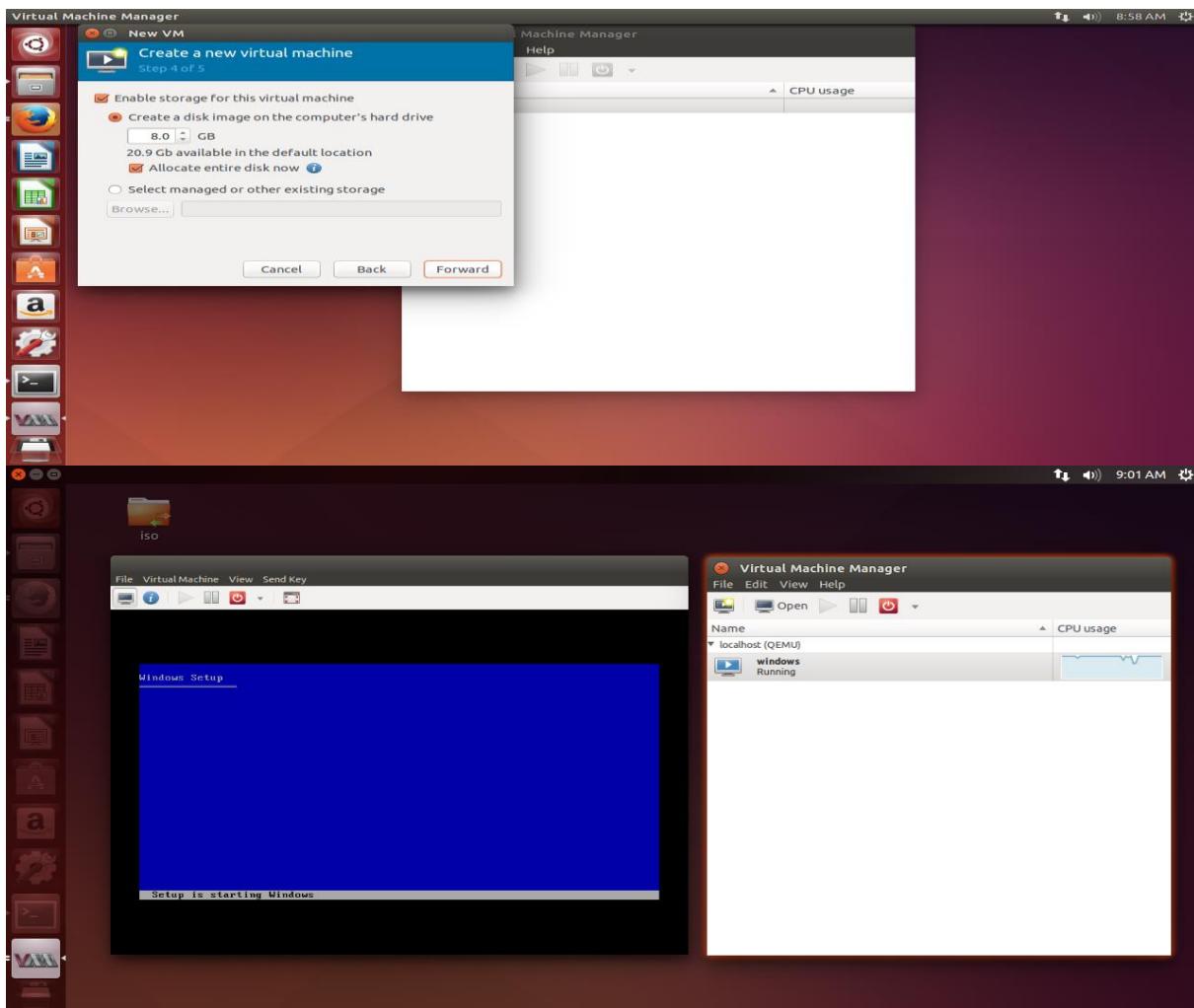
```
root@ubuntu:/home/tsec# virsh -c qemu:///system list  
  Id   Name      State  
-----  
root@ubuntu:/home/tsec#
```

5)Open Virtual Machine Manager application and Create Virtual Machine

```
#virt-manager  
root@ubuntu:/home/tsec# virt-manager
```

6) Create and run Virtual Machines





```
t$ sudo virsh list --all
```

Id	Name	State
1	ubuntu-vm	running

```
$ virsh
```

```
Welcome to virsh, the virtualization interactive terminal.
```

```
Type: 'help' for help with commands
      'quit' to quit
```

```
virsh #
```

```
virsh #
```

```
virsh # list --all
```

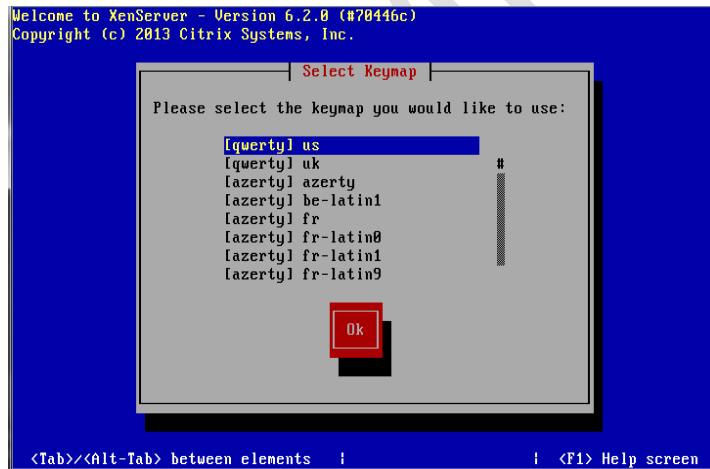
Id	Name	State
1	Windows	running

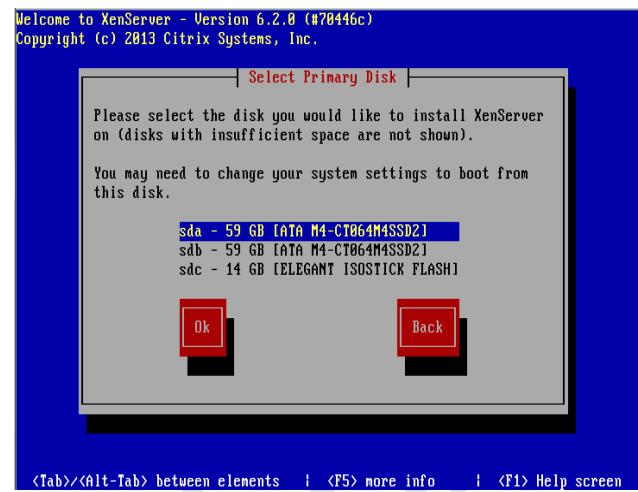
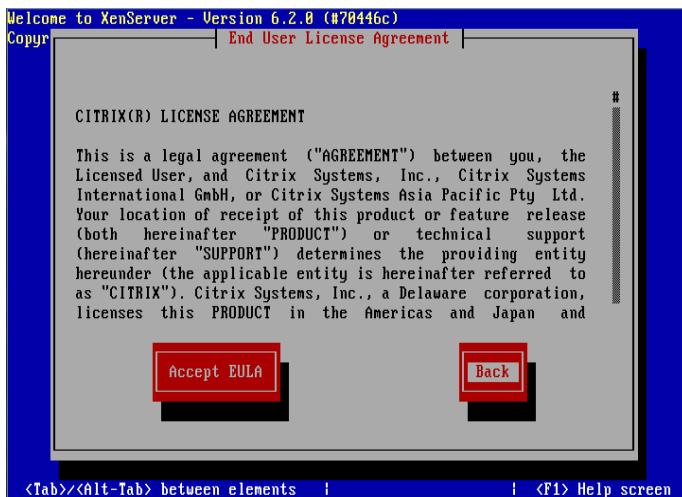
LAB 2**Creating and running virtual machines on Bare-Metal Hypervisor Xen Server****Step 1: Install Xen Server**

Step i-: Insert Bootable Xen Server CD into CDROM and Make first boot device as a CDROM from BIOS

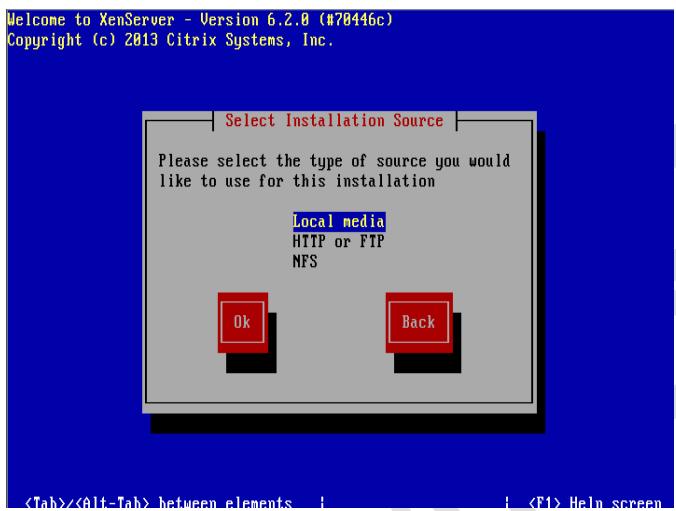


Step ii-: press F2 to see the advanced options, otherwise press Enter to start installation

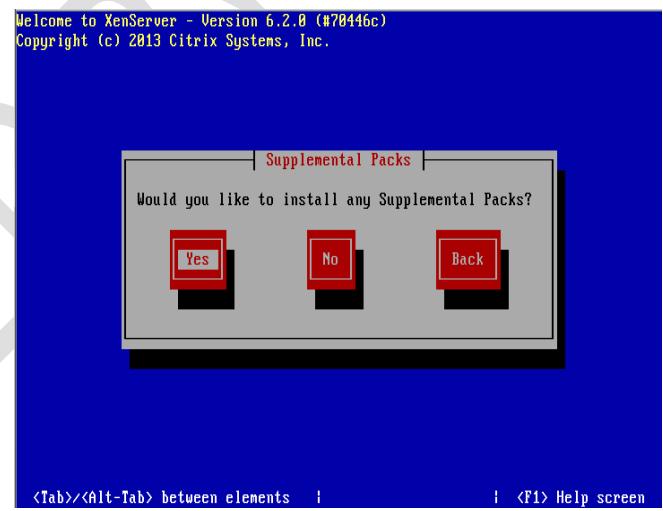
**Step iii :- Select Keyboard Layout****Step iv :-Press Enter to load Device Drivers****Step v :-Press Enter to Accept End user license Agreement****Step vi :-Select Appropriate disk on which you want to install Xen server**



Step vii :-Select Appropriate installation Media



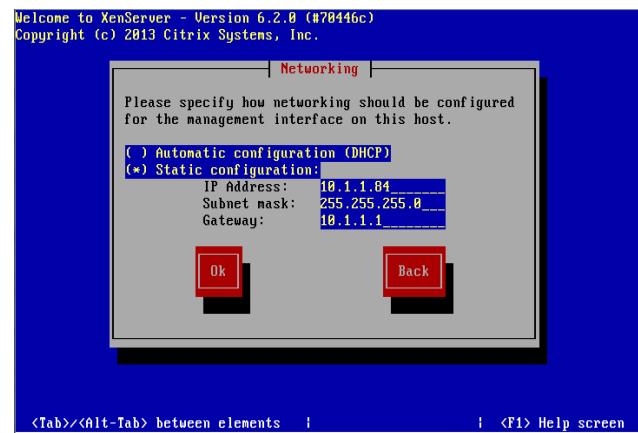
Step viii :-Select Additional Packages for installation



Step ix:- Specify Root password



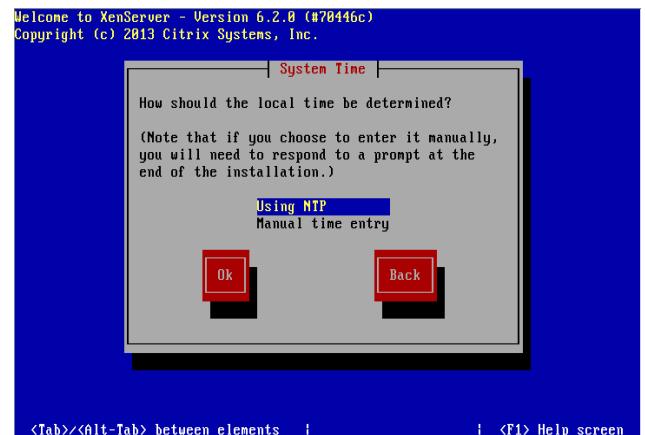
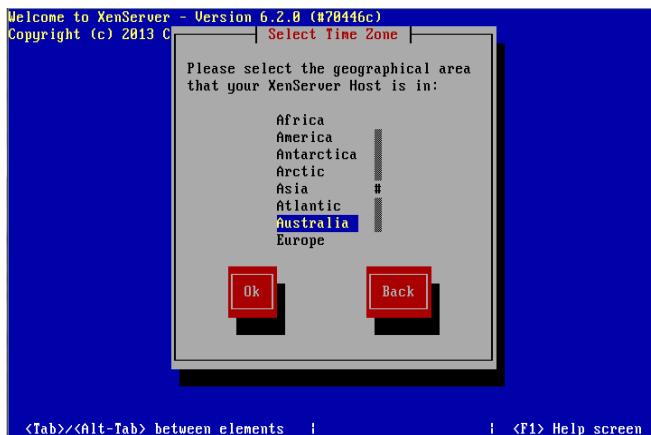
Step x :- Specify IP Address to a Xen Server



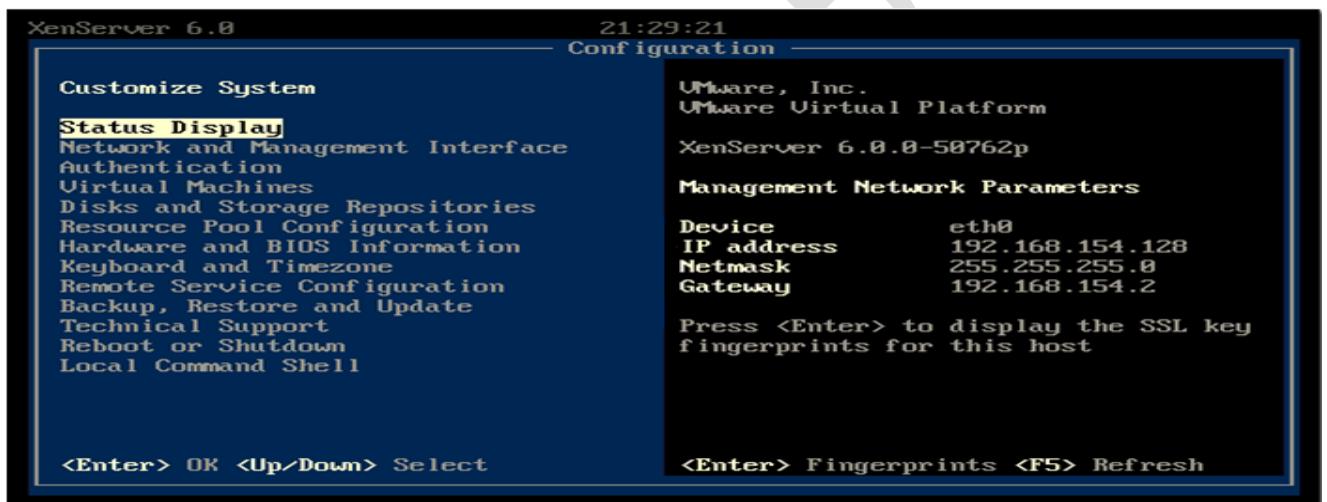
Step xi:-Select Time Zone

Step xii:-Specify NTP Servers address or use manual

time entry then start installation



Once installation is done you will see the final screen shown below.



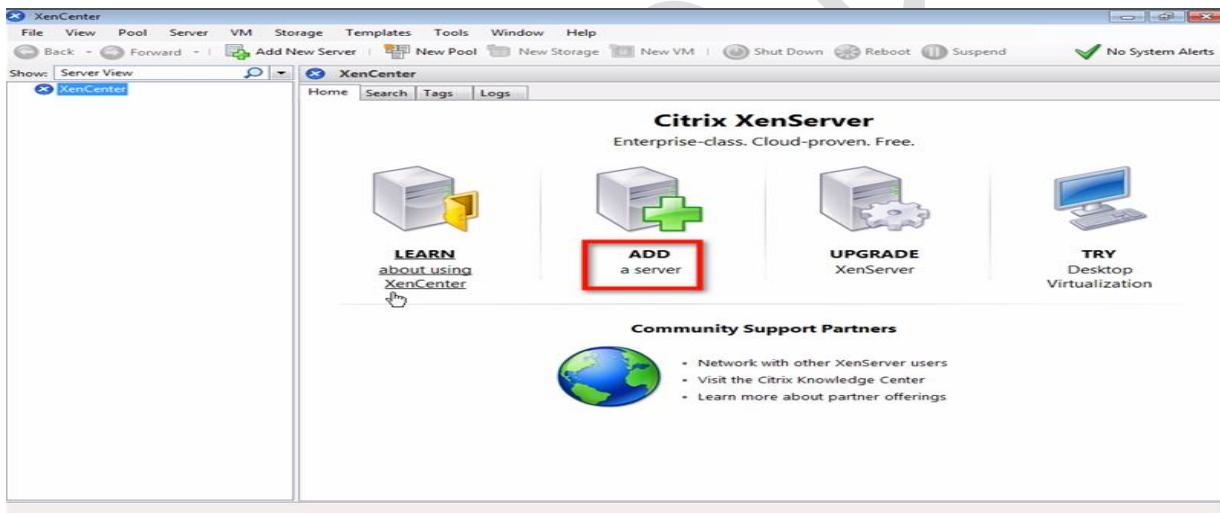
Xen Server Final Screenshot

Step 2: Connect Xen Server to Xen Center

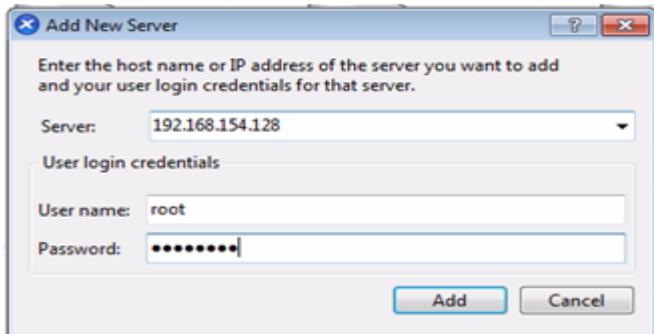
Firstly, download the xen center a management utily from xen server by opening the xen servers IP address as a URL on browser. Once Xen center is downloaded, install it. Open Xen center from start menu of Windows.



Here's how XenCenter looks like (see screenshot below) before any hosts, resource pools, and so on, are added to it. To connect to the XenServer host you configured earlier, click Add a server.



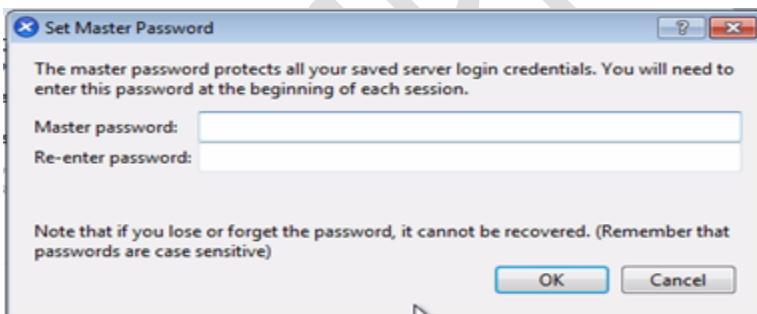
Enter the IP address I asked you to take note of earlier. Also enter the password you assigned for your root account. Click Add.



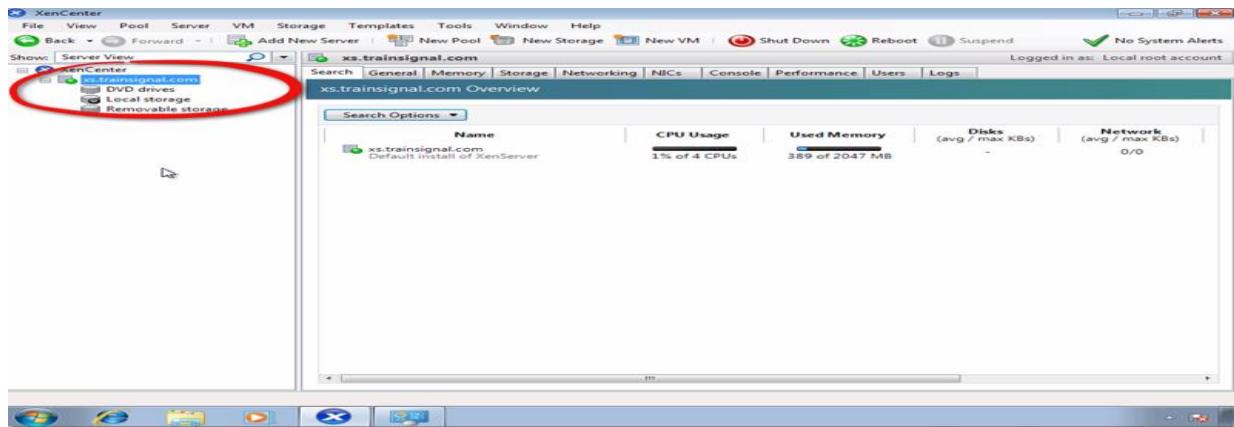
One of the first things you want to make sure as you're adding a new XenServer to XenCenter is to save and restore the server connection state on startup. Check the box that will do just that.



Once you do that, you will be allowed to configure a master password for all the XenServers you'll be associating with this XenCenter. Click the Require a master password checkbox if that's what you want to do, and then enter your desired master password in the fields provided.

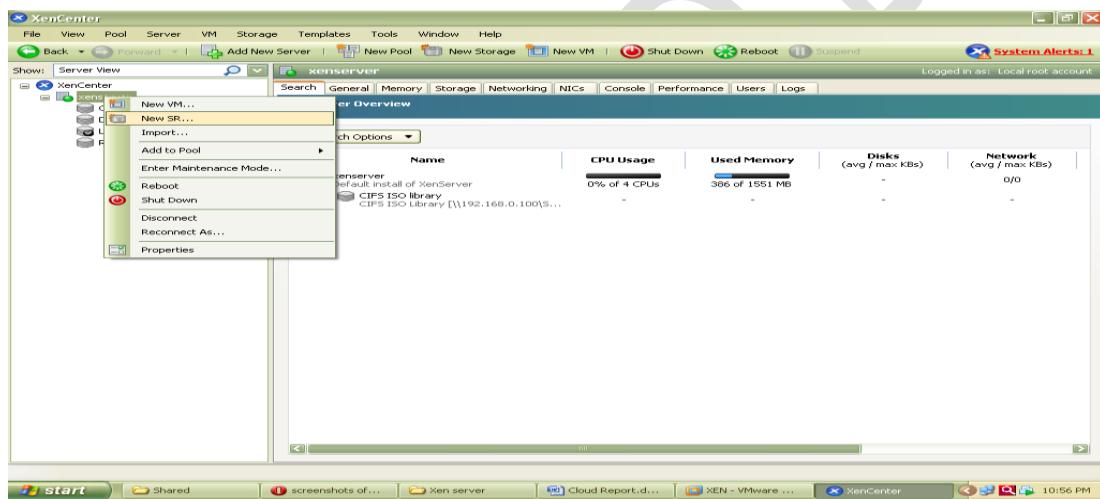


After you click OK, you'll be brought back to the main screen, where you'll see your XenServer already added to XenCenter.

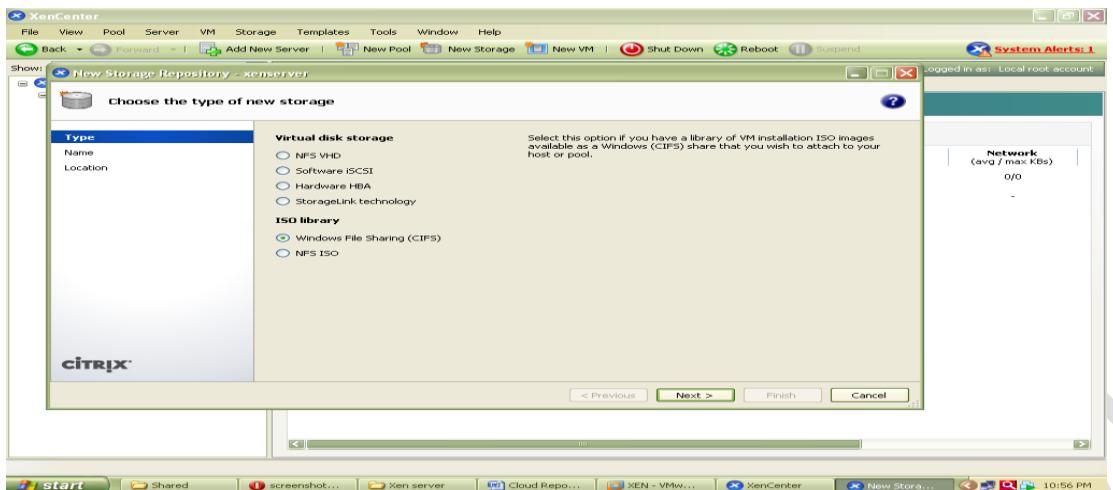


Step-:3 Create Storage Repository and Installing VM

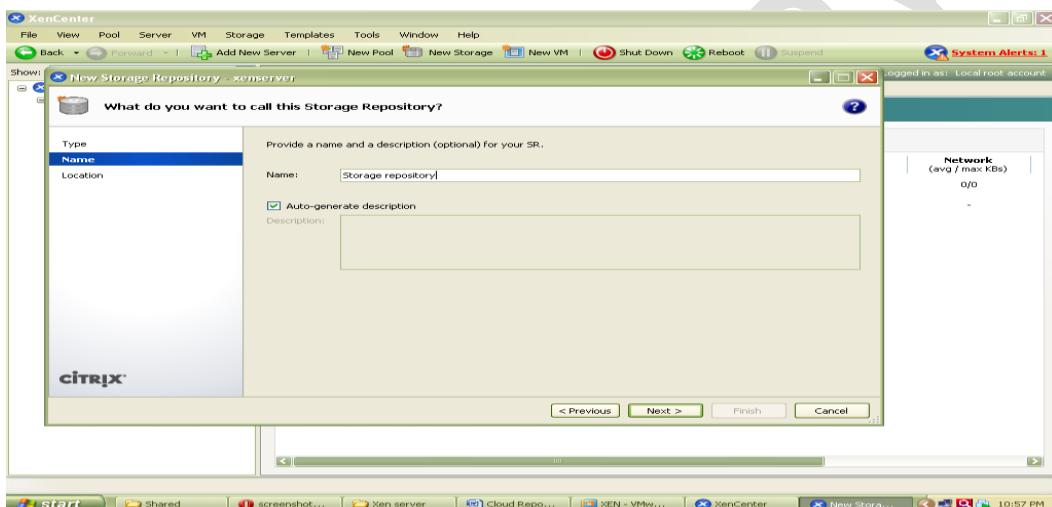
Now Before Creating VM we have to Create Storage Repository first which is nothing but shared directory on Xen Center which holds all iso files and which is required to install Operating system on Xen Server its steps are as follows.Right click on Xenserver icon on xen center and click on New SR



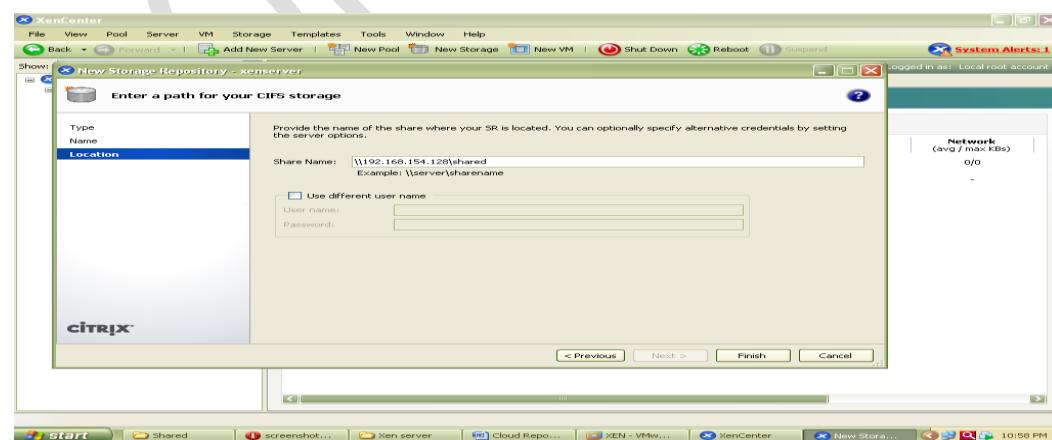
Now Select Windows CIFS library



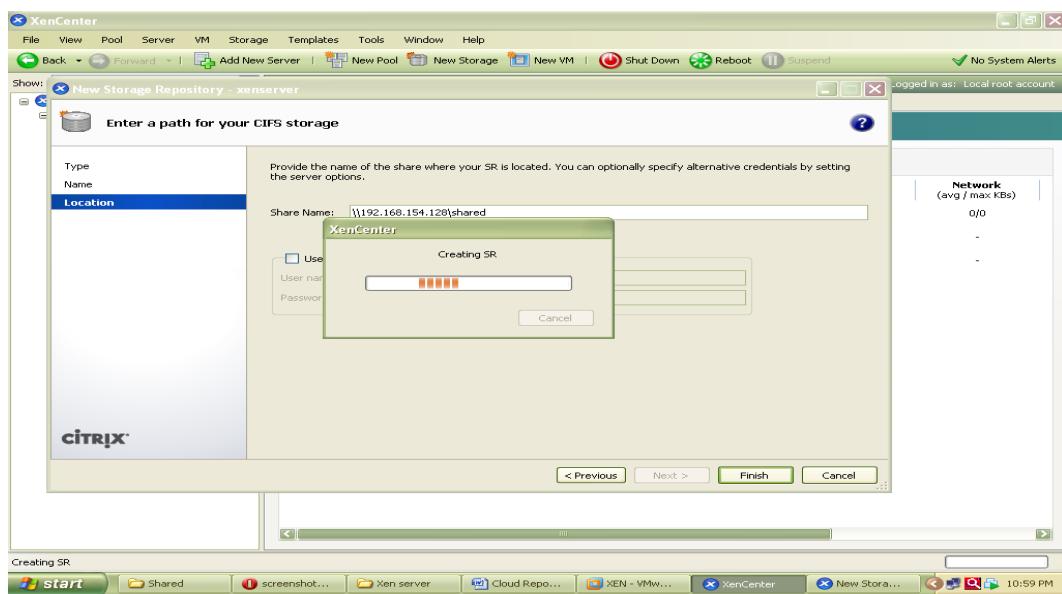
Specify Storage Repository Name



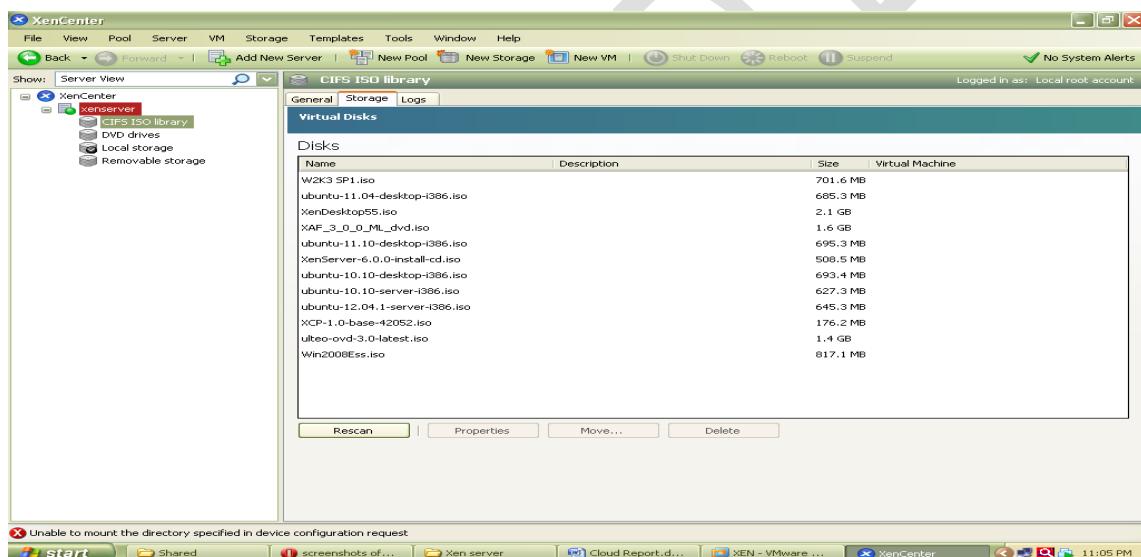
Now specify path of shared folder at client side which holds all iso files of os or VM which we are going to install on Xen Server.



At the end Click on finish to create SR.

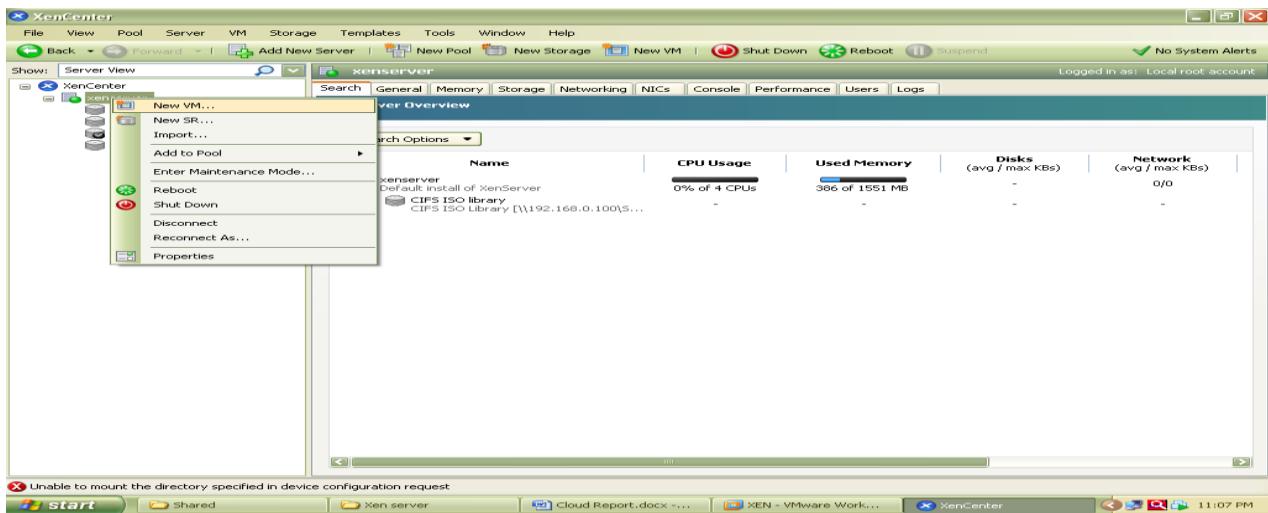


To check all iso files click on CIFS library and select storage this will show you all iso files.

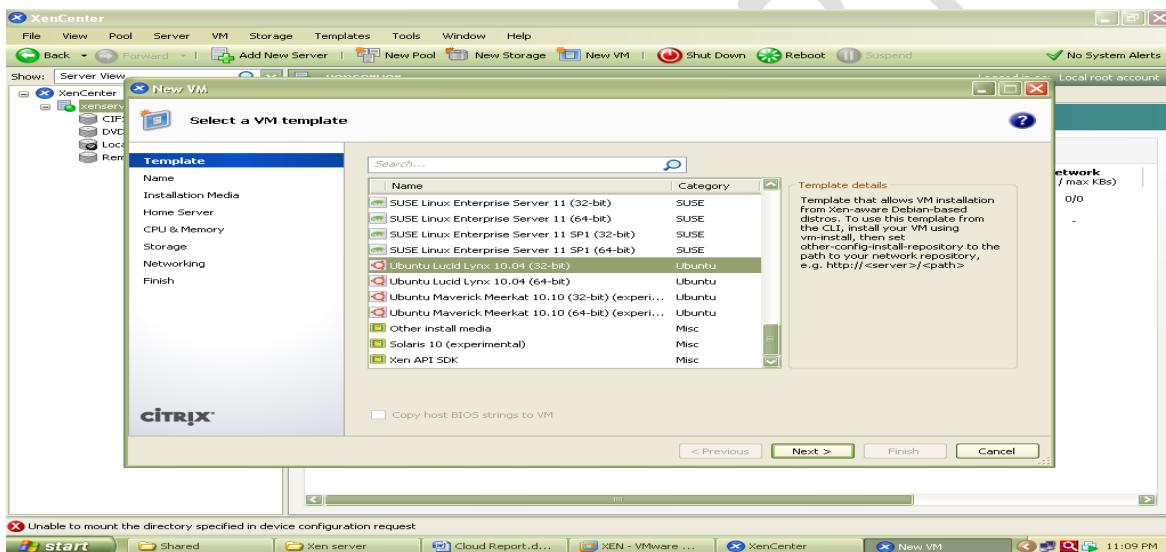


Installation of UBUNTU Server on Xen Server

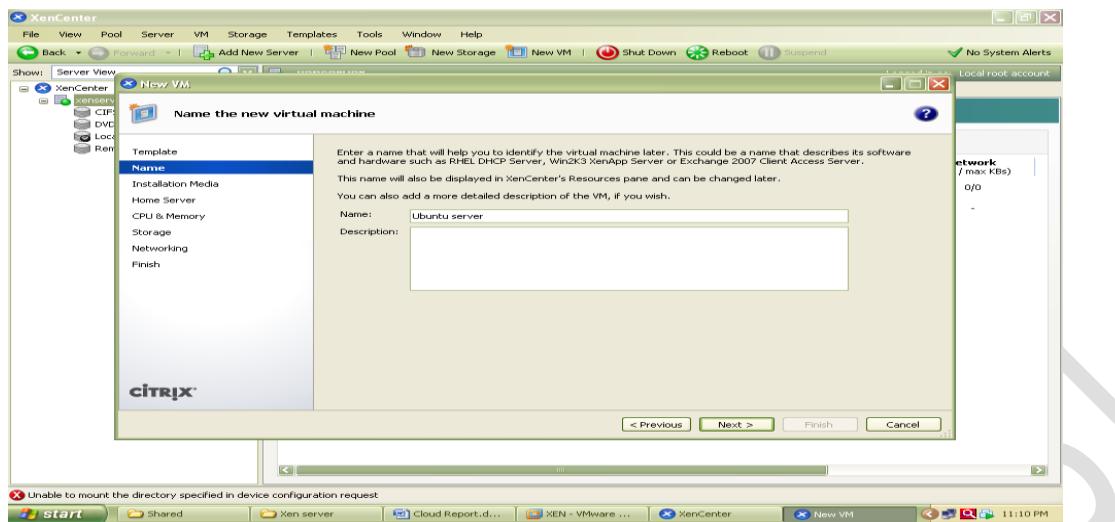
Step 1 :- Right click on Xenserver icon on xen center and select New VM



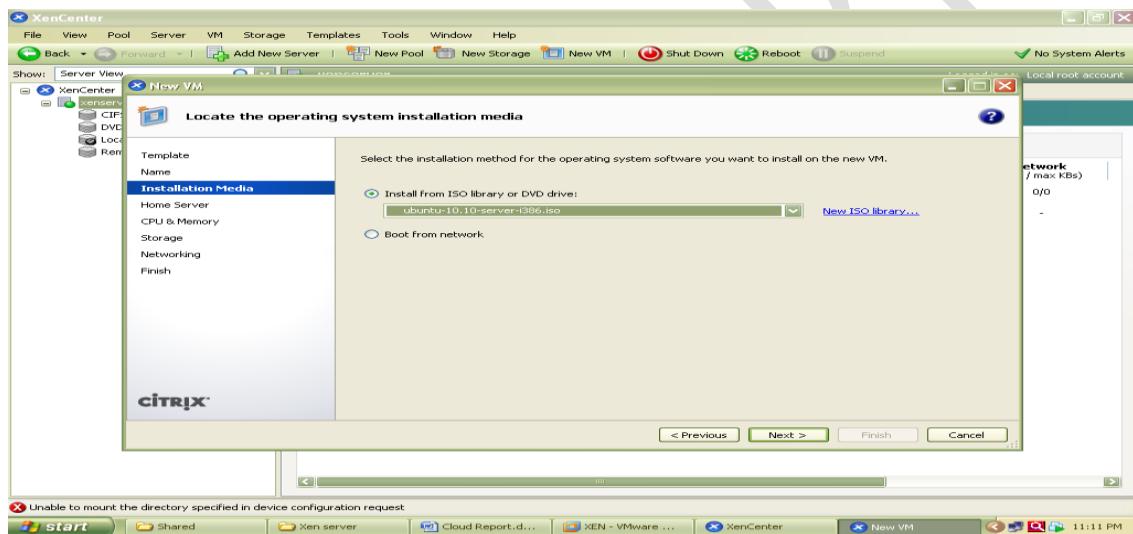
Now select an Operating System to be install here select Ubuntu Lucid Lynx and click on next



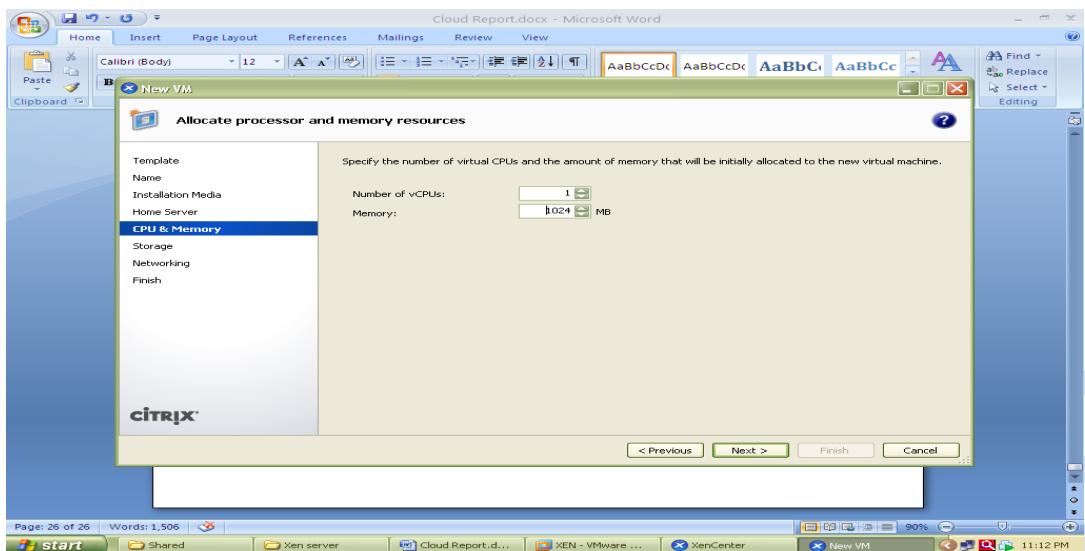
Now specify Instance Name as ubuntu server



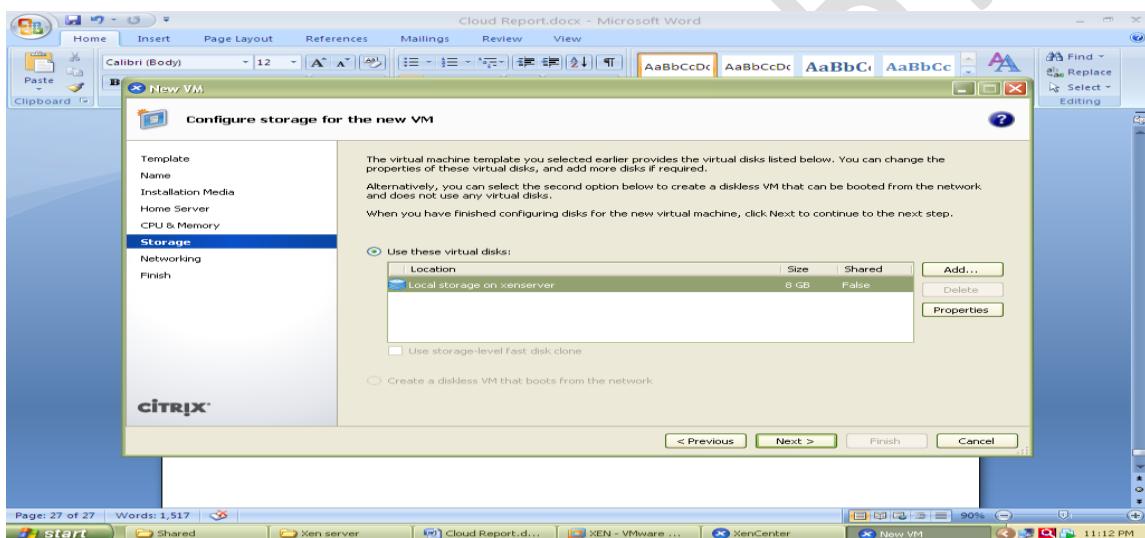
Select iso file of Ubuntu server 10.10 to be install



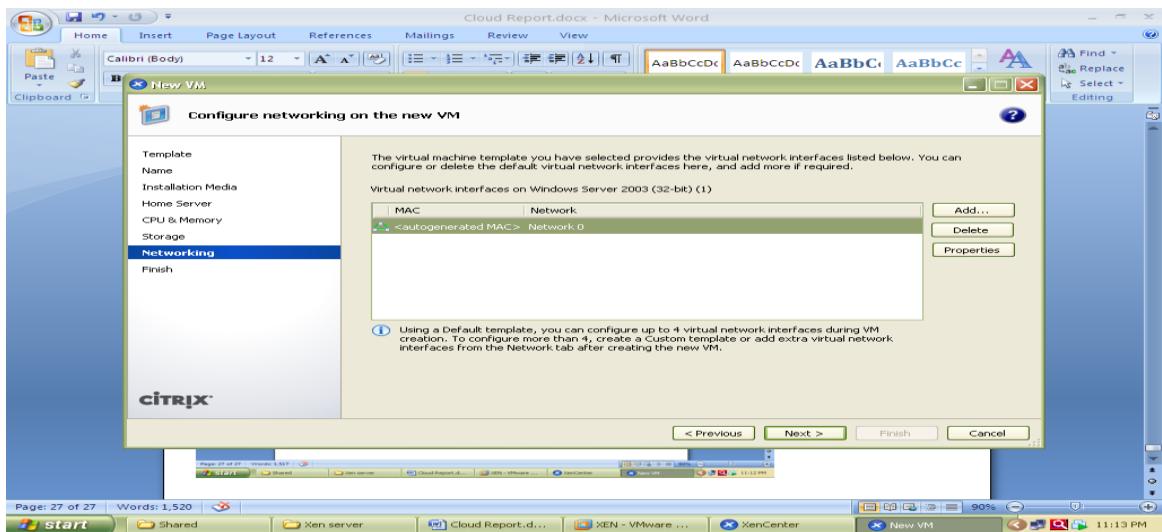
Now select hardware for vm i.e. no. of cpu's and memory



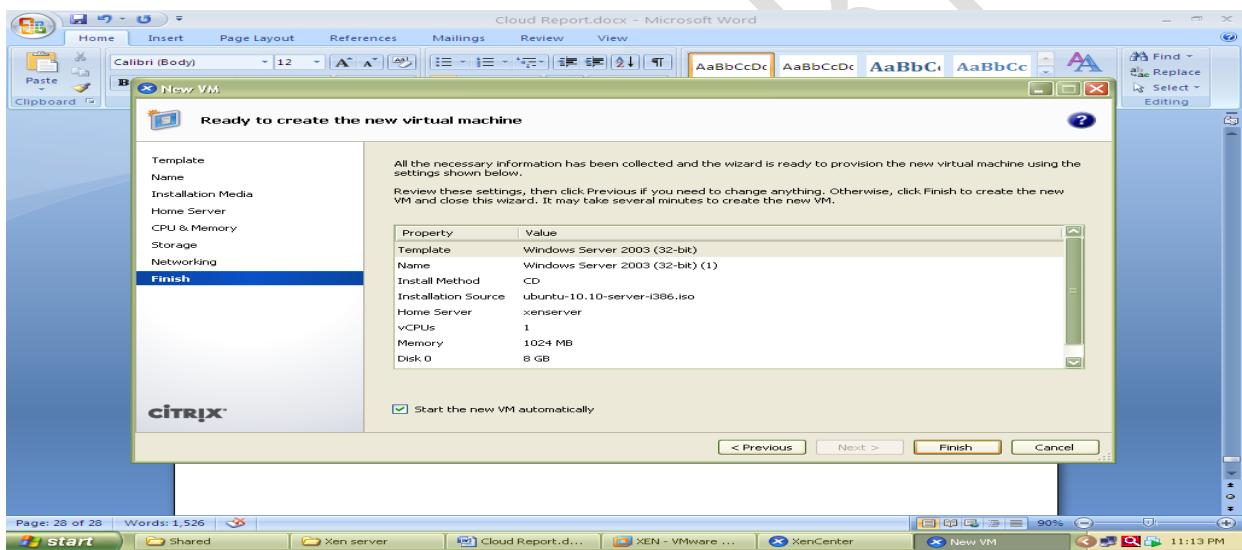
Select local storage



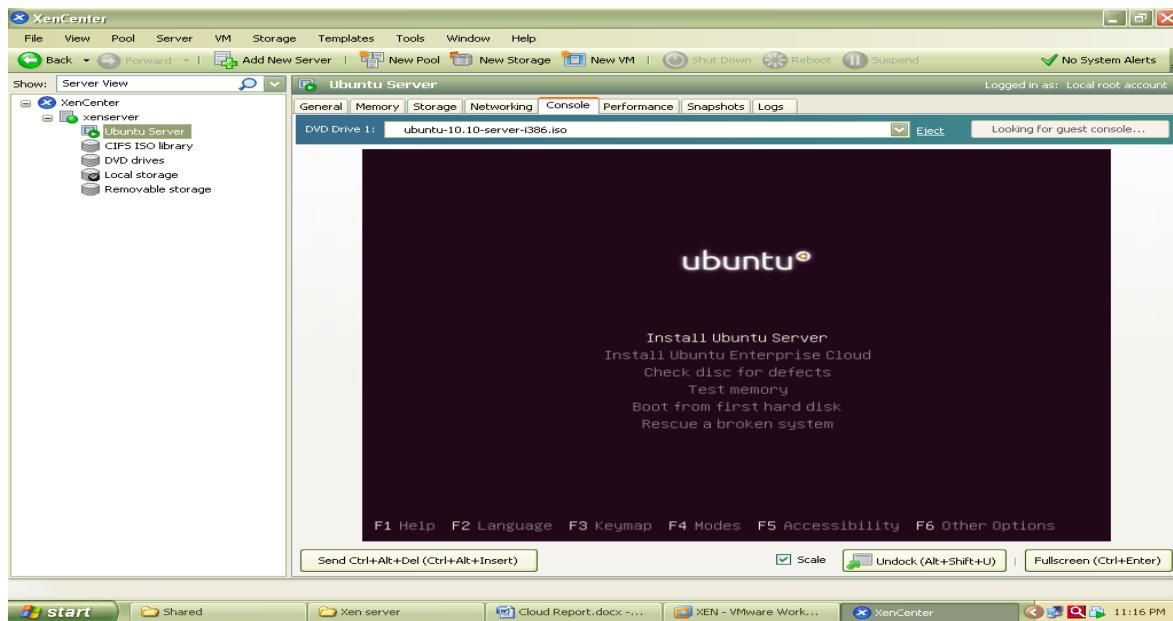
Select network



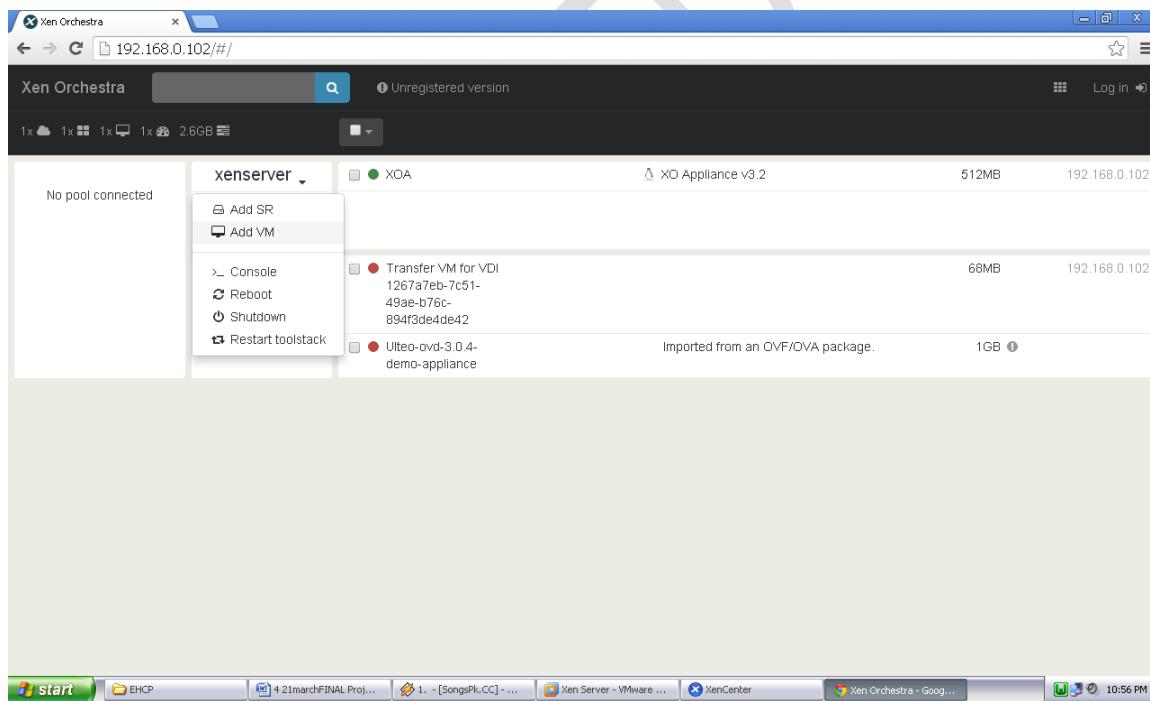
And click on finish



Now go to Console tab to install ubuntu and follow installation Steps.

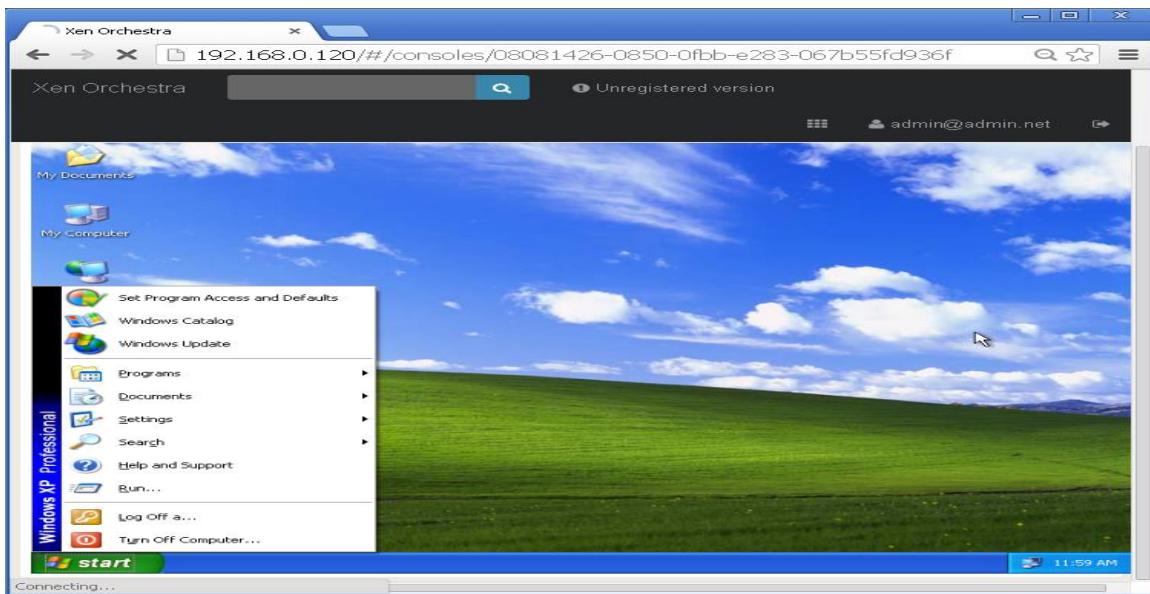


The Xen orchestra provides web based functionality of Xen Center.it provides access to all the VMs with their lifecycle management which are installed over Xen Server shown in figure 5.28



Xen Orchestra (XOA) Portal

The Windows XP image running on Xen Orchestra over Google chrome web browser is shown in following screenshot



Windows XP running on Xen orchestra (XOA)

LAB 3	Installation and Configuration of Ulteo to demonstrate on demand Application delivery over web browser to explore SaaS Environment.
--------------	--

Ulteo is an open source Virtual Desktop infrastructure project that can deliver various operating systems desktops - including Windows and Linux desktops or applications - to end users. The Open Virtual Desktop allows corporates to deploy virtualized GNU/Linux and/or Windows desktops. Parts of Ulteo products are based on Debian and Ubuntu. Ulteo Open Virtual Desktop is an open source alternative to Citrix and VMWare solutions.

The steps for installation and configuration of Ulteo are given as follows

Step 1: Install Ulteo through DVD or Open Ulteo OVF file in Vmware player by selecting import VM button

1) If you haven't an Ulteo OVD DVD-ROM yet, please download the corresponding ISO file from this place at www.ulteo.com and burn it to a fresh DVD.

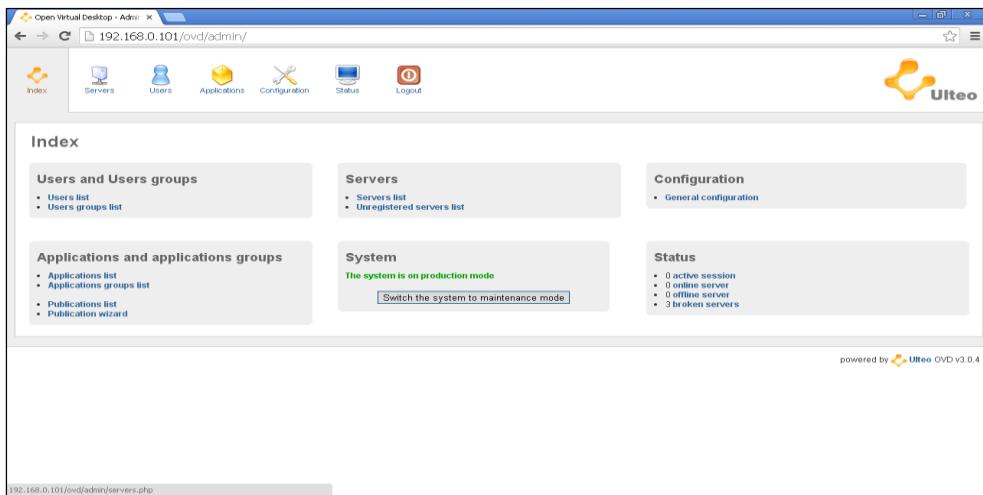
- 2) Insert the Ulteo OVD DVD-ROM into your computer and restart it. If you selected the DVD-ROM as first boot device you'll see the boot loader Screen.
- 3) Select Install Ulteo Option
- 4) The first step is used to select the system language. Choose your language from the list and click on Forward.
- 5) In the second step, the system asks you to define your location. Either select a point on the map or choose one from the Selected city form and click on Forward.
- 6) The third step is used to define the keyboard layout. Select yours and click on Forward.
- 7) Then, you have to select the partitioning method. We suggest the automatic method: Erase and use the entire disk.
- 8) These questions are about the installed operating system itself, user login and password used to access the OS, along with the hostname of the machine.
- 9) Type a password and confirm it. Useful address is displayed to you for a near future use of OVD.
- 10) Then read carefully the installation summary, then click on Install and wait til installation completes
- 11) Finally, click on Restart now to finish installation process.

Step 2: In Management machine Open following URLs

<https://Ulteo-Server-ipaddress/ovd> for Client access

<https://Ulteo-Server-ipaddress/admin> for Admin access

Step 3: Login on Admin portal specify Username and Password as Admin



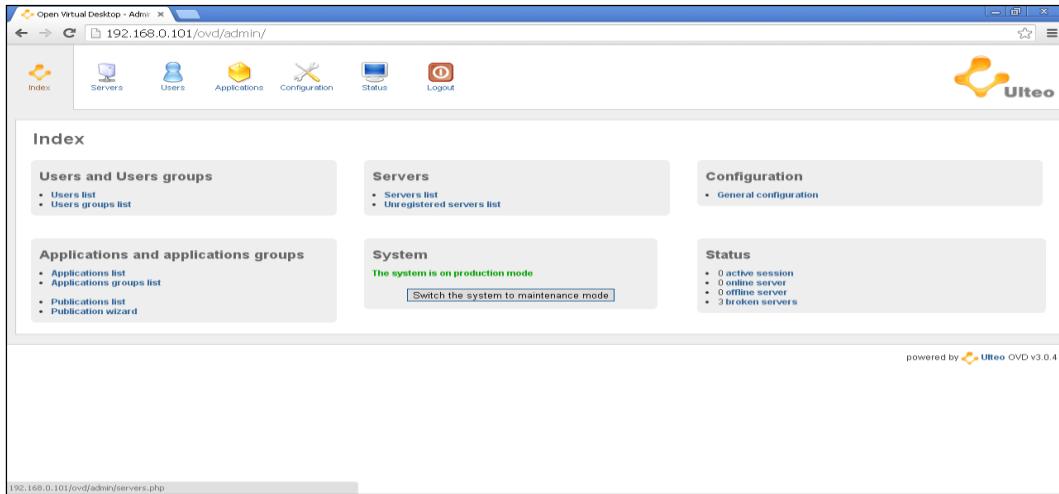
Under server tab Register server, Click on manage to add ip address of Ulteo Server

- Go to user tab to add multiple users
- Go to User tab then select user group then create a new user group and add users in to them
- Go to Application Tab to Create Application Group

Applications					
Applications Groups		Name	Description	Type	
Mime-Types		Adobe Reader 9	PDF Viewer	Linux	Manage
Static applications		Bulk Rename	Rename Multiple Files	Linux	Manage
Publications		Firefox Web Browser	Browse the World Wide Web	Linux	Manage
Publication wizard		GIMP Image Editor	Create Images and edit photographs	Linux	Manage
		Mousepad	Simple text editor	Linux	Manage
		OpenOffice.org Database	Manage databases, create queries and reports to track and manage your information.	Linux	Manage
		OpenOffice.org Drawing	Create and edit drawings, flow charts, and logos.	Linux	Manage
		OpenOffice.org Formula	Create and edit scientific formulas and equations.	Linux	Manage
		OpenOffice.org Impress	Create and edit presentations for slideshows, meeting and Web pages.	Linux	Manage

Map User group with Application group And use the services at client side

The Administrator panel is limited to Administrator who can manage Applications, users and groups. Once admin logged in to this portal, he can create users, user groups, Application groups maps users to User group and Application group, manages applications or installs softwares based on users requirement Shown in figure 5.11.



Ulteo Administrator panel

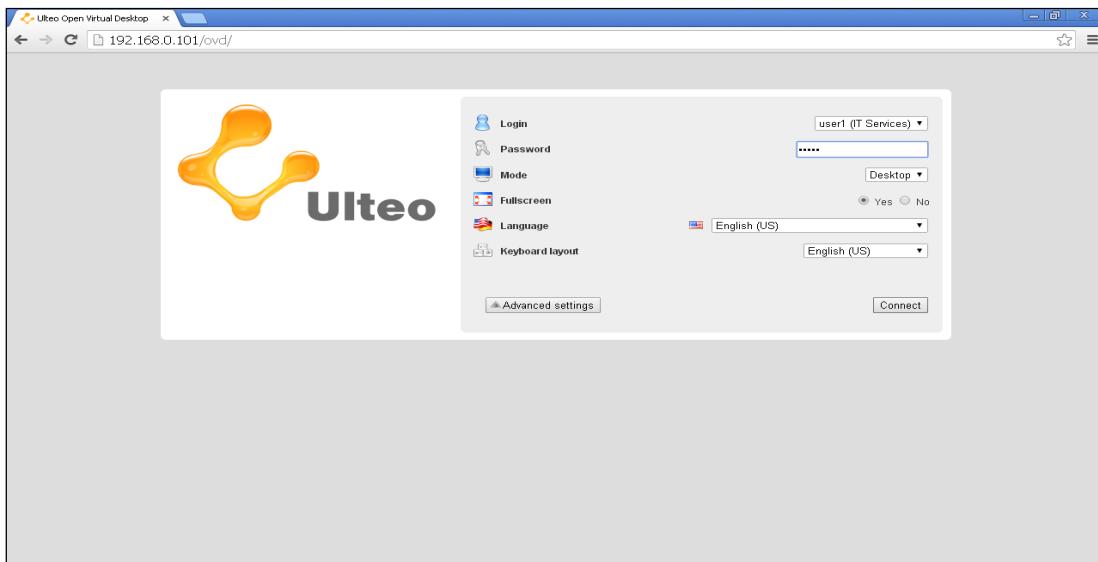
The Application menu of admin panel shows available applications which can be mapped to users or user groups Shown in

Applications			
Applications Groups	Name	Description	Type
Mime-types	Adobe Reader 9	PDF Viewer	linux
Static applications	Bulk Rename	Rename Multiple Files	linux
Publications	Firefox Web Browser	Browse the World Wide Web	linux
Publication wizard	GIMP Image Editor	Create images and edit photographs	linux
	Mousepad	Simple text editor	linux
	OpenOffice.org Database	Manage databases, create queries and reports to track and manage your information.	linux
	OpenOffice.org Drawing	Create and edit drawings, flow charts, and logos.	linux
	OpenOffice.org Formula	Create and edit scientific formulas and equations.	linux
	OpenOffice.org Impress	Create and edit presentations for slideshows, meeting and Web pages.	linux

Ulteo Application Menu

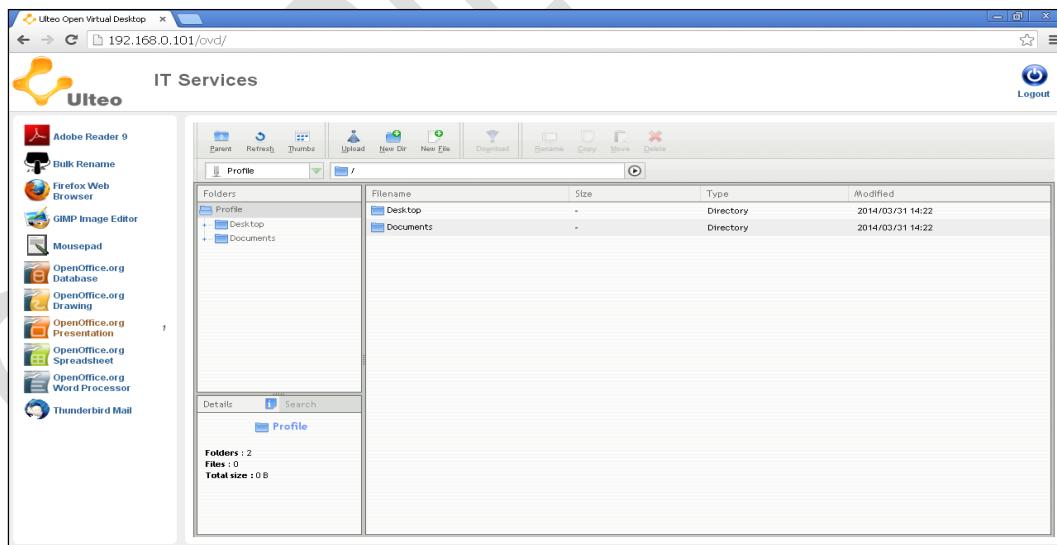
Step 4: At client side open <https://Ulteo-Server-ipaddress/ovd> for Client access,Specify Username and Password and Access the softwares added in Application group

Once user selects Access Ulteo option it shows login page of Ulteo session manager shown below. The user can get login name and password by filling Registration form through main page of cloud portal Shown below.



Ulteo user Login Portal

Once user is validated he can access the services using portal mode or Desktop mode. Both the modes give access to software applications which are installed on Linux Application Server and Windows Application Server. In portal mode the user get applications in vertical pane Shown in figure 5.9.



Ulteo Portal mode

While in Desktop mode use gets full flagged Linux desktop running on browser with selected Applications Shown in



Ulteo Desktop mode

LAB 4	To demonstrate installation and Configuration of Open stack Private cloud.
--------------	---

The OpenStack installation can be done using many ways like RDO Pack stack, Mirantis or Devstack who have series of shell scripts which carries automated installation of OpenStack. The DevStack is a series of extensible scripts used to quickly bring up a complete OpenStack environment based on the latest versions of everything from git master.

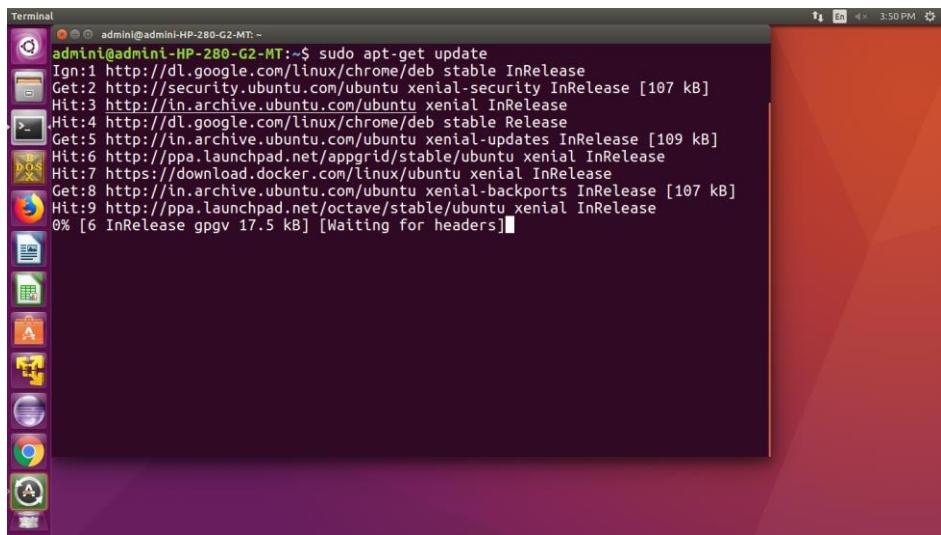
To install OpenStack using Devstack the Prerequisites are Intel or AMD Multicore CPU, Minimum 6-8GB RAM, 250 GB Hard disk and preinstalled Ubuntu server/Desktop Operating system version 16.04 or above and internet speed should be minimum 4 MBPS. (The installation steps can be found at <https://docs.openstack.org/devstack/latest/>)

The steps for installing Openstack using Devstack in a single server (All in one Single machine setup) are given as follows

Step 1:- Update the ubuntu repository and install git package

The current version of Ubuntu OpenStack is Newton. So, that's what we are going to install. To begin with the installation, first, we need to use the git command to clone devstack.

```
$sudo apt-get update  
$sudo apt-get install git
```

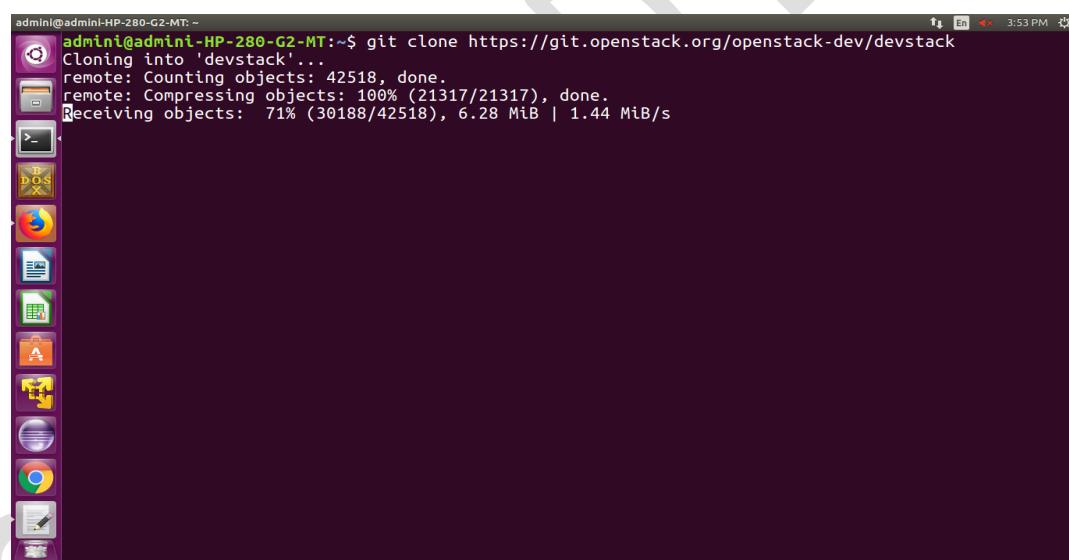


A screenshot of an Ubuntu desktop environment. A terminal window is open in the top-left corner, showing the command `sudo apt-get update` being run. The output of the command is displayed, listing various package sources and their status. The desktop background is a standard Ubuntu orange gradient. A vertical dock on the left contains icons for various applications like Dash, Home, and System Settings.

```
Terminal
admin@admini-HP-280-G2-MT:~$ sudo apt-get update
Ign:1 http://dl.google.com/linux/chrome/deb stable InRelease
Get:2 http://security.ubuntu.com/ubuntu xenial-security InRelease [107 kB]
Hit:3 http://in.archive.ubuntu.com/ubuntu xenial InRelease
Hit:4 http://dl.google.com/linux/chrome/deb stable Release
Get:5 http://in.archive.ubuntu.com/ubuntu xenial-updates InRelease [109 kB]
Hit:6 http://ppa.launchpad.net/appgrid/stable/ubuntu xenial InRelease
Hit:7 https://download.docker.com/linux/ubuntu xenial InRelease
Get:8 http://in.archive.ubuntu.com/ubuntu xenial-backports InRelease [107 kB]
Hit:9 http://ppa.launchpad.net/octave/stable/ubuntu xenial InRelease
0% [6 InRelease gpgv 17.5 kB] [Waiting for headers]
```

Step 2 :- Download the latest git repository for openstack

```
$ git clone https://git.openstack.org/openstack-dev/devstack
```

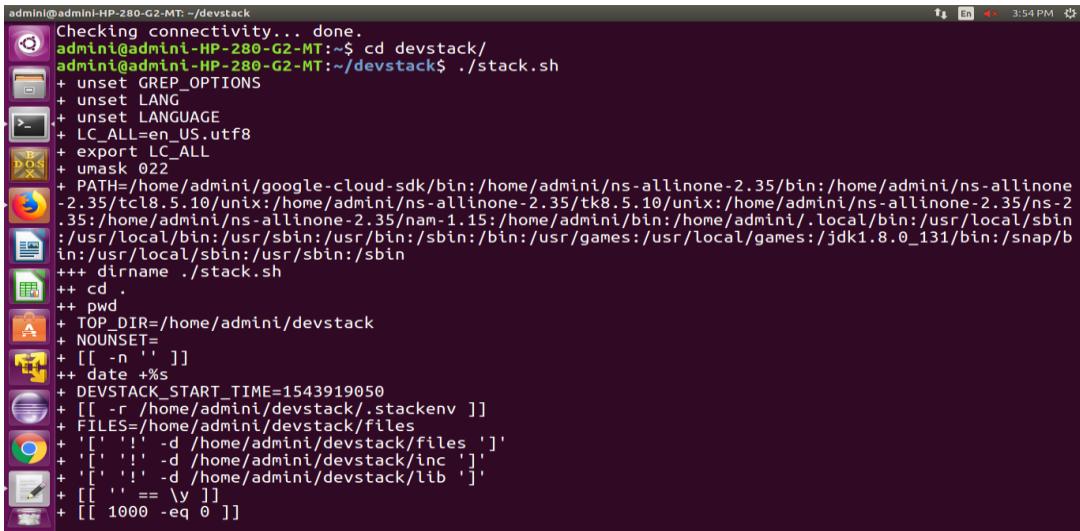


A screenshot of an Ubuntu desktop environment. A terminal window is open in the top-left corner, showing the command `git clone https://git.openstack.org/openstack-dev/devstack` being run. The output of the command shows the cloning process, including object counting, compression, and receiving objects. The desktop background is a standard Ubuntu orange gradient. A vertical dock on the left contains icons for various applications like Dash, Home, and System Settings.

```
Terminal
admin@admini-HP-280-G2-MT:~$ git clone https://git.openstack.org/openstack-dev/devstack
Cloning into 'devstack'...
remote: Counting objects: 42518, done.
remote: Compressing objects: 100% (21317/21317), done.
Receiving objects: 71% (30188/42518), 6.28 MiB | 1.44 MiB/s
```

Step 3:- Open Devstack directory and start installation by executing stack.sh shell script

```
$cd Devstack
$./stack.sh
```

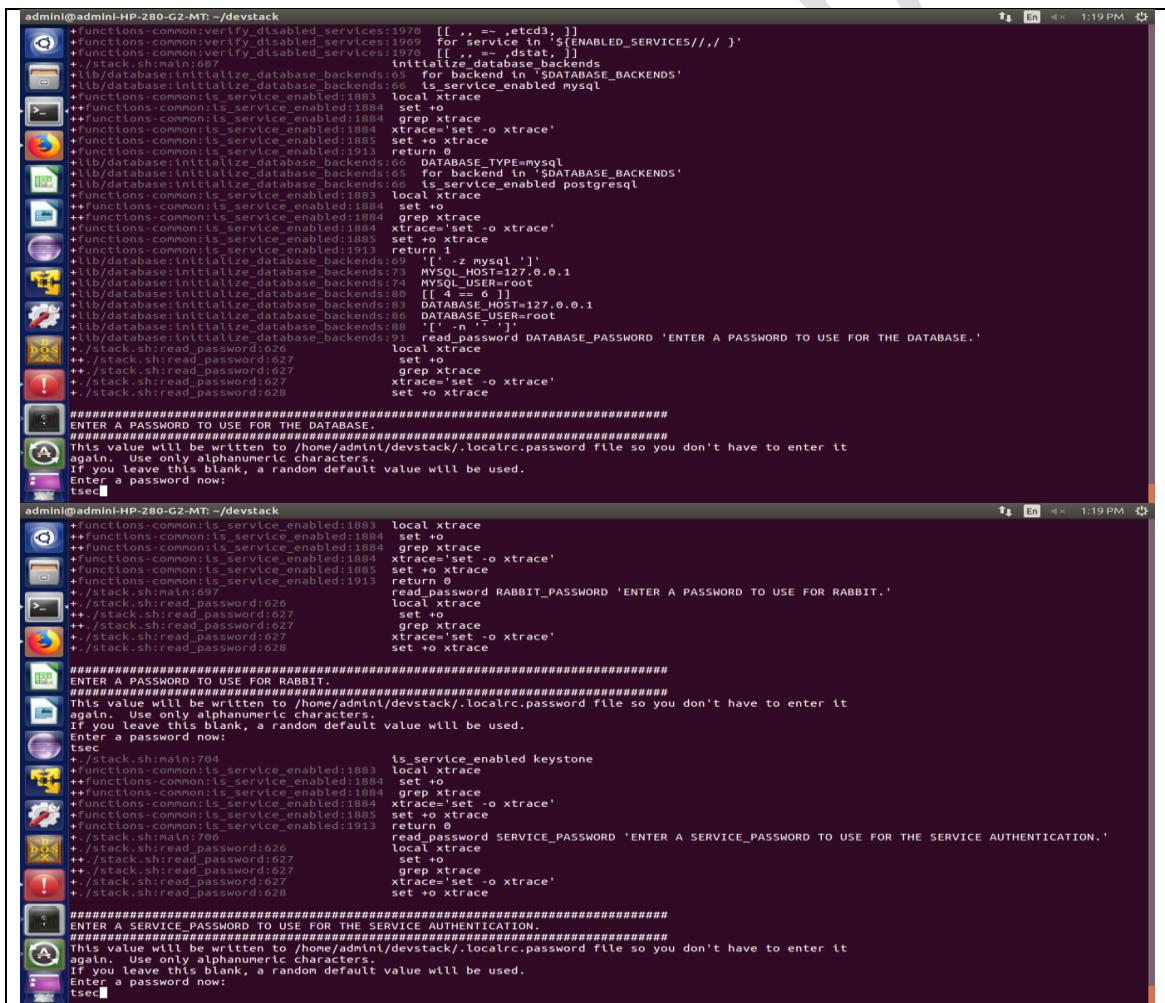


```

admini@admini-HP-280-G2-MT:~/devstack
Checking connectivity... done.
admini@admini-HP-280-G2-MT:~$ cd devstack/
admini@admini-HP-280-G2-MT:~/devstack$ ./stack.sh
+ unset GREP_OPTIONS
+ unset LANG
+ unset LANGUAGE
+ LC_ALL=en_US.utf8
+ export LC_ALL
+ umask 022
+ PATH=/home/admini/google-cloud-sdk/bin:/home/admini/ns-allinone-2.35/bin:/home/admini/ns-allinone
2.35/tcl8.5.10/unix:/home/admini/ns-allinone-2.35/tk8.5.10/unix:/home/admini/ns-allinone-2.35/ns-2
35:/home/admini/ns-allinone-2.35/nam-1.15:/home/admini/bin:/home/admini/.local/bin:/usr/local/sbin
in:/usr/local/sbin:/usr/sbin:$sbin
+++ dirname ./stack.sh
++ cd .
++ pwd
+ TOP_DIR=/home/admini/devstack
+ NOUNSET=
+ [[ -n '' ]]
+ date +%
+ DEVSTACK_START_TIME=1543919050
+ [[ -r /home/admini/devstack/.stackenv ]]
+ FILES=/home/admini/devstack/files
+ '[' '!' -d /home/admini/devstack/files ']'
+ '[' '!' -d /home/admini/devstack/inc ']'
+ '[' '!' -d /home/admini/devstack/lib ']'
+ [[ '' == \y ]]
+ [[ 1000 -eq 0 ]]

```

At the initial stage, the installer will ask passwords for database, rabbit, service authentication, horizon and keystone.



```

admini@admini-HP-280-G2-MT:~/devstack
+functions-common:verify_disabled_services:1970 [[ , =~ _etcd3 ]]
+functions-common:verify_disabled_services:1970 for service in ${ENABLED_SERVICES//,/ }
+functions-common:verify_disabled_services:1970 [[ , =~ _dstat, ]]
+./stack.sh:main:687 initialize_database_backends
+lib/database:initialize_database_backends:65 for backend in ${DATABASE_BACKENDS}
+lib/database:initialize_database_backends:65 local_xtrace
+functions-common:is_service_enabled:1883 service_enabled mysql
+functions-common:is_service_enabled:1883 local_xtrace
+set +o
+functions-common:is_service_enabled:1884 grep xtrace
+functions-common:is_service_enabled:1884 xtrace=set -o xtrace
+functions-common:is_service_enabled:1884 set +o xtrace
+functions-common:is_service_enabled:1913 return 0
+lib/database:initialize_database_backends:66 DATABASE_TYPE=mysql
+lib/database:initialize_database_backends:65 for backend in ${DATABASE_BACKENDS}
+lib/database:initialize_database_backends:65 local_xtrace
+functions-common:is_service_enabled:1883 service_enabled postgresql
+functions-common:is_service_enabled:1883 local_xtrace
+set +o
+functions-common:is_service_enabled:1884 grep xtrace
+functions-common:is_service_enabled:1884 xtrace=set -o xtrace
+functions-common:is_service_enabled:1884 set +o xtrace
+functions-common:is_service_enabled:1913 return 0
+lib/database:initialize_database_backends:69 '[' -z mysql ']'
+lib/database:initialize_database_backends:73 MYSQL_HOST=127.0.0.1
+lib/database:initialize_database_backends:73 MYSQL_USER=root
+lib/database:initialize_database_backends:80 [[ $4 -eq 0 ]]
+lib/database:initialize_database_backends:80 DATABASE_HOST=127.0.0.1
+lib/database:initialize_database_backends:80 DATABASE_USER=root
+lib/database:initialize_database_backends:80 [[ $4 -ne 0 ]]
+./stack.sh:read_password:626 read_password DATABASE_PASSWORD 'ENTER A PASSWORD TO USE FOR THE DATABASE.'
+./stack.sh:read_password:626 local_xtrace
+./stack.sh:read_password:626 set +o
+./stack.sh:read_password:627 grep xtrace
+./stack.sh:read_password:627 xtrace=set -o xtrace
+./stack.sh:read_password:628 set +o xtrace
+./stack.sh:read_password:628

#####
# ENTER A PASSWORD TO USE FOR THE RABBIT #
#####
admini@admini-HP-280-G2-MT:~/devstack
This value will be written to /home/admini/devstack/.localrc.password file so you don't have to enter it again. Use only alphanumeric characters.
If you leave this blank, a random default value will be used.
tsec
tsec

+functions-common:is_service_enabled:rabbit:1883 local_xtrace
+functions-common:is_service_enabled:rabbit:1884 set +o
+functions-common:is_service_enabled:rabbit:1884 grep xtrace
+functions-common:is_service_enabled:rabbit:1884 xtrace=set -o xtrace
+functions-common:is_service_enabled:rabbit:1884 set +o xtrace
+functions-common:is_service_enabled:rabbit:1913 return 0
+./stack.sh:read_password:626 read_password RABBIT_PASSWORD 'ENTER A PASSWORD TO USE FOR RABBIT.'
+./stack.sh:read_password:626 local_xtrace
+./stack.sh:read_password:626 set +o
+./stack.sh:read_password:627 grep xtrace
+./stack.sh:read_password:627 xtrace=set -o xtrace
+./stack.sh:read_password:628 set +o xtrace
+./stack.sh:read_password:628

#####
# ENTER A PASSWORD TO USE FOR RABBIT #
#####
admini@admini-HP-280-G2-MT:~/devstack
This value will be written to /home/admini/devstack/.localrc.password file so you don't have to enter it again. Use only alphanumeric characters.
If you leave this blank, a random default value will be used.
tsec
tsec

+functions-common:is_service_enabled:keystone:1883 local_xtrace
+functions-common:is_service_enabled:keystone:1884 set +o
+functions-common:is_service_enabled:keystone:1884 grep xtrace
+functions-common:is_service_enabled:keystone:1884 xtrace=set -o xtrace
+functions-common:is_service_enabled:keystone:1884 set +o xtrace
+functions-common:is_service_enabled:keystone:1913 return 0
+./stack.sh:main:704 read_password SERVICE_PASSWORD 'ENTER A SERVICE_PASSWORD TO USE FOR THE SERVICE AUTHENTICATION.'
+./stack.sh:main:704 local_xtrace
+./stack.sh:main:704 set +o
+./stack.sh:main:705 grep xtrace
+./stack.sh:main:705 xtrace=set -o xtrace
+./stack.sh:main:706 set +o
+./stack.sh:read_password:626
+./stack.sh:read_password:626
+./stack.sh:read_password:627
+./stack.sh:read_password:627
+./stack.sh:read_password:627
+./stack.sh:read_password:627
+./stack.sh:read_password:628
+./stack.sh:read_password:628

#####
# ENTER A SERVICE_PASSWORD TO USE FOR THE SERVICE AUTHENTICATION. #
#####
admini@admini-HP-280-G2-MT:~/devstack
This value will be written to /home/admini/devstack/.localrc.password file so you don't have to enter it again. Use only alphanumeric characters.
If you leave this blank, a random default value will be used.
tsec
tsec

```

```

admin@admin-HP-280-G2-MT: ~/devstack
#####
# This value will be written to /home/admin/devstack/.localrc.password file so you don't have to enter it
# again. Use only alphanumeric characters.
# If you leave this blank, a random default value will be used.
Enter a password now:
tsec
+ ./stack.sh:main:704      is_service_enabled keystone
+ functions-common:is_service_enabled:1883  local xtrace
++ functions-common:is_service_enabled:1884  set +o
++ functions-common:is_service_enabled:1884  grep xtrace
+ functions-common:is_service_enabled:1885  xtrace='set -o xtrace'
+ functions-common:is_service_enabled:1913  set +o xtrace
+ ./stack.sh:main:706      read_password SERVICE_PASSWORD 'ENTER A SERVICE_PASSWORD TO USE FOR THE SERVICE AUTHENTICATION.'
+ ./stack.sh:read_password:626  local xtrace
++ ./stack.sh:read_password:627  set +o
++ ./stack.sh:read_password:627  grep xtrace
+ ./stack.sh:read_password:627  xtrace='set -o xtrace'
+ ./stack.sh:read_password:628  set +o xtrace
#####
# ENTER A SERVICE_PASSWORD TO USE FOR THE SERVICE AUTHENTICATION.
#####
# This value will be written to /home/admin/devstack/.localrc.password file so you don't have to enter it
# again. Use only alphanumeric characters.
# If you leave this blank, a random default value will be used.
Enter a password now:
tsec
+ ./stack.sh:main:708      read_password ADMIN_PASSWORD 'ENTER A PASSWORD TO USE FOR HORIZON AND KEYSTONE (20 CHARS OR LESS).'
+ ./stack.sh:read_password:626  local xtrace
++ ./stack.sh:read_password:627  set +o
++ ./stack.sh:read_password:627  grep xtrace
+ ./stack.sh:read_password:627  xtrace='set -o xtrace'
+ ./stack.sh:read_password:628  set +o xtrace
#####
# ENTER A PASSWORD TO USE FOR HORIZON AND KEYSTONE (20 CHARS OR LESS).
#####
# This value will be written to /home/admin/devstack/.localrc.password file so you don't have to enter it
# again. Use only alphanumeric characters.
# If you leave this blank, a random default value will be used.
Enter a password now:
tsec

```

The installer may take up to 30 minutes to complete the installation depends on the internet bandwidth. Once installation is done you may see the following screen which displays ip address of dashboard i.e. horizon through which you can gain access to open stack VMs and resources

```

nfig
2016-10-28 04:32:37.963 | +functions-common:run_plugins:1863
2016-10-28 04:32:37.971 | +functions-common:run_plugins:1864
local plugins=
local plugin

=====
DevStack Component Timing
=====

Total runtime      4466

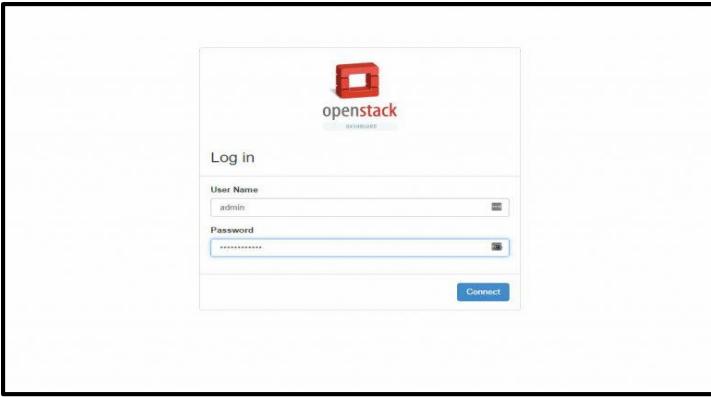
run_process        58
test_with_retry    2
apt-get-update     3
pip_install        799
restart_apache_server 11
wait_for_service   13
git_timed          2037
apt-get            775
=====

This is your host IP address: 192.168.0.116
This is your host IPv6 address: ::1
Horizon is now available at http://192.168.0.116/dashboard
Keystone is serving at http://192.168.0.116/identity/
The default users are: admin and demo
The password: yourpassword
stack@cloudstack1:/devstacks

```

As you can see, two users have been created for you; admin and demo. Your password is the password you set earlier. These are the usernames you will use to login to the OpenStack Horizon Dashboard.

Open up a browser, and put the **Horizon Dashboard address** in your address bar. <http://192.168.0.116/dashboard> you should see a **login page** like this.



To start with, log in with the admin users credentials. In admin panel, you will need to use the demo user, or create a new user, to create and deploy instances. As you can see, two users have been created for you; admin and demo. Your password is the password you set earlier. These are the usernames you will use to login to the OpenStack Horizon Dashboard. Take note of the Horizon web address listed in your terminal.

A screenshot of the OpenStack Horizon Dashboard's Identity / Projects page. The top navigation bar shows "Project", "Admin", and "Identity". Under "Identity", "Projects" is selected. The main area displays a table of projects. The columns are: Name, Description, Project ID, Domain Name, Enabled, and Actions. The table contains five entries:

Name	Description	Project ID	Domain Name	Enabled	Actions
service		658a0f583a7b4a978734531968c859ec	Default	Yes	Manage Members
alt_demo		b9fb25d580c844fabbb3001d61182385e	Default	Yes	Manage Members
demo		c0f5b140bfee447ea2930a78d8996a72	Default	Yes	Manage Members
invisible_to_admin		c9609da33fdc45fe9d30b195a143a3e8	Default	Yes	Manage Members
admin	Bootstrap project for initializing the cloud.	fa715732d0cf441381194d7521f1eec2	Default	Yes	Manage Members

Displaying 5 items

Creating and running Instances

To launch an instance from OpenStack dashboard, first we need to finish following steps:

- Create a Project and add a member to the Project.
- Create Image and Flavor
- Create Network for the Project.
- Create Router for the Project.

- Create a Key pair

A) Create a Project and add a member to the Project.

Login to the dashboard using Admin credentials and Go to **Identity Tab -> Projects and Click on Create Project.**

The screenshot shows the 'Projects' section of the OpenStack Dashboard. A modal window titled 'Create Project' is open. In the 'Project Information' tab, the 'Name' field contains 'Innovation' and the 'Description' field contains 'Innovation'. The 'Enabled' checkbox is checked. On the right side of the modal, there are tabs for 'Project Members' and 'Quota *'. Below the tabs is a table with two rows, each labeled 'Yes' under 'Enabled' and 'Actions' with a 'Manage Members' dropdown. At the bottom of the modal are 'Cancel' and 'Create Project' buttons, with 'Create Project' being the active one.

Click on “Create Project” , We can also set the Quota for the project from Quota Tab. **To create Users , Go to Identity Tab-> Users-> Click on ‘Create User’ Button then specify User Name, email, password, Primary Project and Role and click on create user to add in to OpenStack workspace.**

The screenshot shows the 'Users' section of the OpenStack Dashboard. A modal window titled 'Create User' is open. The form fields are: User Name: 'linuxtech1', Email: 'linuxtech1@example.com', Password: '*****', Confirm Password: '*****', Primary Project: 'Innovation', and Role: '_member_'. There is a checked checkbox for 'Enabled'. To the right of the modal, there is a table with multiple rows, each with an 'Edit' button. At the bottom of the modal are 'Cancel' and 'Create User' buttons, with 'Create User' being the active one.

B) Create Image and Flavor

To create a flavor login in dashboard using admin credentials, Go to **Admin Tab -> Flavors -> Click on create Flavor.**

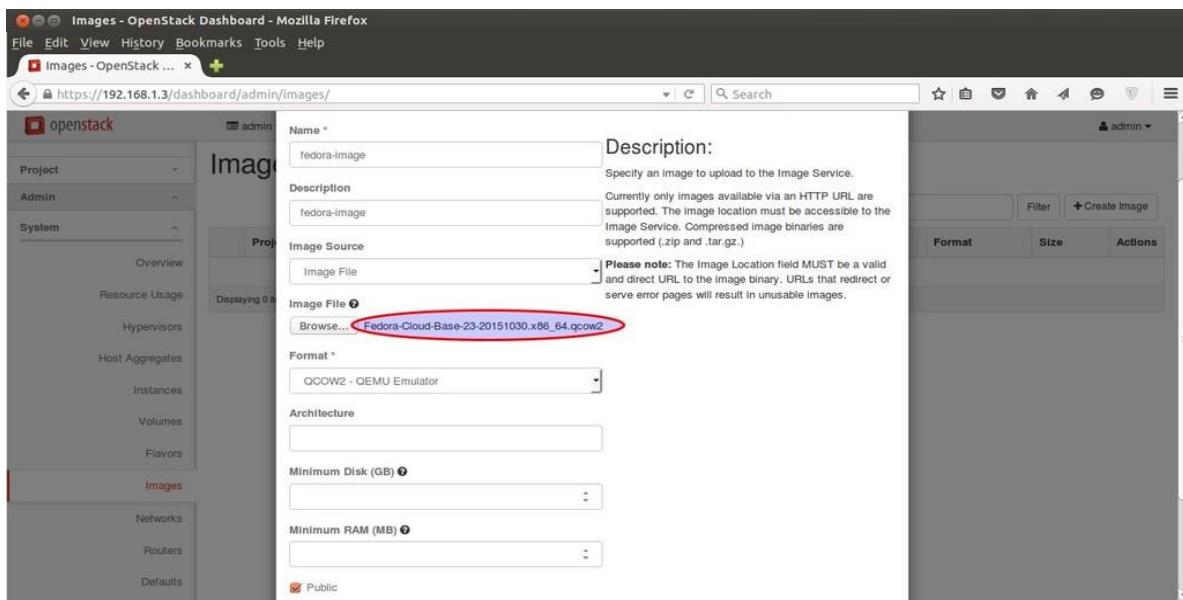
Flavors										
	Flavor Name	VCPUs	RAM	Root Disk	Ephemeral Disk	Swap Disk	ID	Public	Metadata	Actions
	m1.tiny	1	512MB	1GB	0GB	0MB	1	Yes	No	<button>Edit Flavor</button>
	m1.small	1	2GB	20GB	0GB	0MB	2	Yes	No	<button>Edit Flavor</button>
	m1.medium	2	4GB	40GB	0GB	0MB	3	Yes	No	<button>Edit Flavor</button>
	m1.large	4	8GB	80GB	0GB	0MB	4	Yes	No	<button>Edit Flavor</button>
	m1.xlarge	8	16GB	160GB	0GB	0MB	5	Yes	No	<button>Edit Flavor</button>

Specify the Flavor Name (fedora.small) , VCPU , Root Disk , Ephemeral Disk & Swap disk.

Name *	fedora.small	Flavor Access
ID *	auto	Flavors define the sizes for RAM, disk, number of cores, and other resources and can be selected when users deploy instances.
VCPUs *	1	Public
RAM (MB) *	1024	Metadata
Root Disk (GB) *	10	Actions
Ephemeral Disk (GB)	2	<button>Edit Flavor</button>
Swap Disk (MB)	512	<button>Create Flavor</button>

To Create Image , Go to Admin Tab → Images—> Click on Create Image.

Specify the Image Name , Description, Image Soure (in my case i am using Fedora Image File which i have already downloaded from fedora website with Format QCOW2)



C) Create Network for the Project.

To create Network and router for Innovation project sign out of admin user and login as local user in dashboard.

For my convenience i have setup my network as above

Internal Network = 10.10.10.0/24

External Network or Floating IP Network = 192.168.1.0/24

Gateway of External Network = 192.168.1.1

Now, Go to the Network Tab —> Click on Networks —> then Click on Create Network

Specify the Network Name as Internal

Network Name	internal	Create a new network. In addition, a subnet associated with the network can be created in the next panel.
Admin State	UP	
<input checked="" type="checkbox"/> Create Subnet		
<input type="button" value="Cancel"/> <input type="button" value="« Back"/> <input type="button" value="Next »"/>		

Click on Next. Then Specify the Subnet name (sub-internal) and Network Address (10.10.0.0/24)

Create Network

The screenshot shows the first step of creating a subnet. The title is "Create Network". Below it is a breadcrumb navigation: "Network > Subnet > Subnet Details". The main form fields include:

- Subnet Name:** sub-internal
- Network Address:** 10.10.0.0/24
- IP Version:** IPv4
- Gateway IP:** (empty field)
- Disable Gateway:** (unchecked checkbox)

A note on the right says: "Create a subnet associated with the network. Advanced configuration is available by clicking on the "Subnet Details" tab." At the bottom are "Cancel", "« Back", and a blue "Next »" button.

Click on Next. Now, VMs will be getting internal IP from DHCP Server because we enable DHCP option for internal network.

The screenshot shows the second step of creating a subnet, specifically the "Subnet Details" section. The breadcrumb navigation is "Network > Subnet > Subnet Details". The main form fields include:

- Enable DHCP:** (checked checkbox)
- Allocation Pools:** (empty text area)
- DNS Name Servers:** (empty text area)
- Host Routes:** (empty text area)

A note on the right says: "Specify additional attributes for the subnet." At the bottom are "Cancel", "« Back", and a blue "Create" button.

Now **Create External Network**. Click on “Create Network” again, Specify Network Name as “external”

Create Network

The screenshot shows the first step of creating a new network. The title is "Create Network". Below it is a breadcrumb navigation: "Network > Subnet > Subnet Details". The main form fields include:

- Network Name:** external
- Admin State:** UP
- Create Subnet:** (checked checkbox)

A note on the right says: "Create a new network. In addition, a subnet associated with the network can be created in the next panel." At the bottom are "Cancel", "« Back", and a blue "Next »" button.

Click on Next. Specify subnet Name as “**sub-external**” & Network Address as “**192.168.1.0/24**”

Create Network

Network > Subnet > Subnet Details

Subnet Name: sub-external

Network Address: 192.168.1.0/24

IP Version: IPv4

Gateway IP: 192.168.1.1

Disable Gateway

Create a subnet associated with the network. Advanced configuration is available by clicking on the "Subnet Details" tab.

Cancel | « Back | Next »

Click on Next

Untick “**Enable DHCP**” option and Specify the ip address pool for external network.

Network > Subnet > Subnet Details

Specify additional attributes for the subnet.

Enable DHCP

Allocation Pools: 192.168.1.19, 192.168.1.30

DNS Name Servers

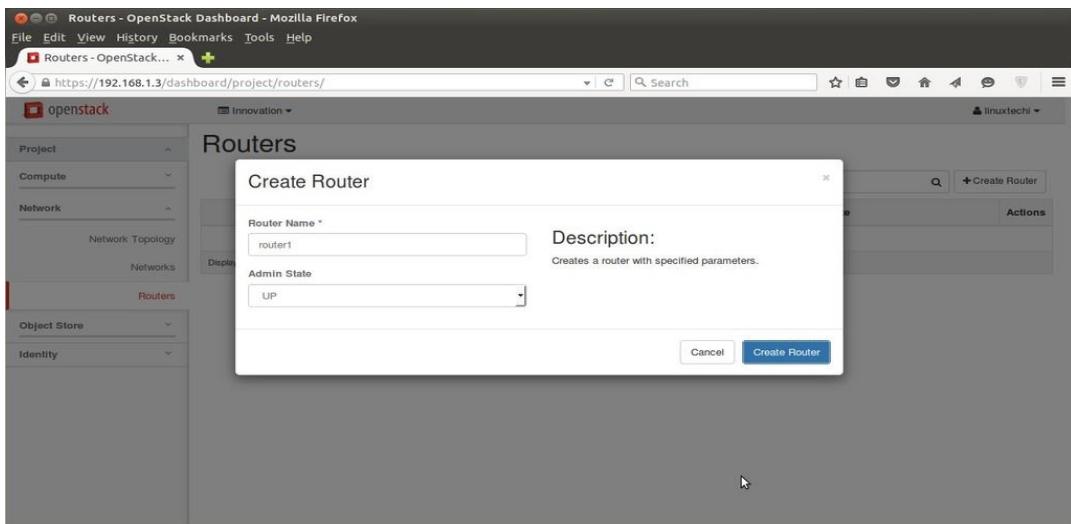
Host Routes

Cancel | « Back | Create

Click on Create.

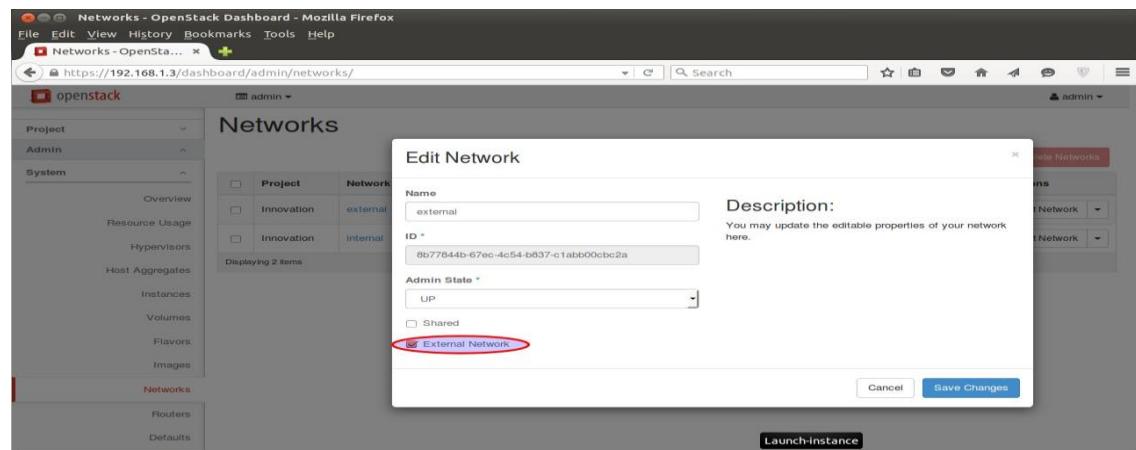
D) Create Router for the Project

Now time to create a Router. To create router Go To Network Tab → Routers → Click on ‘+ Create Router’



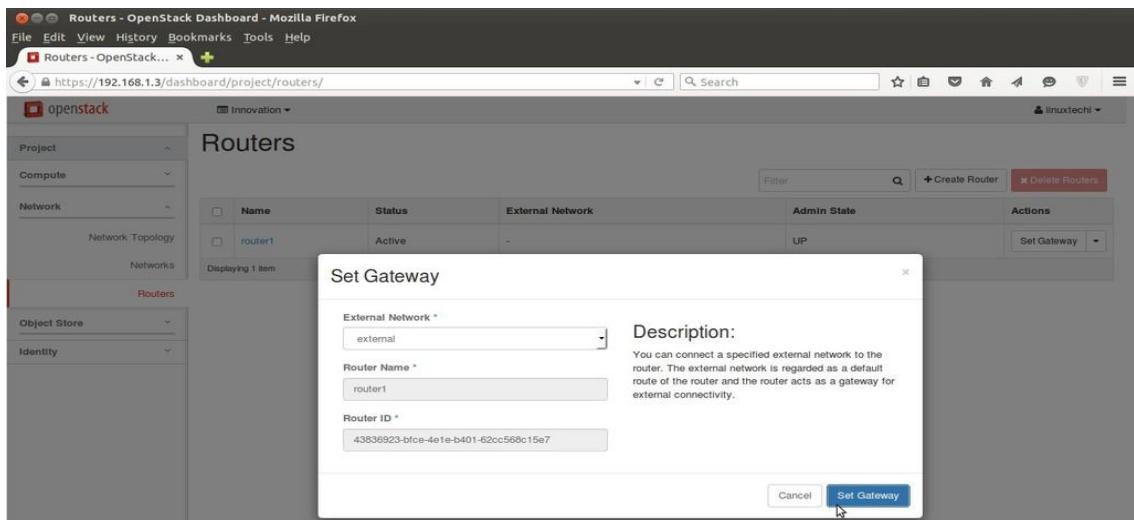
Now Mark External network as “External” , this task can be completed only from admin user , so logout from linuxtech user and login as admin.

Go to **Admin Tab** —> **Networks**—> Click on **Edit Network** for “External”

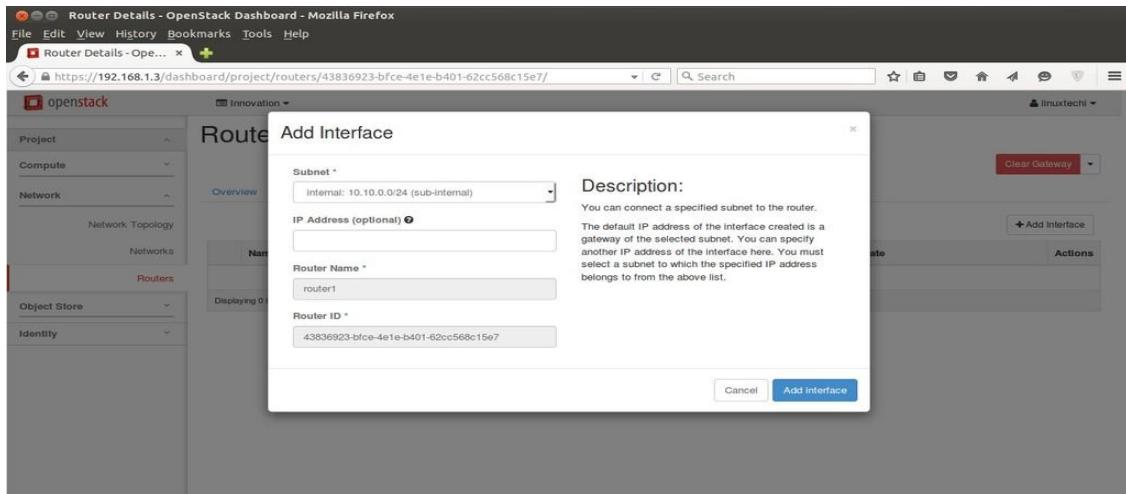


Click on Save Changes. Now Logout from admin user and login as local user. Go to **Network Tab** —> **Routers** —> for Router1 click on “Set Gateway”



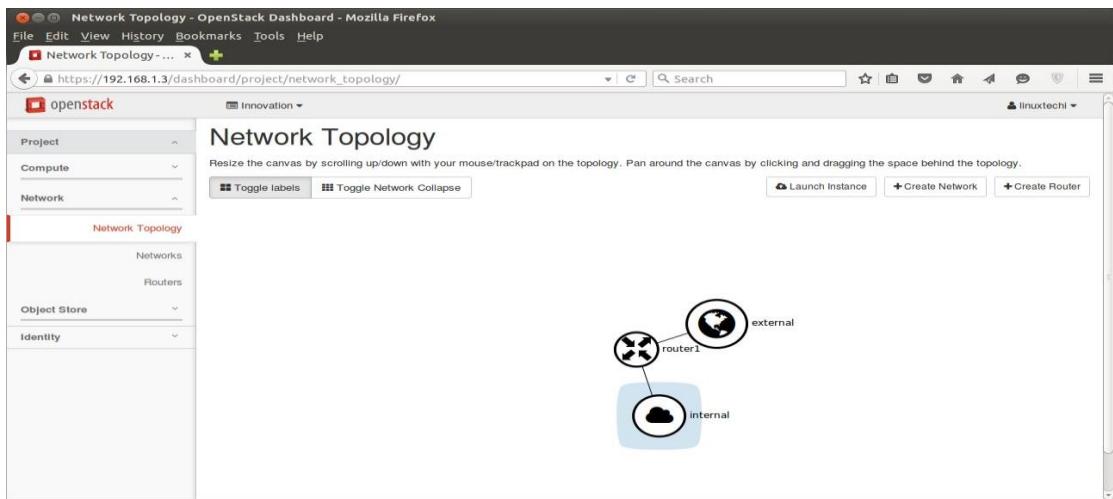


Click on “**Set Gateway**”, this will add a interface on router and will assign the first ip of external subnet (192.168.1.0/24).



Add internal interface to router as well , Click on the “**router1**” and select on “**interfaces**” and then click on “**Add interface**”

Now, Network Part is completed now & we can view Network Topology from “**Network Topology**” Tab as below.



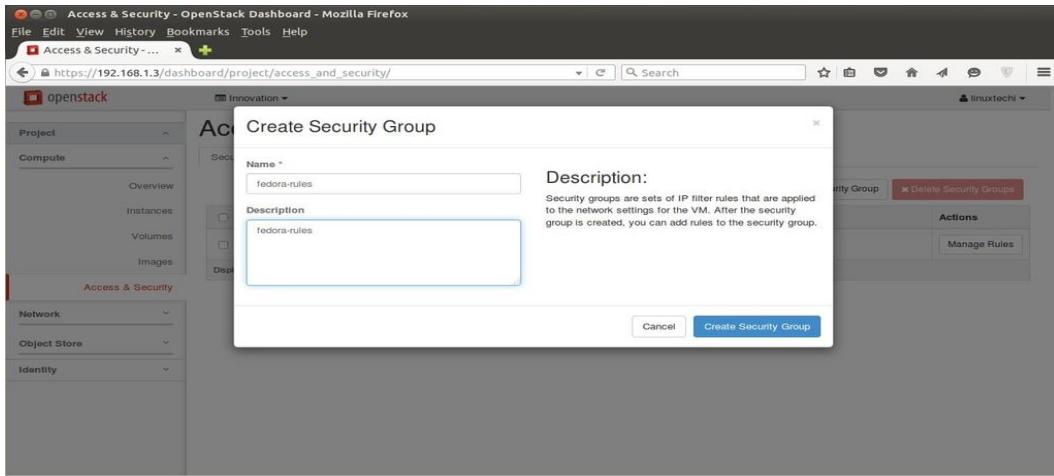
Now Create a key pair that will be used for accessing the VM and define the Security firewall rules.

E) Create a key pair

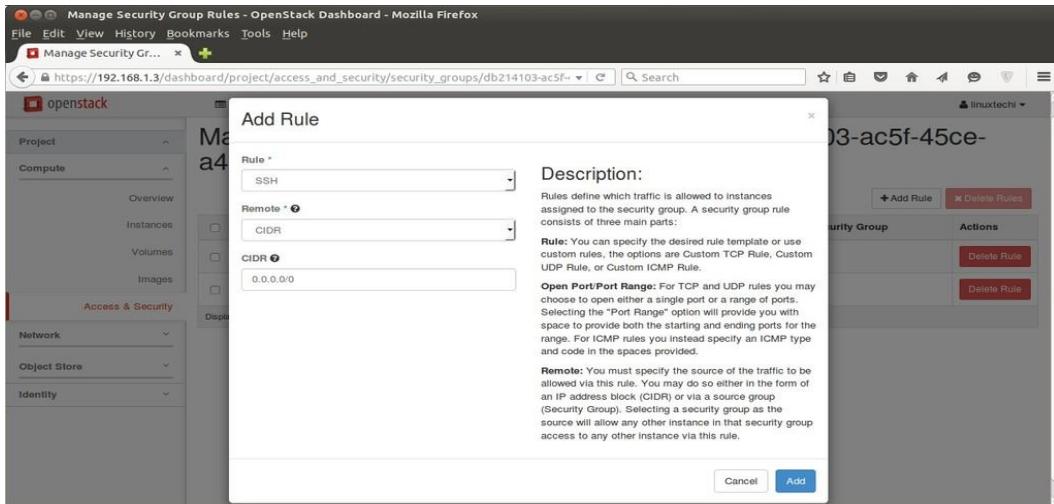
Go to ‘Access & Security’ Tab → Click on Key Pairs → then click on ‘Create Key Pair’

The screenshot shows the Access & Security section of the OpenStack Dashboard. On the left, there's a sidebar with 'Compute' dropdowns for Overview, Instances, Volumes, and Images, and a 'Network' section under 'Access & Security'. The main area has a 'Key Pairs' tab selected. A modal dialog box is open, titled 'Create Key Pair'. It has a 'Key Pair Name *' input field containing 'myssh-keys'. Below it is a 'Description:' section with a detailed description of what key pairs are and how they work. At the bottom of the dialog are 'Cancel' and 'Create Key Pair' buttons.

It will create a Key pair with name “**myssh-keys.pem**” Add a new Security Group with name ‘**fedora-rules**’ from Access & Security Tab. Allow 22 and ICMP from Internet (0.0.0.0).



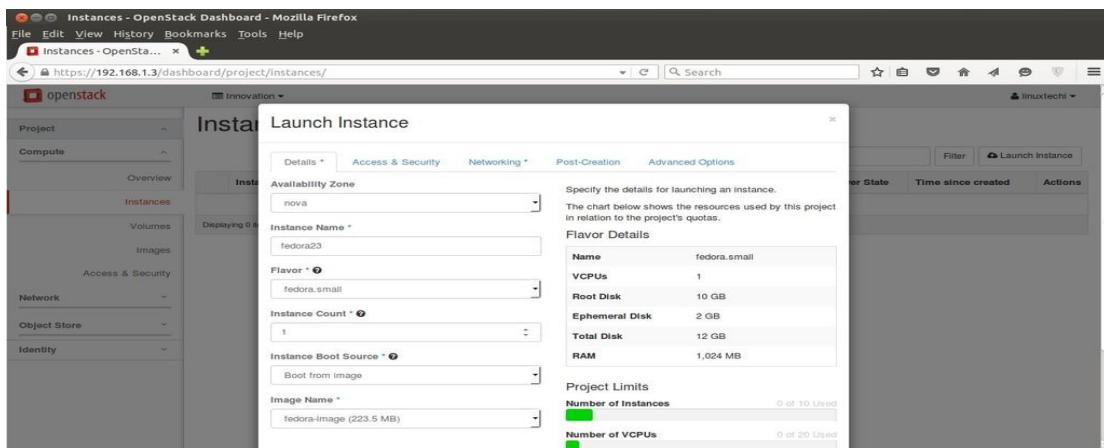
Once the Security Group ‘fedora-rules’ created , click on Manage Rules and allow 22 & ICMP ping.



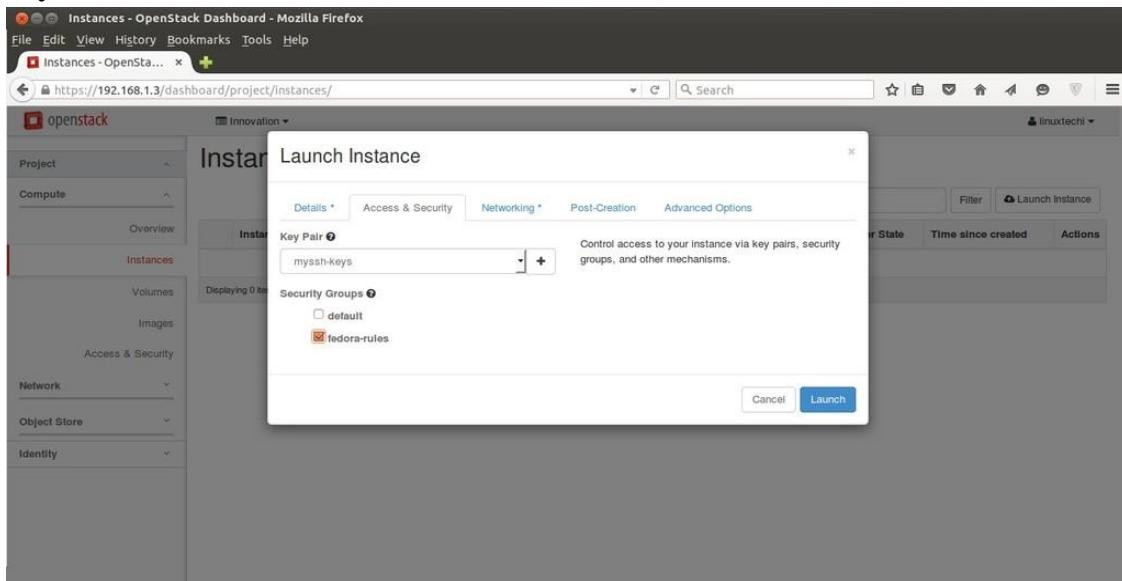
Click on Add , Similarly add a rule for ICMP.

F) Launch Instance

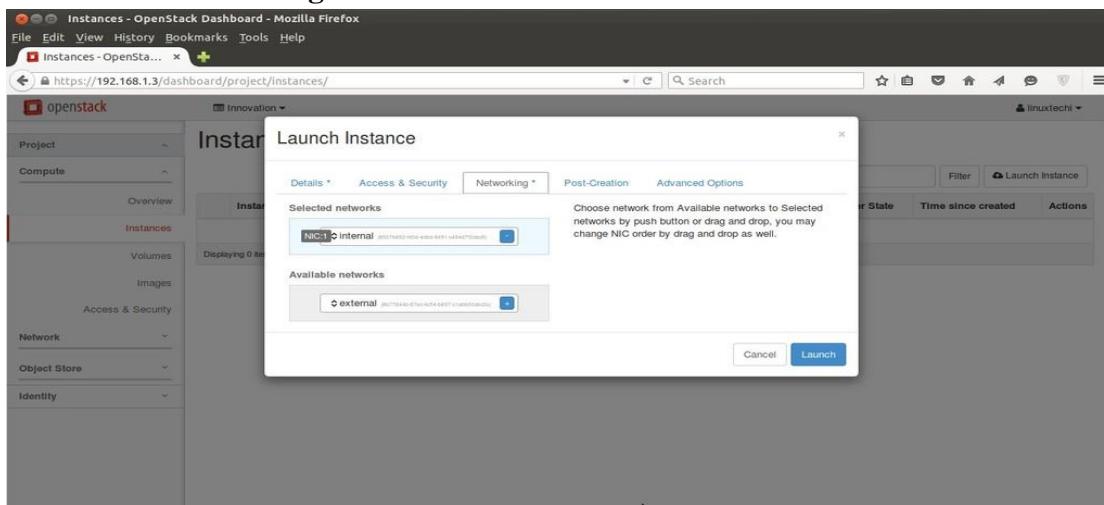
Now finally it's time to launch an instance. To launch instance, Go to **Compute Tab** → **Click on Instances** → **then click on ‘Launch Instance’** Then Specify the Instance Name, Flavor that we created in above steps and **‘Boot from image’** from Instance Boot Source option and Select Image Name **‘fedora-image’**.



Click on ‘Access & Security’ and Select the Security Group ‘**fedora-rules**’ & Key Pair ”**myssh-keys**“



Now Select **Networking** and add ‘Internal’ Network and the Click on Launch



Once the VM is launched , Associate a floating ip so that we can access the VM.

The screenshot shows the OpenStack Dashboard's Instances page. A single instance, 'fedora23', is listed. A context menu is open over this instance, and the 'Associate Floating IP' option is highlighted with a red box. Other options in the menu include Create Snapshot, Attach Interface, Detach Interface, Edit Instance, Edit Security Groups, Console, View Log, Pause Instance, Suspend Instance, Shelve Instance, Resize Instance, Lock Instance, and Unlock Instance.

Click on ‘Associate Floating IP‘ to get public IP addresses

The screenshot shows the 'Allocate Floating IP' dialog box. It has a dropdown for 'Pool' set to 'external'. Below it is a 'Description:' field containing the text 'Allocate a floating IP from a given floating IP pool.' There is also a 'Project Quotas' section showing 'Floating IP (0)' and '50 Available'. At the bottom of the dialog are 'Cancel' and 'Allocate IP' buttons, with the 'Allocate IP' button highlighted by a red box.

Click on Allocate IP.

The screenshot shows the 'Manage Floating IP Associations' dialog box. It has fields for 'IP Address' (set to '192.168.1.20') and 'Port to be associated' (set to 'fedora23: 10.10.0.3'). A message in the center says 'Select the IP address you wish to associate with the selected instance or port.' At the bottom are 'Cancel' and 'Associate' buttons, with the 'Associate' button highlighted by a red box. A success message 'Success: Allocated Floating IP 192.168.1.20.' is displayed in a green box. A footer note 'Current workspace: "Workspace 1"' is visible at the bottom right.

Click on Associate

The screenshot shows the 'Instances' page of the OpenStack Dashboard. On the left, there's a sidebar with 'Project' dropdowns for 'Compute', 'Network', 'Object Store', and 'Identity'. Under 'Compute', 'Instances' is selected. The main area is titled 'Instances' and contains a table with one row. The table columns are: Instance Name, Image Name, IP Address, Size, Key Pair, Status, Availability Zone, Task, Power State, Time since created, and Actions. The single entry is: fedora23, fedora-image, 10.10.0.3 (Floating IPs: 192.168.1.20), fedora.small, myssh-keys, Active, nova, None, Running, 12 minutes, and a 'Create Snapshot' button.

Now try to access the VM with floating IP (192.168.1.20) using keys.

```
[root@openstack ~]# ls -l myssh-keys.pem
-rw-r--r--. 1 root root 1683 Jan  2 12:30 myssh-keys.pem
[root@openstack ~]# chmod 600 myssh-keys.pem

[root@openstack ~]# ssh -i myssh-keys.pem 192.168.1.20 -l fedora
The authenticity of host '192.168.1.20 (192.168.1.20)' can't be established.
RSA key fingerprint is 99:d0:da:3d:d2:41:89:fc:df:97:d3:25:b9:29:9b:22.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.20' (RSA) to the list of known hosts.
[fedora@fedora23 ~]$ sudo su -
[root@fedora23 ~]# cat /etc/redhat-release
Fedora release 23 (Twenty Three)
[root@fedora23 ~]#
```

As we can see above that we are able to access the VM using keys. Our task of launching a VM from Dashboard is Completed Now.

LAB 5	Explore Storage as a service using own Cloud for remote file access using web interfaces.
--------------	--

ownCloud is a suite of client–server software for creating and using file hosting services. ownCloud is functionally very similar to the widely used Dropbox, with the primary functional difference being that the Server Edition of ownCloud is free and open-source, and thereby allowing anyone to install and operate it without charge on a private server. It also supports extensions that allow it to work like Google Drive, with online document editing, calendar and contact synchronization, and more. Its openness avoids enforced quotas on storage space or the number of connected clients, instead having hard limits (like on storage space or number of users) defined only by the physical capabilities of the server.

Installation and configuration of OwnCloud

Own cloud can be installed over the any flavor of linux like Ubuntu, Centos, Fedora etc. but Ubuntu is preferable. The Steps for installation are as follows

Step 1 – Installing ownCloud

The ownCloud server package does not exist within the default repositories for Ubuntu. However, ownCloud maintains a dedicated repository for the distribution that we can add to our server.

To begin, download their release key using the curl command and import it with the apt-key utility with the add command:

```
$curl https://download.owncloud.org/download/repositories/10.0/Ubuntu_18.04/Release.key |  
sudo apt-key add -
```

The 'Release.key' file contains a PGP (Pretty Good Privacy) public key which apt will use to verify that the ownCloud package is authentic.

Now execute following commands on the terminal

```
1) $ echo 'deb http://download.owncloud.org/download/repositories/10.0/Ubuntu_18.04/' | sudo  
tee /etc/apt/sources.list.d/owncloud.list  
2) $sudo apt update  
3)$ sudo apt install php-bz2 php-curl php-gd php-imagick php-intl php-mbstring php-xml php-  
zip owncloud-files
```

Step 2 :- Set the Document Root

The ownCloud package we installed copies the web files to /var/www/owncloud on the server. Currently, the Apache virtual host configuration is set up to serve files out of a different

directory. We need to change the DocumentRoot setting in our configuration to point to the new directory.

```
$sudo apache2ctl -t -D DUMP_VHOSTS | grep server_domain_or_IP
```

Now edit the Configuration file and add following lines so that it points to the /var/www/owncloud directory:

\$sudo nano /etc/apache2/sites-enabled/server_domain_or_IP.conf	<VirtualHost *:80> ... DocumentRoot /var/www/owncloud ... </VirtualHost>
---	--

When you are finished, check the syntax of your Apache files to make sure there were no detectable typos in your configuration

```
$sudo apache2ctl configtest
```

Output:- Syntax OK

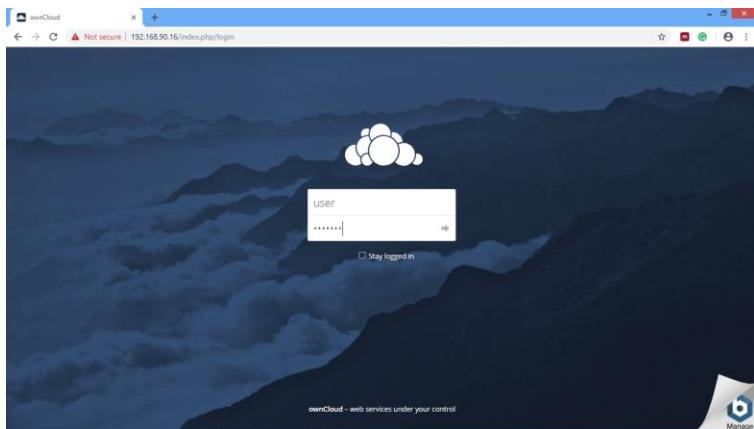
Step 3 – Configuring the MySQL Database

Open mysql prompt, create database and execute following commands

- 1) \$mysql -u root -p
- 2) mysql>CREATE DATABASE owncloud;
- 3) mysql>GRANT ALL ON owncloud.* to 'owncloud'@'localhost' IDENTIFIED BY 'owncloud_database_password';
- 4) mysql>FLUSH PRIVILEGES;

Step 4:- Configure ownCloud

To access the ownCloud web interface, open a web browser and navigate to the servers IP address as shown below

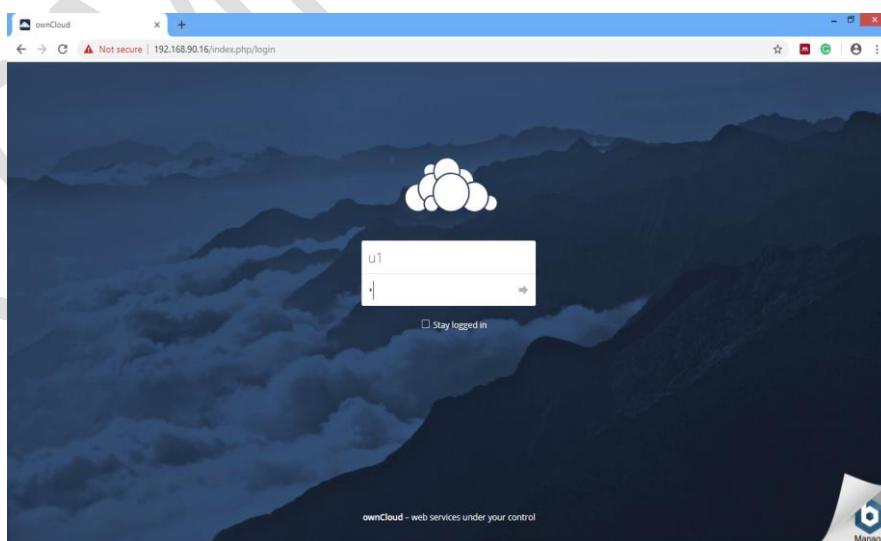


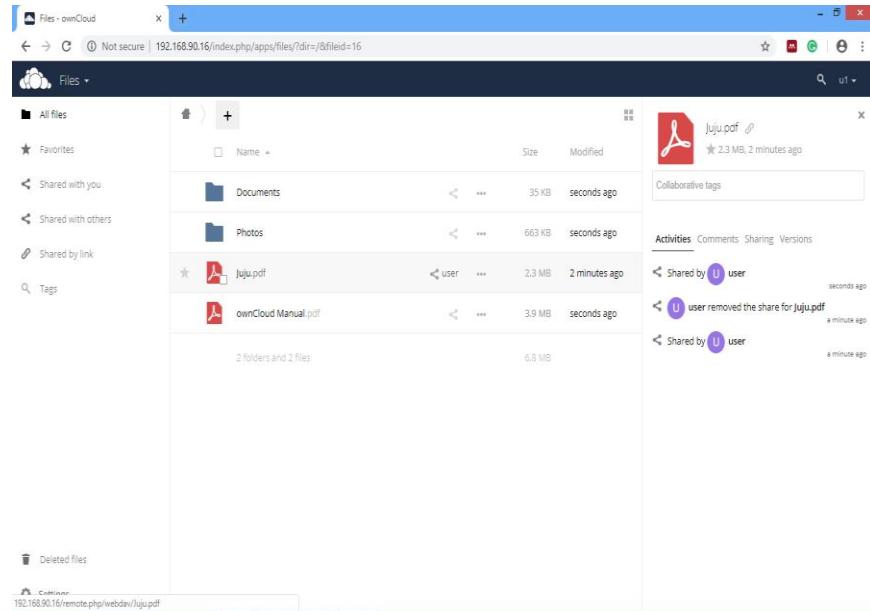
Own Cloud portal has two types of users like Admin user and local user. The admin user can create users/groups, assigns storage quota, assigns privileges and can manage users and group activities.

The image displays four separate browser windows side-by-side, each showing a different aspect of the openCloud web interface:

- File - openCloud**: Shows the main file list with files like "ownCloud Manual.pdf" and "Juju.pdf". A context menu is open over "Juju.pdf", showing options like "Upload", "Folder", and "Text file".
- File - openCloud**: Shows the same file list, but with a different view where "Juju.pdf" is highlighted.
- File - openCloud**: Shows the file list with a "Sharing" tab selected. It shows a "Share link" button and a URL "http://192.168.90.16/index.php/s/RqpwetVblydpQCK".
- Admin - openCloud**: Shows the server configuration page. Under the "Sharing" section, several checkboxes are checked, including "Allow app to use the Share API", "Allow users to share via link", and "Allow public uploads". Other sections like "Federation", "File handling", and "Email server" are also visible.

The local user is an restricted user who can perform local activities like upload or share files, delete local shares or can create share etc.





The alternate way to use own cloud is to download the readymade virtual machine from website <https://bitnami.com/stack/owncloud/cloud> which can be run directly on virtualization platform like virtual box or VMware workstation.