

OSU ASCCcloud

User Guide (Rev 2.0), April 2016

Oklahoma State University
School of Electrical and Computer Engineering
Laboratory for Advanced Sensing, computation and Control
<http://ascc.okstate.edu/>
ATRC #320
Tel: 405 744 4799

This guide is prepared to be used with ASCC Private cloud and can not be used as a guide for
accessing public cloud services

Contents

1.	ASCC Private Cloud Infrastructure	1
2.	The Dashboard	3
3.	Introduction to key components	4
3.1.	Instances	4
3.2.	Images	4
3.3.	Flavors	5
3.4.	Volumes	5
4.	Creating Instances.....	5
5.	Configuring Security Groups	9
6.	Verifying Connectivity and Remote Access.....	10
7.	Adding Volume Storage	13
8.	Instance Backup	18
9.	Adding Desktop environment to Ubuntu 14.04 cloud server.....	21
9.1.	Xfce4 Desktop	21
9.2.	Gnome Desktop	22
9.3.	Remote Desktop Access.....	23
10.	Deleting Instances.....	26

1. ASCC Private Cloud Infrastructure

The ASCC cloud is a private cloud infrastructure developed in the Advanced Sensing, Computation and Control lab for cloud based academic and research projects. The cloud provides computing, storage and networking services using the Infrastructure-as-a-Service (IaaS) cloud resource provisioning model. Following are the essential characteristics of the cloud platform;

- *On-demand self-service*: users can provision computing capabilities, such as server time and storage, as needed automatically without requiring administrative privileges.
- *Broad network access*. Resources are available throughout OSU A&M network and can be accessed from heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).
- *Resource pooling*: Computing resources are pooled to serve multiple projects using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to the demand.
- *Rapid elasticity*. Users can rapidly scale computing and storage resources depending on the demand.

The infrastructure is set-up using three Dell Poweredge R420 Servers each possessing a 12 core Intel Xeon E5-2430 processor and 32 GB DDR3 memory. The cloud has a storage capacity of 13TB for ephemeral instance storage, permanent backups. Openstack Juno project is used with Ubuntu 14.04 LTS server for setting up the cloud cluster. The implementation layers of the cloud infrastructure are shown in figure 1.

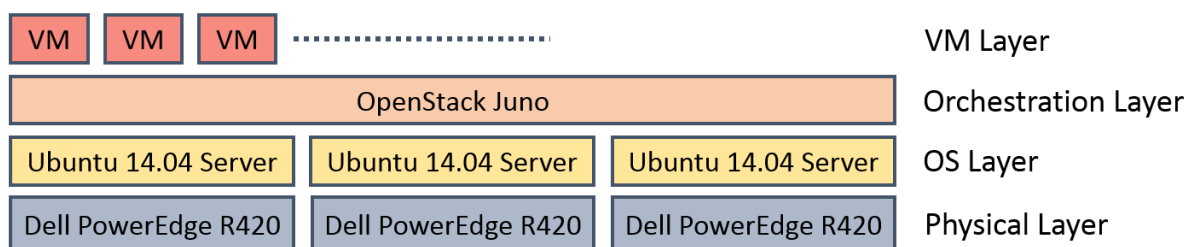


Figure 1. Implementation Layers

These layers are set up using the architecture shown in figure 2. One server is used to set up the controller and storage node while two servers are used as compute nodes. All administrative and management services are implemented on the controller while the compute nodes provide computing and network resources for instances (virtual machines).

The cloud uses a separate network 10.194.240.0/24 which is routable throughout OSU campus.

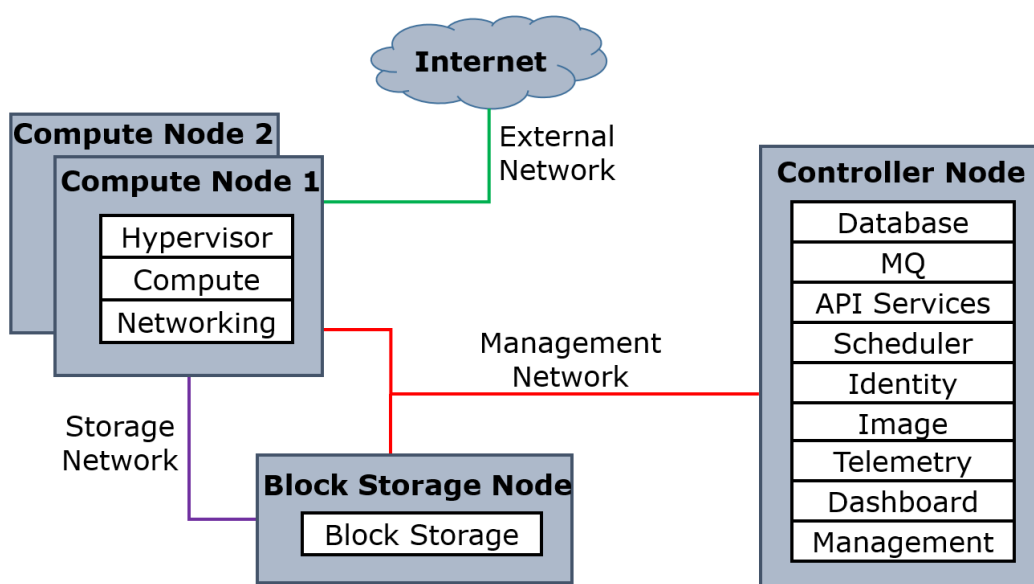


Figure 2. Cloud Architecture

2. The Dashboard

Access to the cloud is provided on a project basis. Admins are responsible for creating user accounts to utilize ASCC cloud resources. The cloud management dashboard can be accessed from the following link;

<http://10.194.240.1/horizon> or alternatively

<http://atrc-cloud1.atrc-ecen.okstate.edu/horizon>

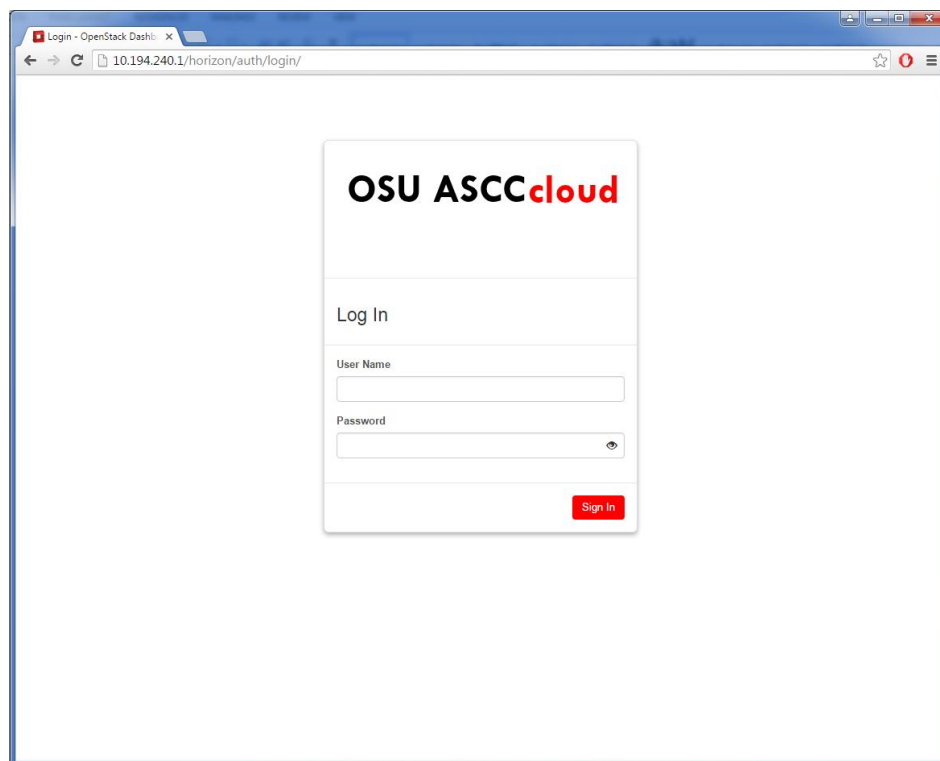


Figure 3. Cloud Management Dashboard

The dashboard provides the tools for creating and managing instances. After logging in using provided user credentials, you will be able to see your allowed usage of instances, VCPUs, Memory, Floating IPs and Security Groups for the project as shown in figure 4. The instance console access on the dashboard is however disabled and instances can only be accessed remotely from a network.

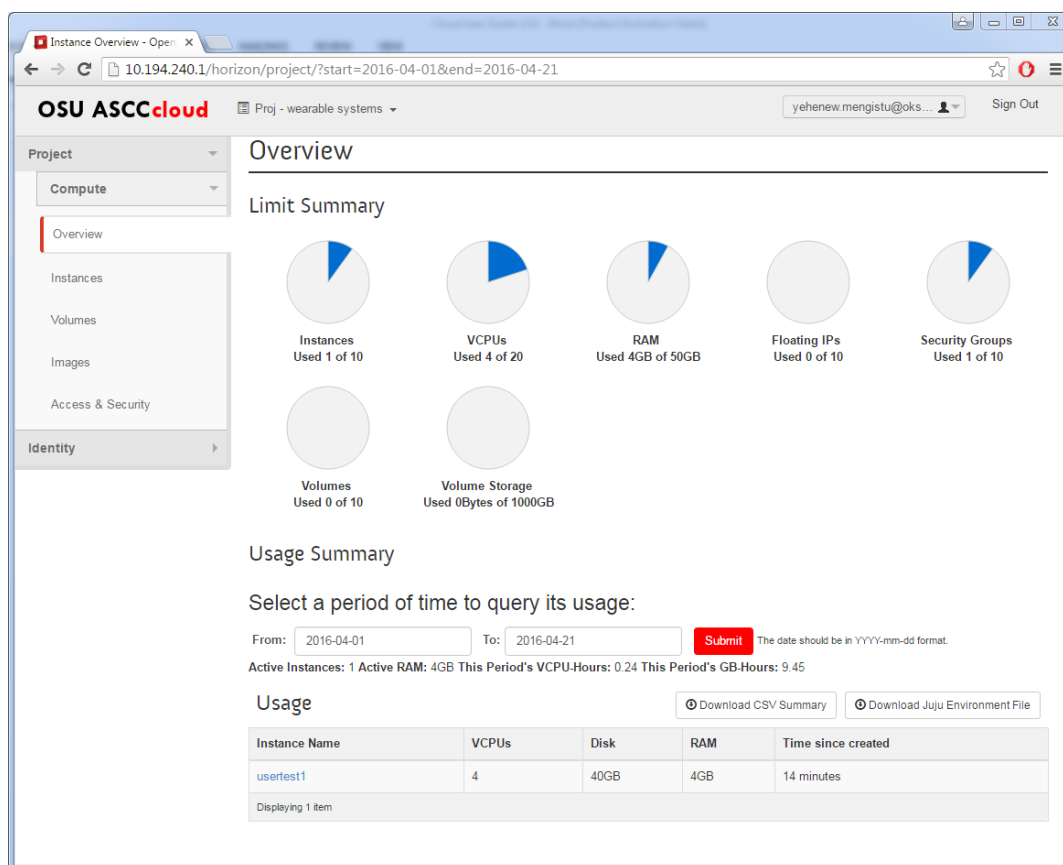


Figure 4. Instance Usage Summary

3. Introduction to key components

3.1. Instances

Instances are virtual servers on the cloud hosted by the QEMU hypervisor on the compute nodes. These servers are different from physical servers in that computing, network and storage resources can be easily allocated on demand without the burden of managing server hardware.

3.2. Images

Two stable Linux server releases are provided on ASCC cloud. Ubuntu Trusty 14.04 LTS is a stable and largely utilized linux server OS based on Debian. The CentOS 7 is another widely used linux server which is considered as an enterprise-grade open source OS based on the open source code of Red Hat Enterprise Linux. The Cirros 0.3.4 image is a small footprint linux OS provided

for testing purposes. Default login user names and passwords for these images are included in the following table.

Image	User name	Password
Cirros 0.3.4	Cirros	Cubswin:)
Ubuntu 14.04*	ubuntu	
CentOS 7*	centos	

Table 1. Default login credentials (*empty default password)

3.3. Flavors

Flavors are virtual hardware templates for the type and amount of resources instances will use once created. The four types of flavors provided on the ASCC cloud have the following resource limits. VCPUs represents a time slot of the processor or approximately a single core.

Flavor Name	VCPUs*	RAM	Root Disk
m1.small	2	2048MB	20GB
m1.medium	4	4096MB	40GB
m1.large	8	8192MB	80GB
m1.xlarge	12	16384MB	100GB

Table 2. Flavors resource limits

3.4. Volumes

Additional volumes can be attached to instances in addition to the root disk which is available upon instance creation. These volumes are persistent unlike the default instance storage which will be erased as the instance is terminated. As a result volumes are ideal for backup or additional storage. While a volume can be attached to one instance at a time, it can however be detached and added to other instances.

4. Creating Instances

Cloud instances or VMs can only be accessed through terminal programs like Putty using SSH protocols. Hence the first step before building instances would be to create a key-pair. You can use Putty Key Generator (Putty Gen) as shown in the following figure to create a key pair. The public key can be generated by creating random movements with the mouse as shown in figure 5. This should be used at instance creation and the corresponding private key should be saved for later use while accessing the instance remotely.

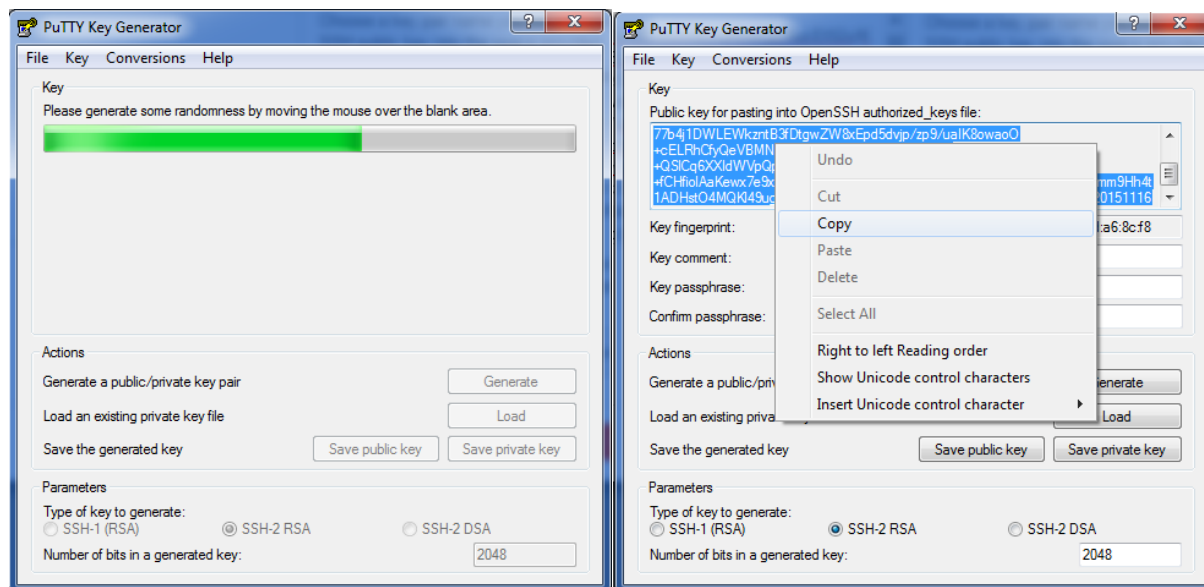


Figure 5. Generating Public and Private keys

Next, go to the Access & Security tab on the left panel of the dashboard and select key-pairs. Go to import key pair and create a new key pair by giving it a unique name preferably related to the project. Copy and paste the public key generated in the previous step as shown in figure 6. Remember to save the private key pair from the Putty Gen window as this is a one-time process and cannot be obtained later.

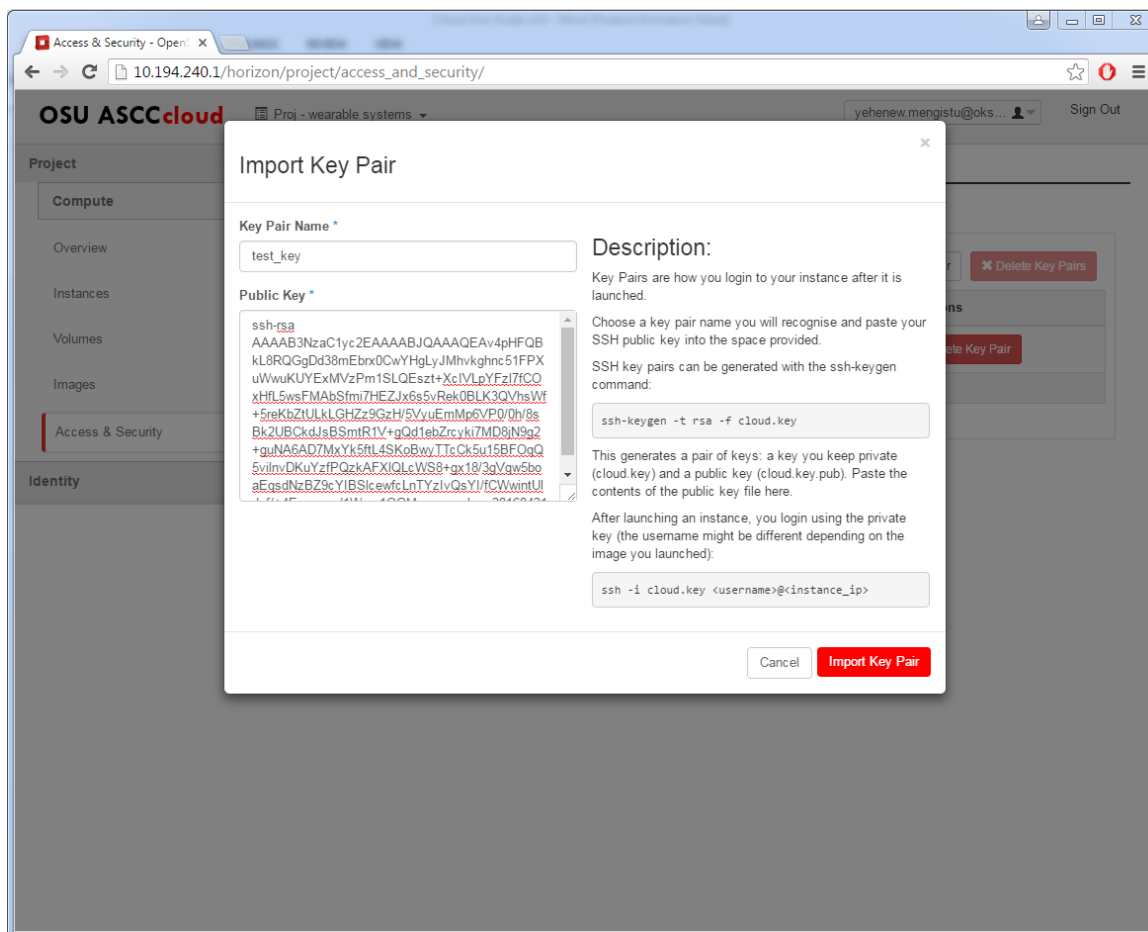


Figure 6. Importing Public Key

After creating the key-pair locate the instances tab and choose launch instance. In the launch instance wizard shown in figure 7, configure the following settings

- i. Instance name - give the instance a unique name related to the project
- ii. Flavor – choose the appropriate flavor based on requirements. The flavor m1.medium is sufficient enough for most deployments using Ubuntu 14.04 server.
- iii. Instance boot source - select the boot from image option. Instances can also boot from a snapshot (backup) image of previous instances.
- iv. Image name - Choose your preferred cloud image and go to the access and security tab.
- v. Access & Security - select the key pair you previously created, choose the default security group and launch the instance.

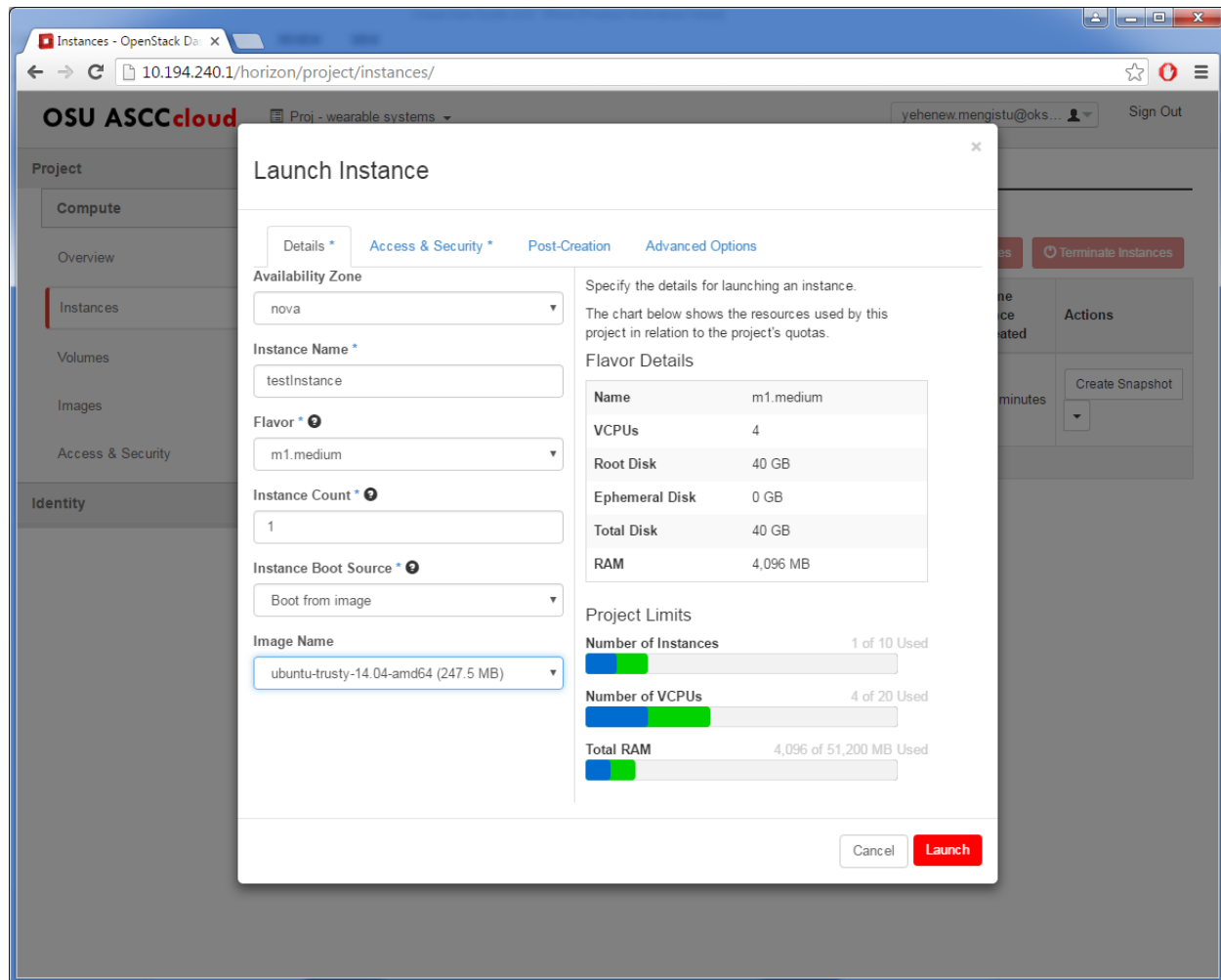


Figure 7. Launch Instance wizard

A success status message will be displayed after creation and the instance goes through different building tasks before it displays a Running status where it should be ready for use. Notice the IP address which is automatically assigned to the instance.

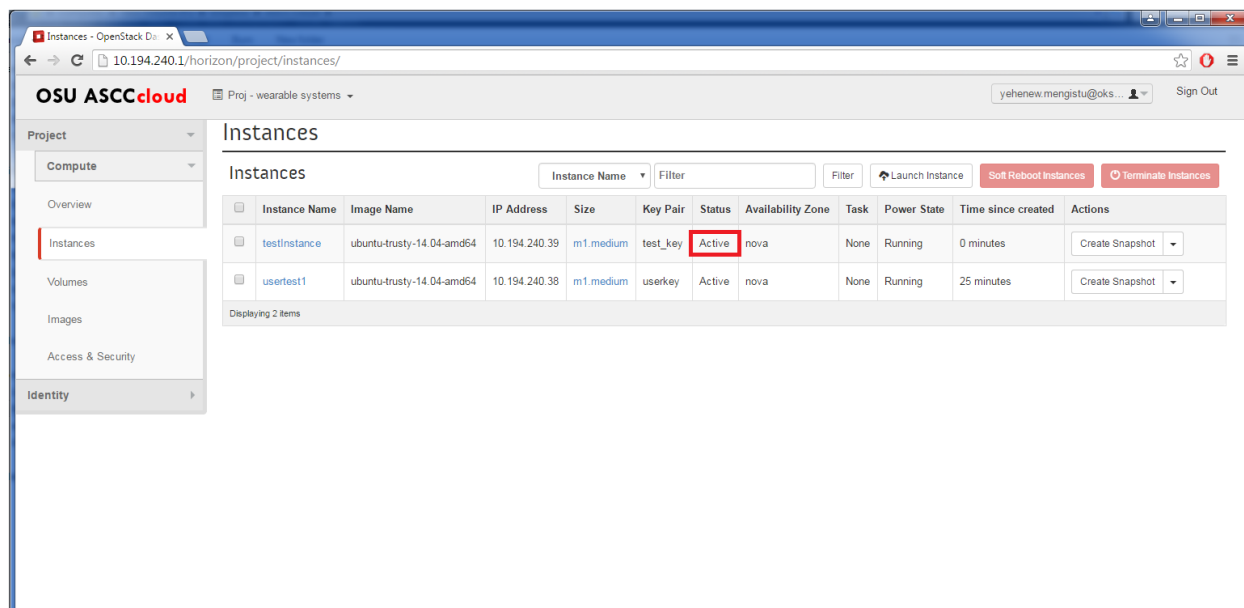


Figure 8. Instance running

Running instances can be restarted using soft or hard reboot options from the actions column as shown in figure 8. A soft reboot attempts a graceful shut down and restart of the instance whereas a hard reboot power cycles the instance.

5. Configuring Security Groups

The default security group requires configuration to allow ICMP and SSH traffic from any host. Navigate security groups from the access & security panel, select the default security group and add **all ICMP** and **all TCP** rules as shown in figure 9. The instance is now ready to be accessed remotely. Additional protocols can also be configured depending on the need.

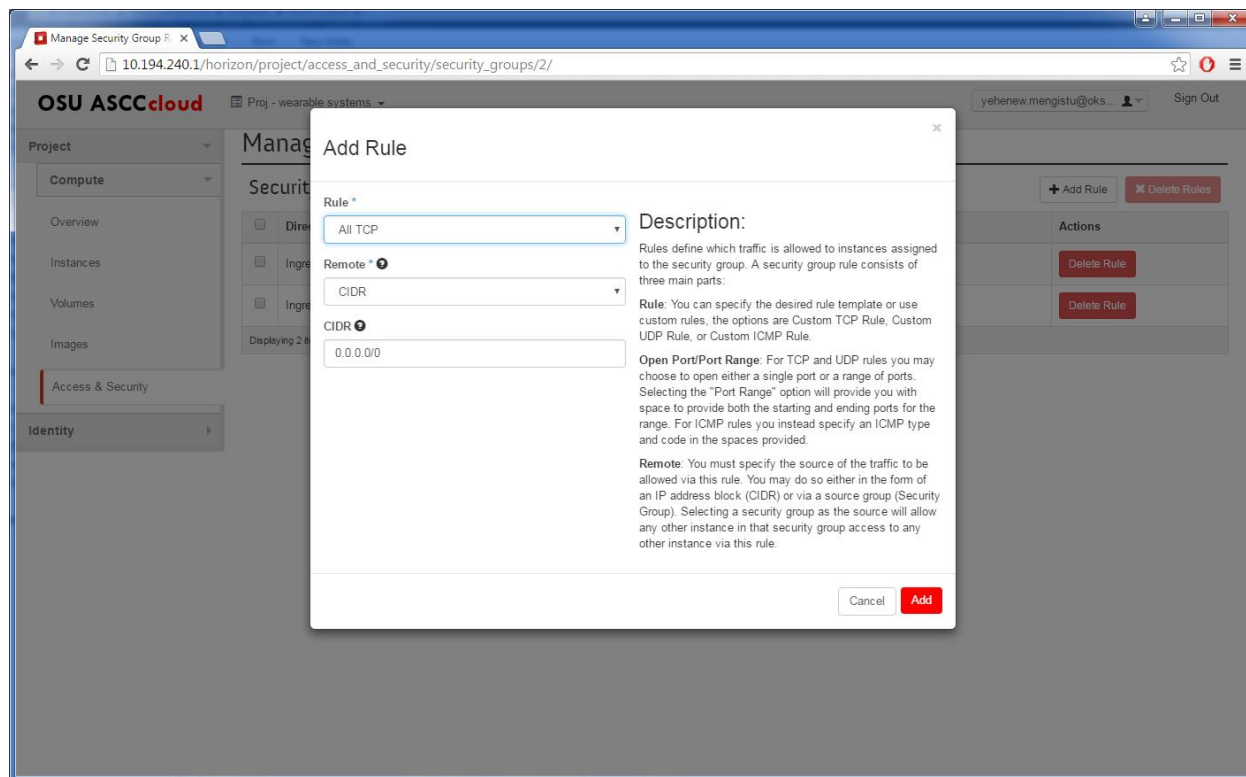


Figure 9. Configure security groups

6. Verifying Connectivity and Remote Access

To check connectivity ping the instance from a local computer. If the computer is not able to ping the instance check network connectivity and contact ASCC admins if problem persists.

The following section describes the steps to remotely access instances using putty. Use the private key saved in section 4 and run **Pageant** authentication agent from Putty folder and add the keys. (Pageant will be docked on the left side of the taskbar along with other icons, right click on it and select view keys to get the wizard shown in figure 10).

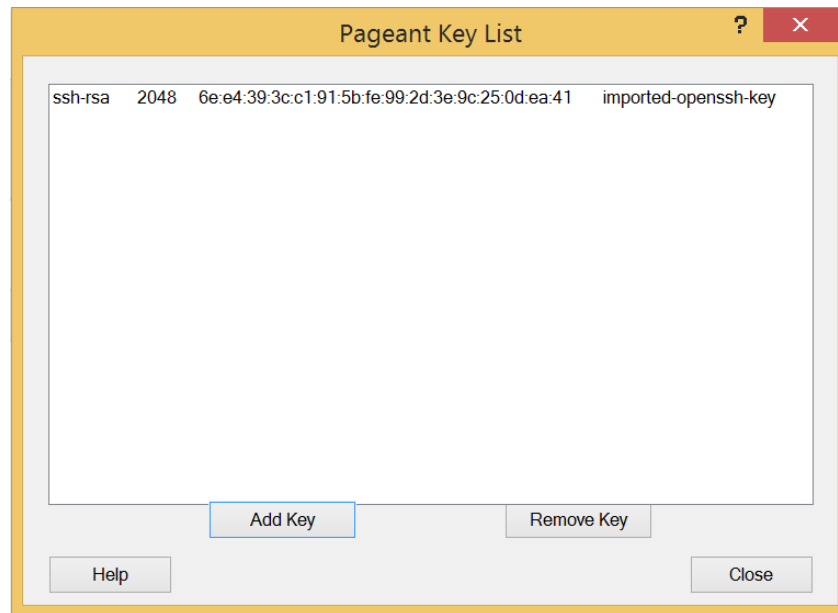


Figure 10. Add private key using pageant

Next, run Putty and configure the following settings to successfully connect to the instance;

- i. Enter the instance IP address
- ii. Set the Auto-login username to either **ubuntu**, **centos** or **Cirros** (based on your instance type) in the connection / data section
- iii. Return to sessions and label and save it for future use
- iv. Navigate to connections / SSH / Auth section and import your key for redundancy and open the connection

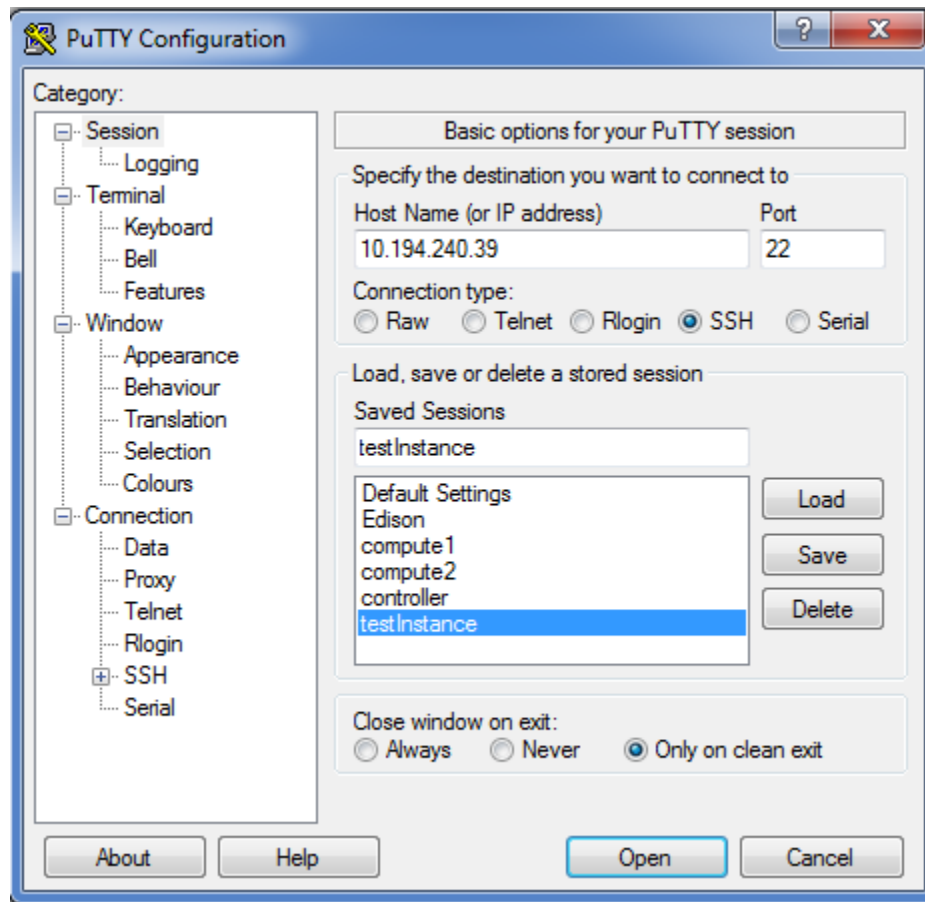
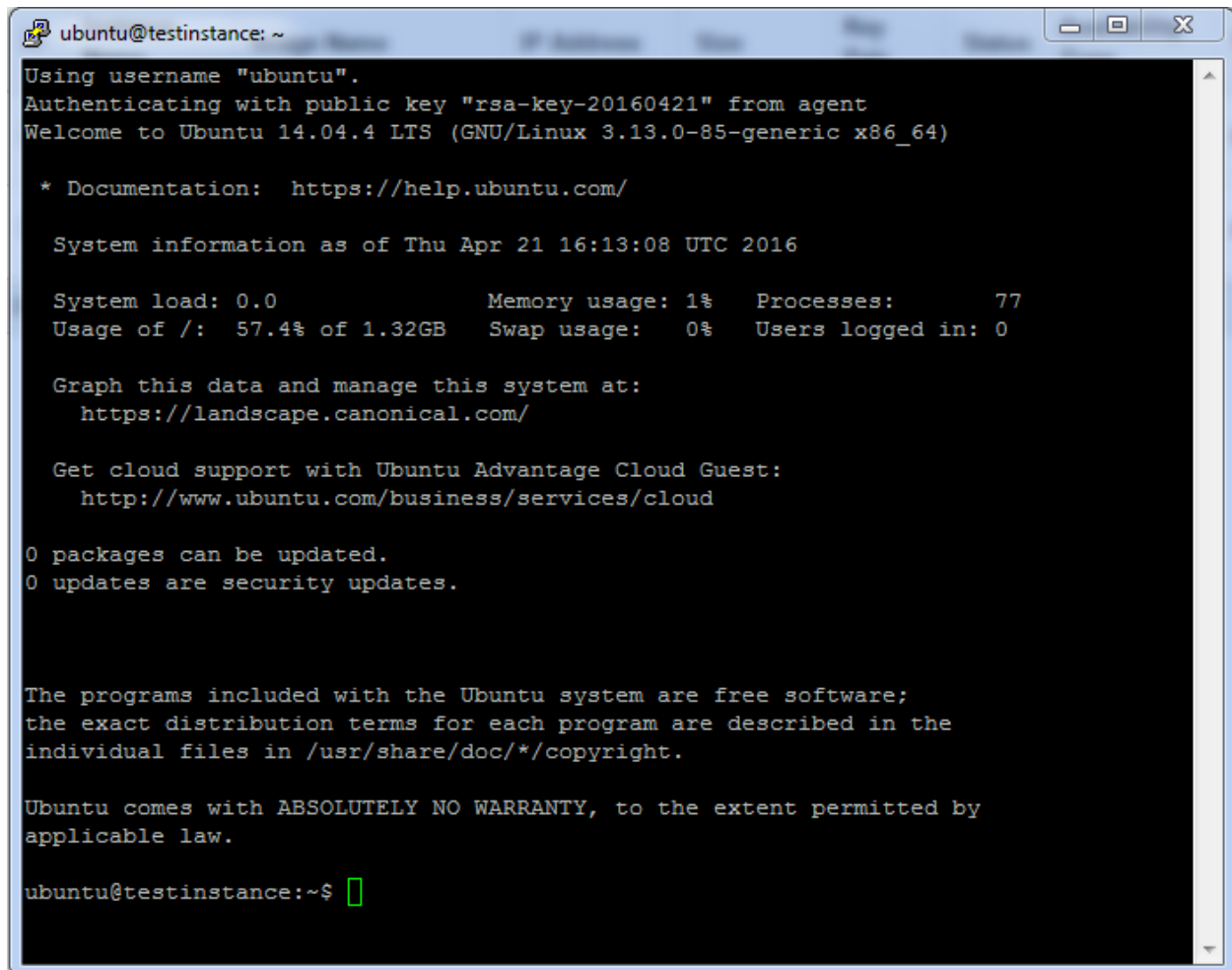


Figure 11. Instance SSH access using Putty

The SSH terminal will open with a security alert at first. After accepting the alert you will be connected to your Server terminal as shown in figure 12.



```
ubuntu@testinstance: ~
Using username "ubuntu".
Authenticating with public key "rsa-key-20160421" from agent
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 3.13.0-85-generic x86_64)

* Documentation:  https://help.ubuntu.com/

System information as of Thu Apr 21 16:13:08 UTC 2016

System load: 0.0           Memory usage: 1%   Processes:      77
Usage of /:  57.4% of 1.32GB Swap usage:   0%   Users logged in: 0

Graph this data and manage this system at:
https://landscape.canonical.com/

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

0 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

ubuntu@testinstance:~$
```

Figure 12. Remote instance terminal

7. Adding Volume Storage

Volume storage will allow a persistent data warehousing for instances. To create an empty volume go to the volumes tab on the control panel and select the create volume option.

The screenshot shows the 'Create Volume' wizard in the OSU ASCCcloud interface. The wizard is a modal dialog box with the following fields and options:

- Volume Name ***: A text input field containing 'backup_vol'.
- Description**: A text area for describing the volume.
- Volume Limits**: Two progress bars showing usage. The first bar is for 'Total Gigabytes (0 GB)' out of '1,000 GB Available'. The second bar is for 'Number of Volumes (0)' out of '10 Available'.
- Volume Source**: A dropdown menu with the option 'No source, empty volume'.
- Type**: A dropdown menu with the option 'No volume type'.
- Size (GB) ***: A text input field containing '10'.
- Availability Zone**: A dropdown menu with the option 'nova'.
- Buttons**: 'Cancel' and 'Create Volume' buttons at the bottom right.

Figure 13. Creating volumes

On the create volume wizard, name the volume and select the size in GB. Set the availability zone to nova and create the volume. Remember volumes sizes are limited to the resource limits allocated to the project. The volume will be ready for use after it displays Available status as shown in figure 14.

The screenshot shows the 'Volumes' page in the OSU ASCCcloud interface. The page displays a table of volumes with the following columns: Name, Description, Size, Status, Type, Attached To, Availability Zone, Bootable, Encrypted, and Actions. The table shows one volume named 'backup_vol' with a size of 10GB and a status of 'Available'. The availability zone is 'nova'.

Name	Description	Size	Status	Type	Attached To	Availability Zone	Bootable	Encrypted	Actions
backup_vol		10GB	Available	-		nova	No	No	Edit Volume

Displaying 1 item

Figure 14. Volume created

To attach the volume to instances, go to the Actions column in the volumes tab, select edit attachments from the drop down menu, select the instance as shown in figure 15 and attach the volume.

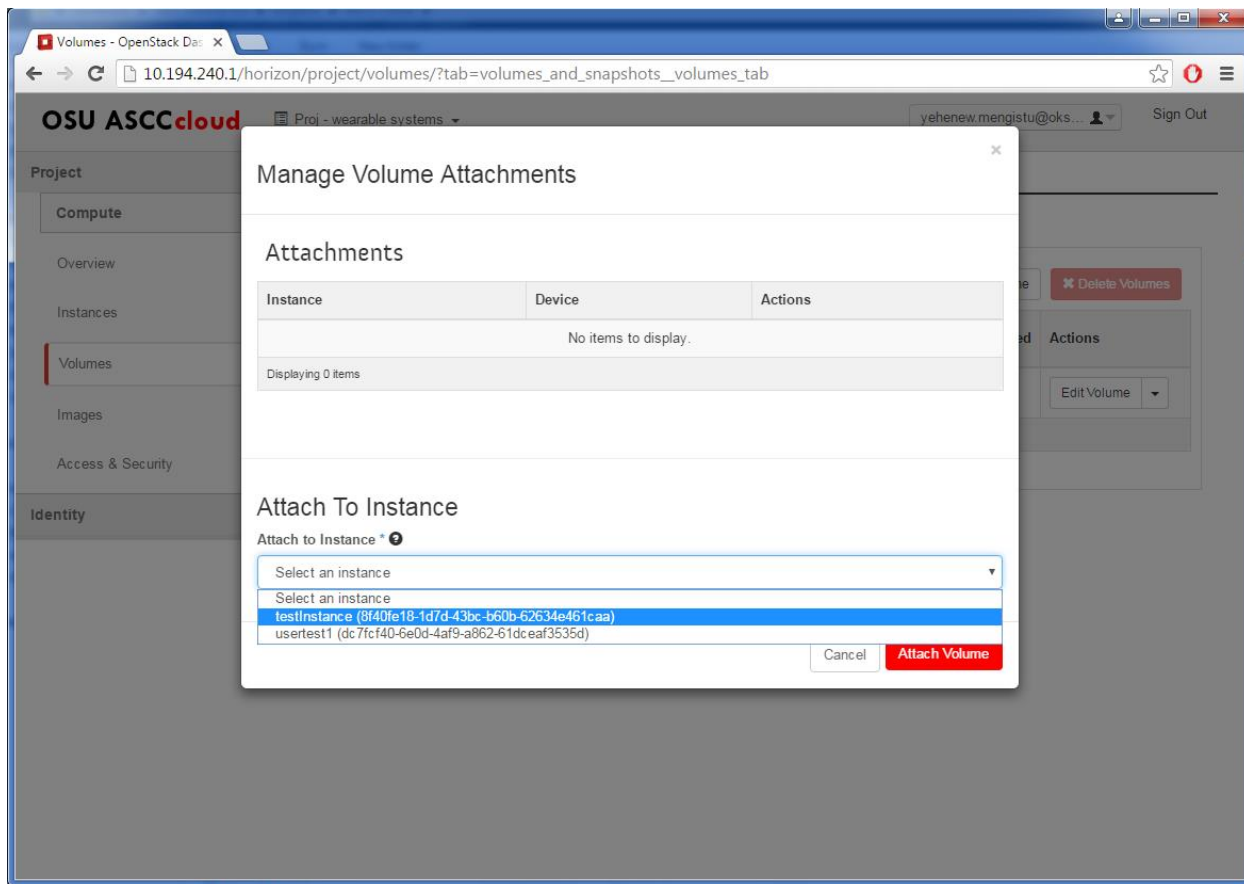


Figure 15. Attaching volumes

The volume will take some time to attach to the instance and once attached the device directory under the instance will be displayed as shown in figure 16.

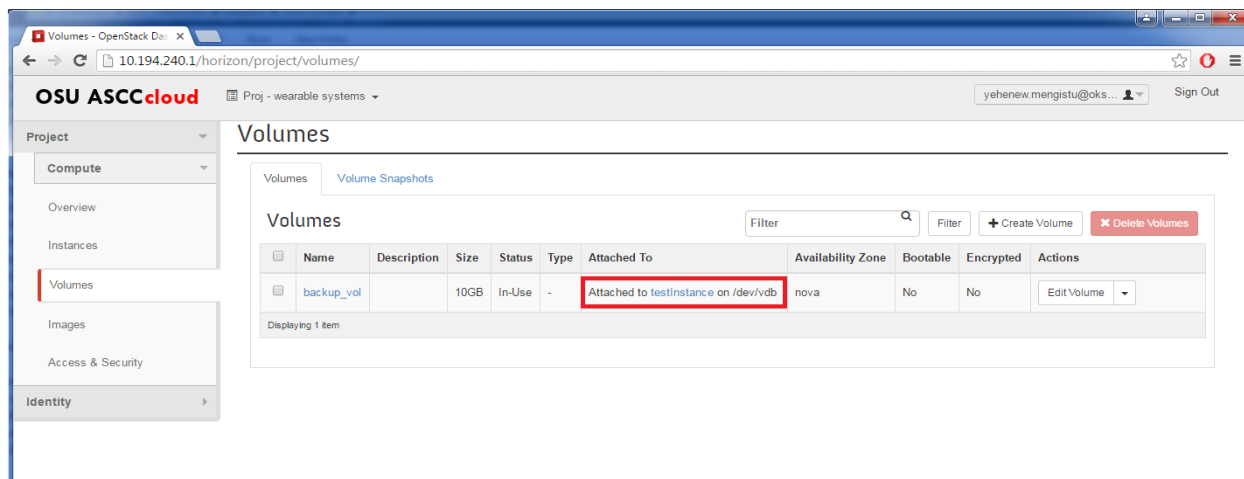


Figure 16. Attached volume device directory

To verify the attachment, go to the terminal and type `fdisk -l` and the volume information will be shown in the device directory as in figure 17. Note `sudo` privilege is required to access this information.

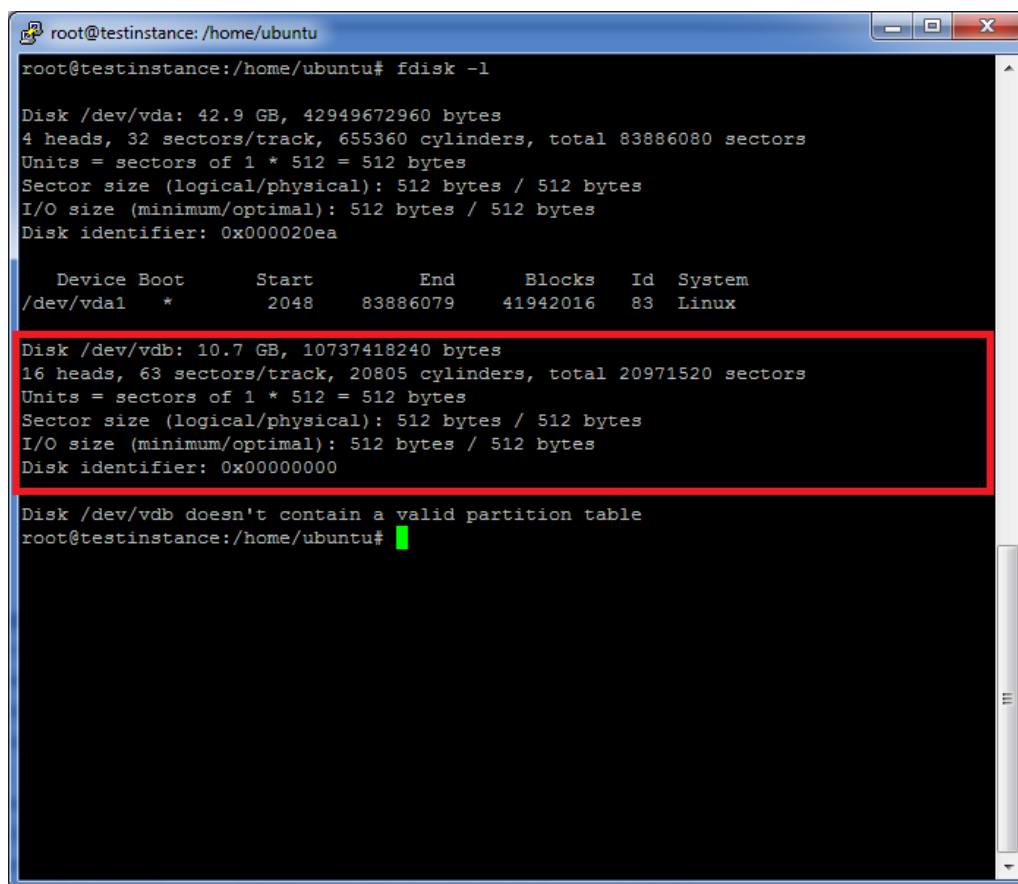
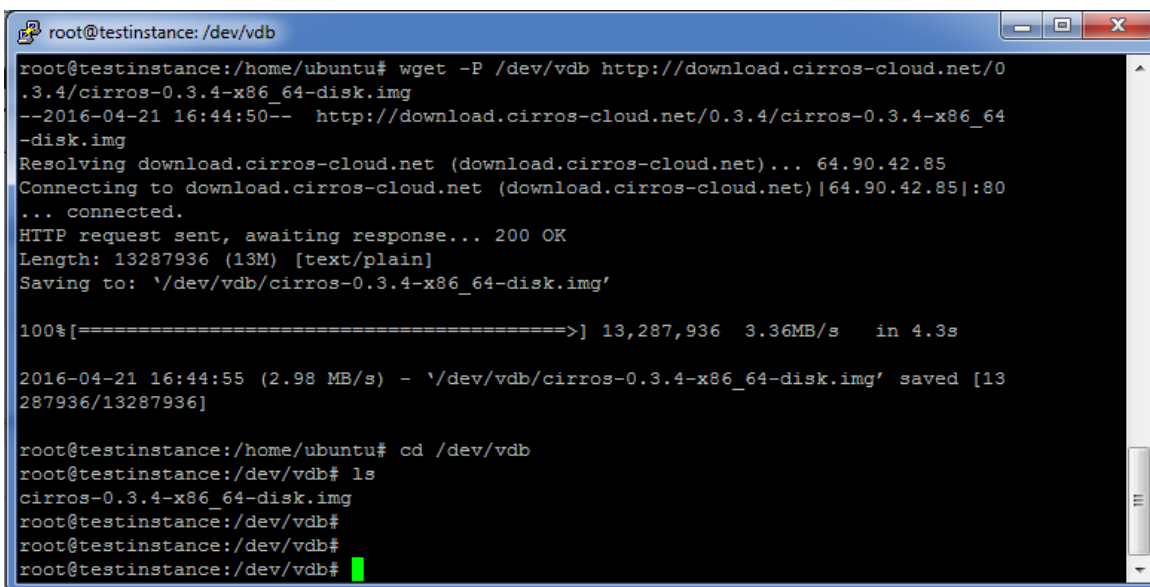


Figure 17. Verifying volume attachment

In addition access to the volume can also be verified by downloading or copying a file to the device directory as shown in figure 18.



```

root@testinstance: /dev/vdb

root@testinstance:/home/ubuntu# wget -P /dev/vdb http://download.cirros-cloud.net/0
.3.4/cirros-0.3.4-x86_64-disk.img
--2016-04-21 16:44:50-- http://download.cirros-cloud.net/0.3.4/cirros-0.3.4-x86_64
-disk.img
Resolving download.cirros-cloud.net (download.cirros-cloud.net)... 64.90.42.85
Connecting to download.cirros-cloud.net (download.cirros-cloud.net)|64.90.42.85|:80
... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13287936 (13M) [text/plain]
Saving to: '/dev/vdb/cirros-0.3.4-x86_64-disk.img'

100%[=====>] 13,287,936 3.36MB/s in 4.3s

2016-04-21 16:44:55 (2.98 MB/s) - '/dev/vdb/cirros-0.3.4-x86_64-disk.img' saved [13
287936/13287936]

root@testinstance:/home/ubuntu# cd /dev/vdb
root@testinstance:/dev/vdb# ls
cirros-0.3.4-x86_64-disk.img
root@testinstance:/dev/vdb#
root@testinstance:/dev/vdb#
root@testinstance:/dev/vdb#

```

Figure 18. Accessing attached volume

This volume will serve as a persistent storage even after the instance is terminated. It can be detached and reattached to other instances using the Manage Volume Attachments wizard from the edit attachments option under the action column of the volumes tab. Figure 19 shows how to detach the volume.

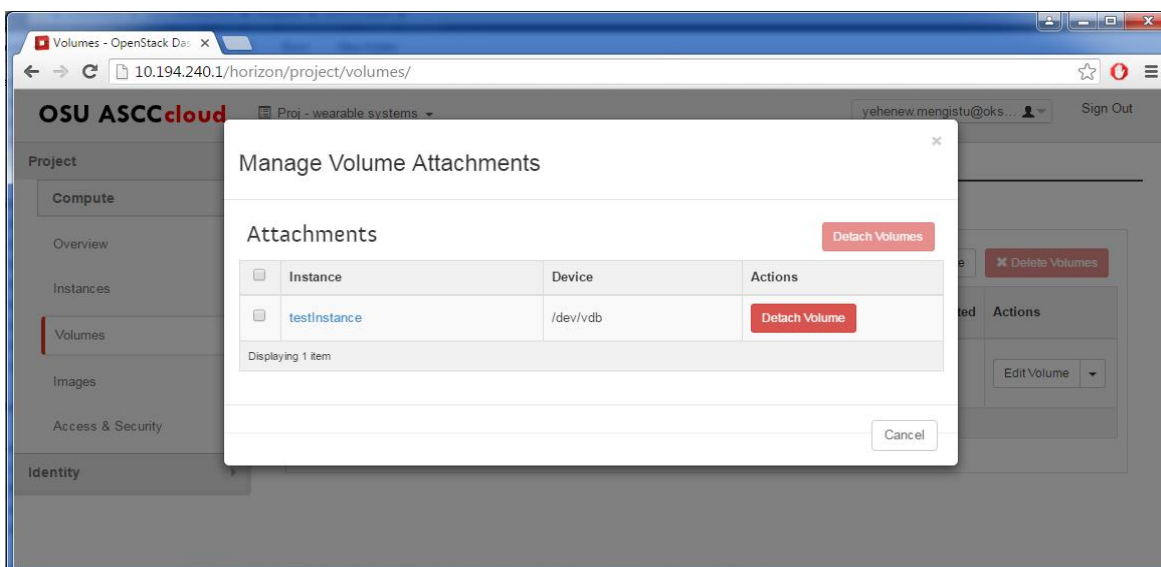


Figure 19. Detaching a volume

8. Instance Backup

Snapshots (Backup images) of running instances can be taken to back up the entire instance before configuration changes. This would be very important, in a development environment going through frequent updates.

To create a snapshot, choose the create snapshot option from the actions column under the Instances tab. Create the snapshot by giving it a name as shown in figure 20.

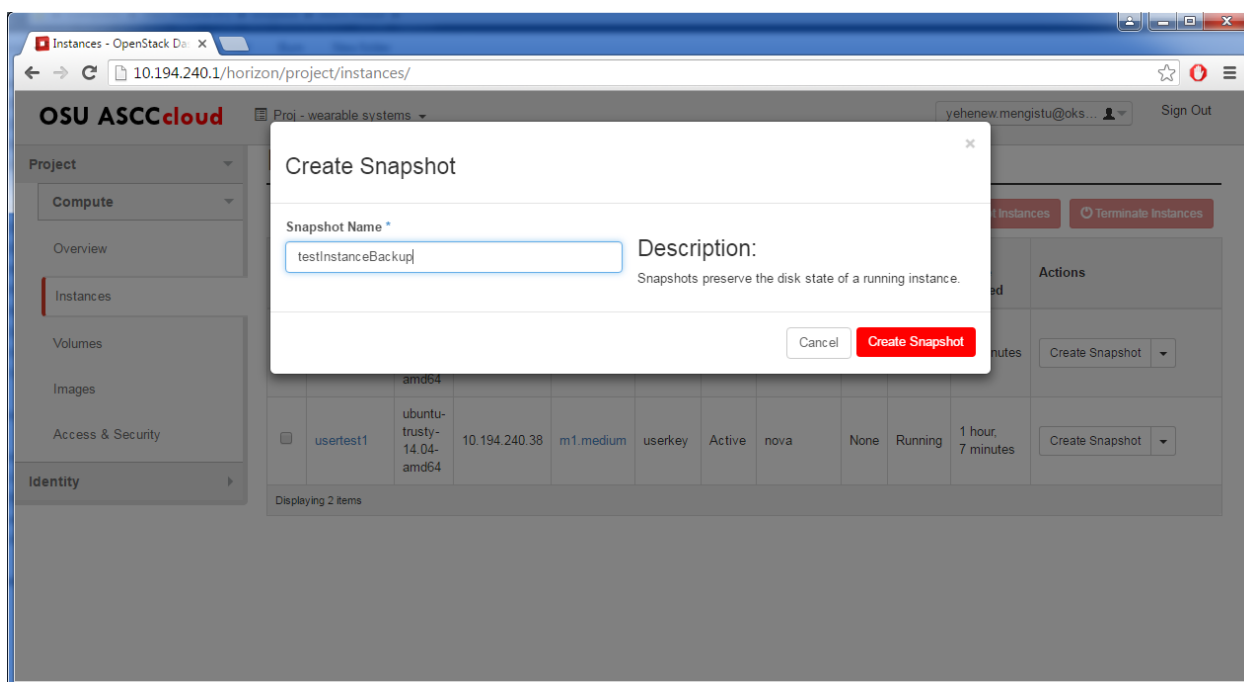


Figure 20. Snapshot Creation

Once created the snapshot goes through different stages and will display Active status. This can be seen under the images tab as shown in figure 21.

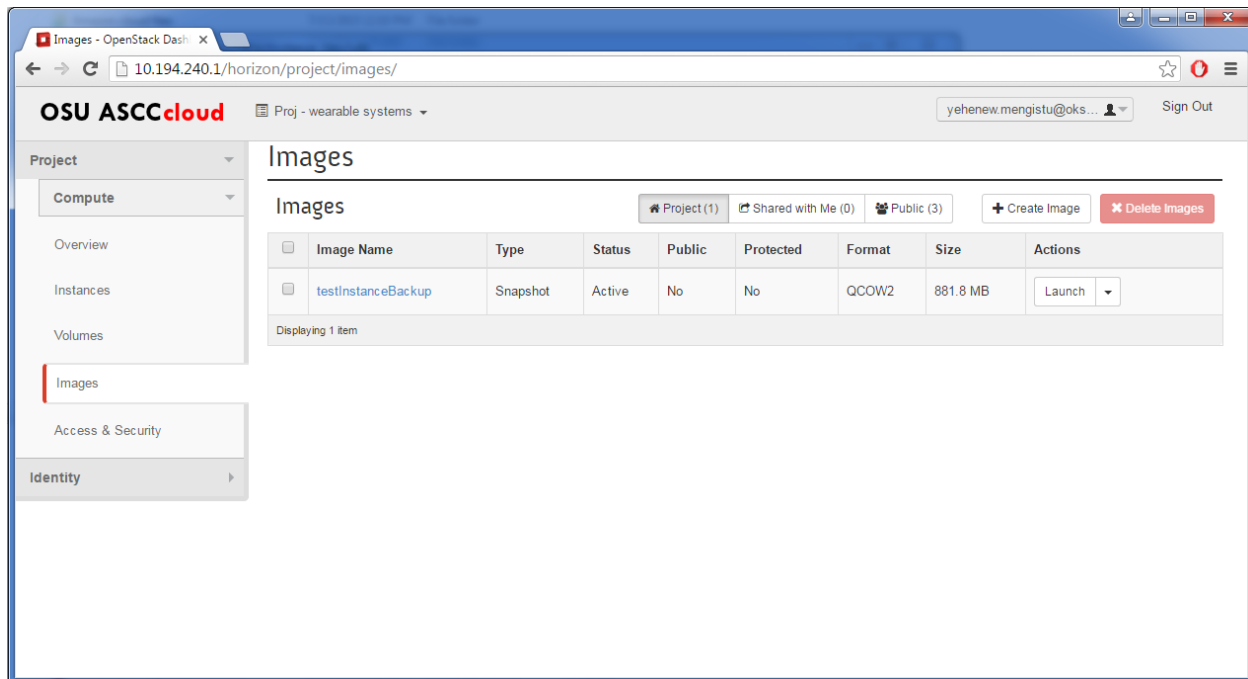


Figure 21. Snapshot created

The backup can be used to launch a new instance or replicate the server configuration using the launch option under the actions column as shown above or by creating a new instance as explained in section 4. The difference here is the instance boot source, the backup snapshot must be chosen to launch the instance. Remember, identical or larger flavor type (from the instance the snapshot is taken) must be chosen for the image to boot.

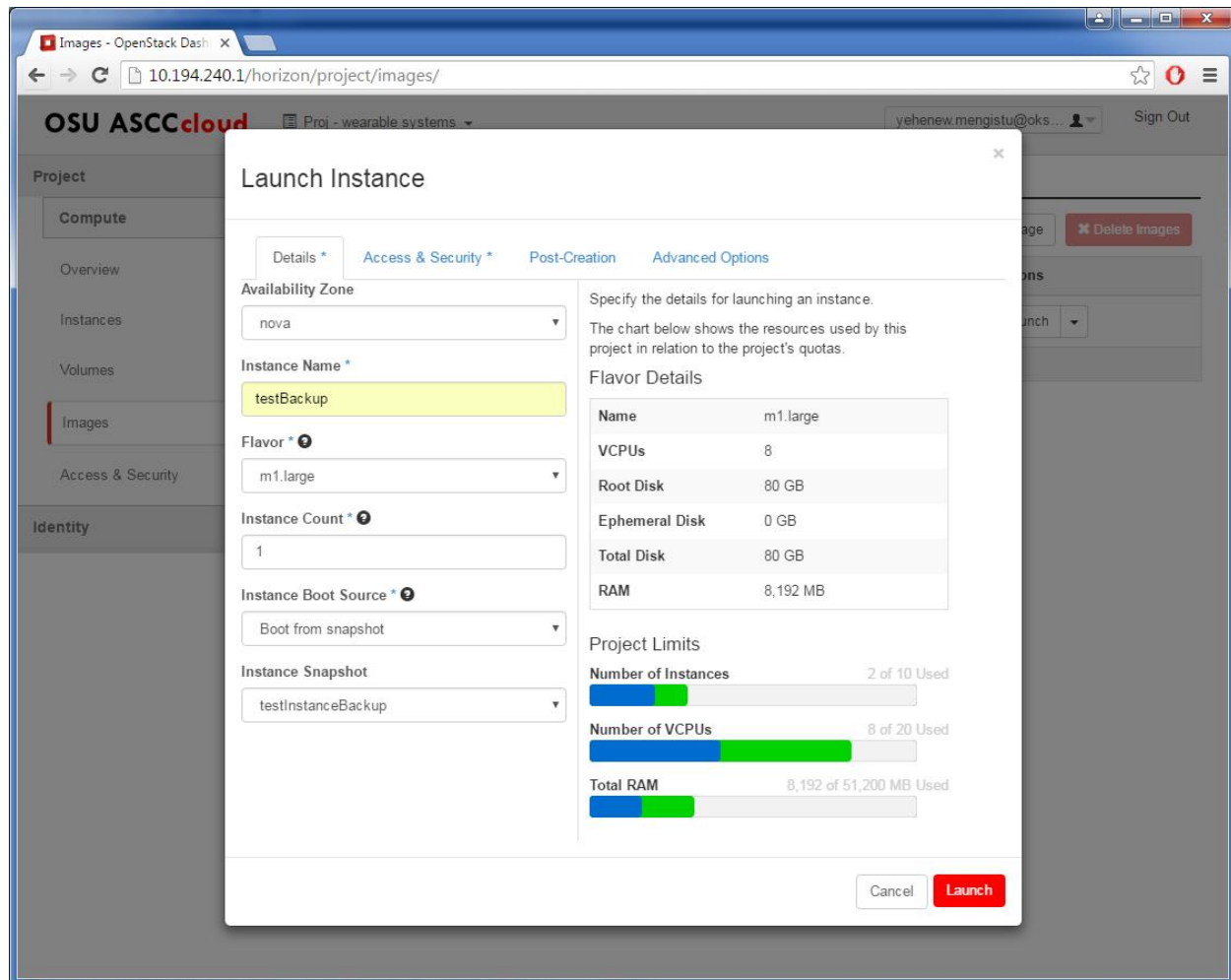


Figure 22. Booting from a backup image

The instance will go through different stages and will display active status as shown in figure 23. The new instance which is a replica of the existing instance can now be accessed using a terminal as described in section 6.

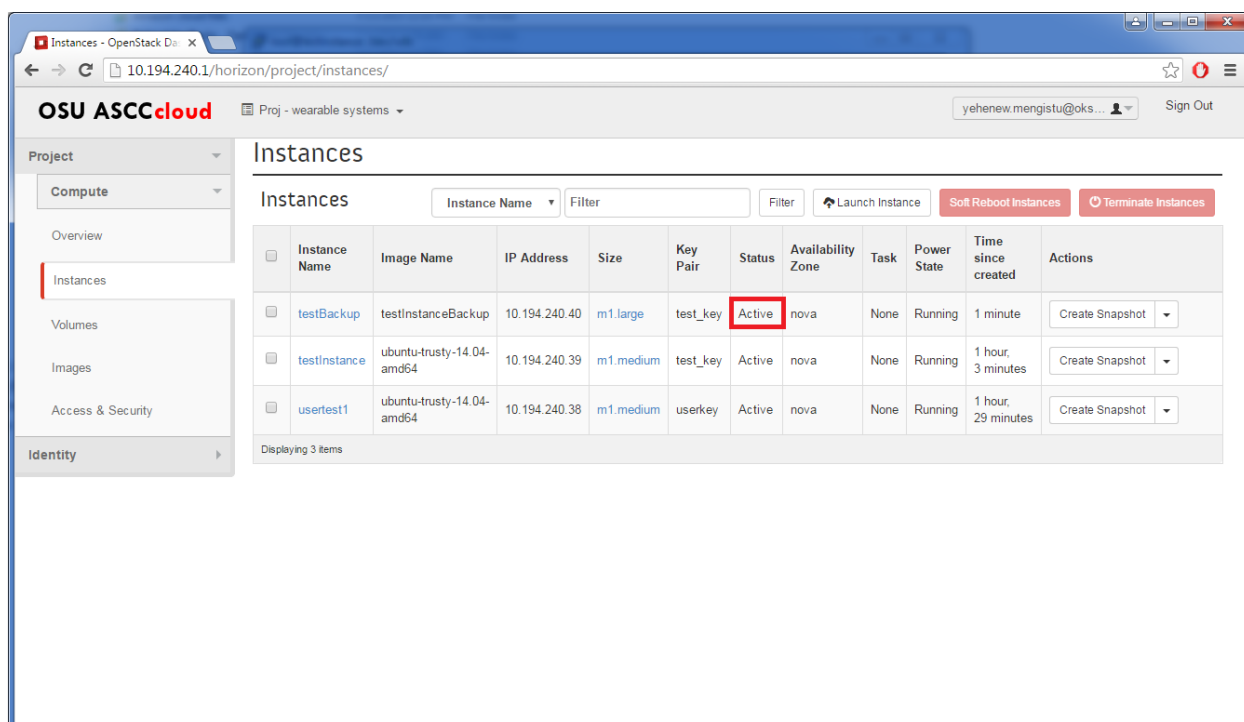


Figure 23. Duplicated instance from a snapshot

9. Adding Desktop environment to Ubuntu 14.04 cloud server

A desktop environment provides a more convenient interface to access a cloud server and this section describes how to add a desktop environment to Ubuntu cloud server and access it remotely.

There are different types of desktops for Ubuntu servers and we recommend using one of the following desktops for use in our servers considering resource utilization.

9.1. Xfce4 Desktop

Xfce is a lightweight desktop environment for UNIX-like operating systems. It aims to be fast and low on system resources, while still being visually appealing and user friendly. It comes with various additional apps and panel plug-ins which greatly enhance the functionality of the DE.

To install Xfce on the instance, use the following commands,

```
sudo apt-get update  
sudo apt-get install -y xfce4 xfce4-goodies
```

After installing the desktop restart your system and go to section 9.3 on how to remotely access the Desktop. The Xfce desktop has the interface shown in figure 24.

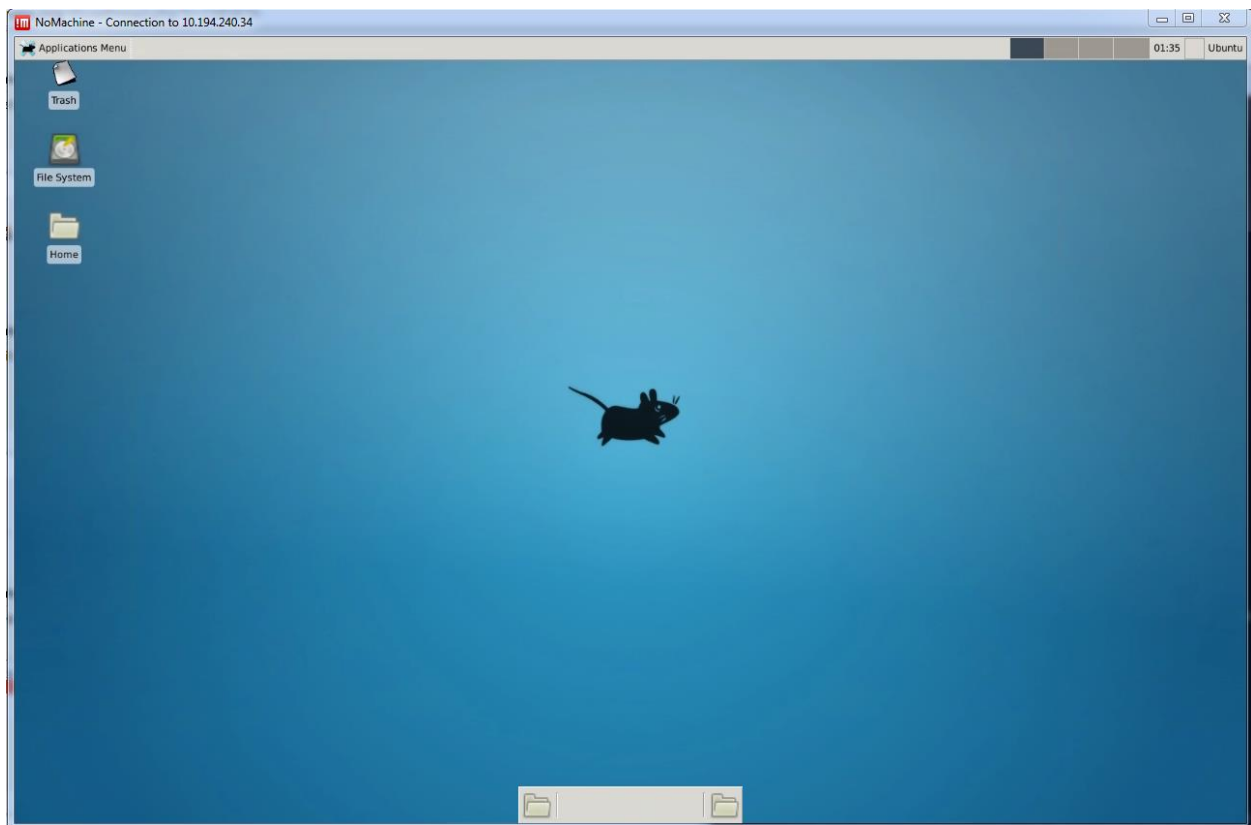


Figure 24. Xfce4 Desktop

9.2. Gnome Desktop

Gnome has different versions and use the following command to install a minimal gnome core desktop.


```
sudo apt-get install xorg gnome-core gnome-system-tools gnome-app-install
```

After installing the desktop restart your system and go to section 9.3 on how to remotely access the Desktop. The Gnome desktop has the interface shown in figure 25.



Figure 25. Gnome Desktop

9.3. Remote Desktop Access

A terminal program called nomachine which uses NX protocol will be used in this guide to remotely access desktops.

The direct download link from Ubuntu terminal is not working as of this writing so we recommend downloading the x64 version .deb file from the link below on a windows machine and copying it to your virtual machine using file sharing tools like winscp or filezilla.

<https://www.nomachine.com/download/download&id=1>

Once the file is ready use the following command to install it

```
sudo dpkg -i nomachine_5.1.9_6_amd64.deb
```

Remember to change the file name for the installer. After nomachine server is installed it will be listening to connections on port 4000.

Next create a password for the user account using the following command.

```
sudo passwd ubuntu
```

Type a suitable password and once the password is changed the server is up and ready for a remote connection. Next download the nomachine client on a local machine from the following link and install it.

<https://www.nomachine.com/download/download&id=22>

After installation choose the new option and select NX protocol to create a connection as shown in figure 26.

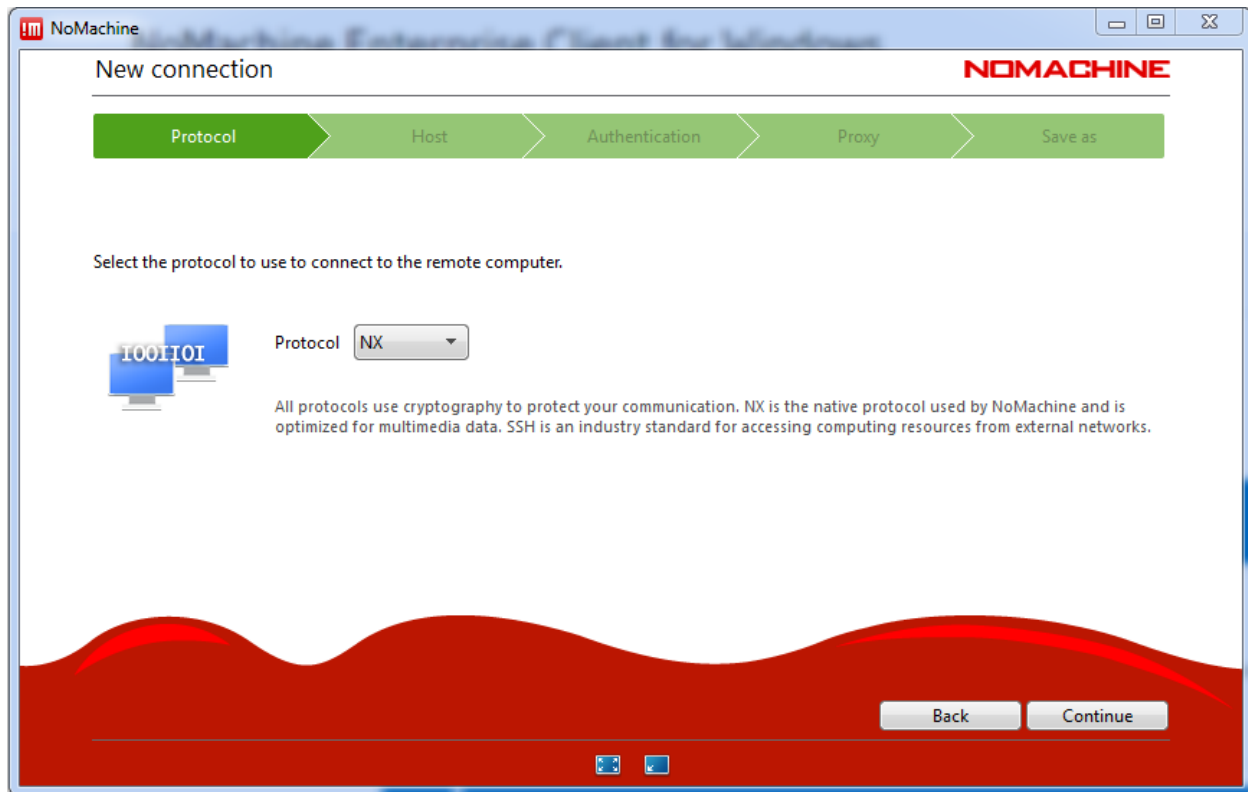


Figure 26. Creating a connection from nomachine client

Enter the instance IP address and choose a password authentication type in the following pages. Select do not use Proxy in the fourth tab and save the connection.

To start remote access, select the connection as shown in Figure 27 and connect. Default settings can be used for the display and devices and finally the desktop environment will be displayed as shown in figures 24 and 25.

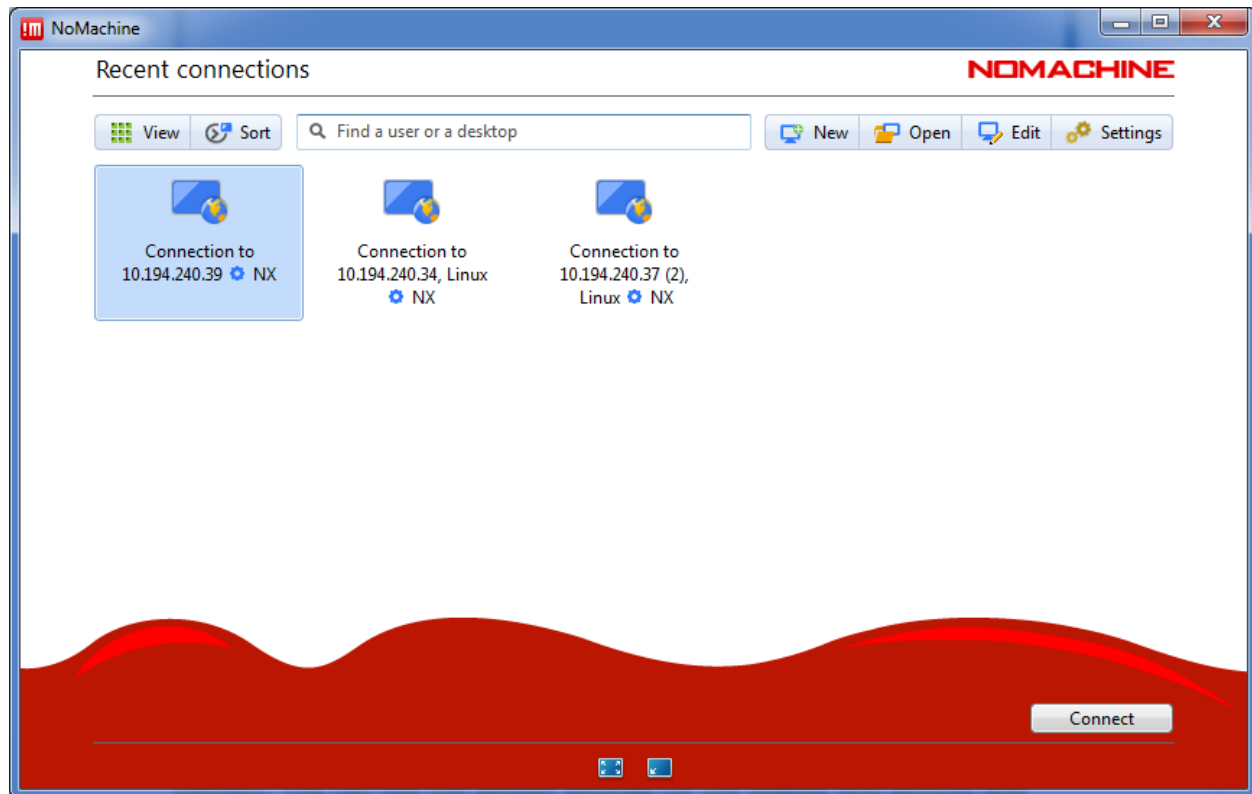


Figure 27. Accessing remote Desktop

10. Deleting Instances

To delete instances first remove attached volumes as described in section 7. Terminating the instance will permanently wipe out the data on it and therefore it is advised to back up important data to a volume drive. To remove the instance select the terminate instance action from the instance tab as shown in figure 28 and confirm the termination.

The screenshot shows the 'Instances' page in the OSU ASCC Cloud interface. The left sidebar contains navigation links: Project, Compute, Overview, Instances (selected), Volumes, Images, Access & Security, and Identity. The main content area displays a table of instances with columns: Instance Name, Image Name, IP Address, Size, Key Pair, Status, Availability Zone, Task, Power State, Time since created, and Actions. Three instances are listed: testBackup, testInstance, and usertest1. The 'testInstance' row is selected, and its actions menu is open, showing options like Associate Floating IP, Disassociate Floating IP, Edit Instance, Edit Security Groups, Console, View Log, Pause Instance, Suspend Instance, Resize Instance, Soft Reboot Instance, Hard Reboot Instance, Shut Off Instance, Rebuild Instance, and Terminate Instance.

Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
testBackup	testInstanceBackup	10.194.240.40	m1.large	test_key	Active	nova	None	Running	3 hours, 30 minutes	Create Snapshot
testInstance	ubuntu-trusty-14.04-amd64	10.194.240.39	m1.medium	test_key	Active	nova	None	Running	4 hours, 33 minutes	Associate Floating IP Disassociate Floating IP Edit Instance Edit Security Groups Console View Log Pause Instance Suspend Instance Resize Instance Soft Reboot Instance Hard Reboot Instance Shut Off Instance Rebuild Instance Terminate Instance
usertest1	ubuntu-trusty-14.04-amd64	10.194.240.38	m1.medium	userkey	Active	nova	None	Running	4 hours, 59 minutes	

Figure 28. Deleting instances