Openstack core components was installed with Ministack and was running into a virtual server on Linux Ubuntu Server. The Horizontal dashboard was used for following 10 tasks:

Task 1 – Creating Virtual Networks

A virtual network (Figure.1) was built comprised of a public network (10.20.20.0) and two private networks which are MeiPrivNetA and MeiPrivNetB. Each of these two private networks were connected to the public network via RouterA and RouterBrespectively.

Figure.1

Steps to build this virtual networks are:

1.1 Select **Network > Network Topology > + Create Network (1.f)**

1.2 Then filled the **Network Name** as per MeiPrivNetA. Later click next to filled up **Subnet** as MeiSubnetA with **network ip address 10.0.0.0/24** and left others defaults to complete the creation of Networks. Same steps repeated for network MeiPrivNetB with MeiSubnetB.

1.3 Then select **+ Create Router,** and filled up the **Router Name** as RouteA and left all defaults and select **Create Router**. Then a router connected to public network will appeared in the screen of network topology. Then **Right click** on the router and select **Add interface** where a box shown and select MeiPrivNetA for **Subnet** and complete with **submit** button. Same steps repeated for creation of RouteB. Lastly a complete networks appeared as Figure.1.

Task 2 – Creating Instances

Three instances with name MeiInst1, MeiInst2 and MeiInst3 were launched as shown Figure.2.1 for the virtual networks.

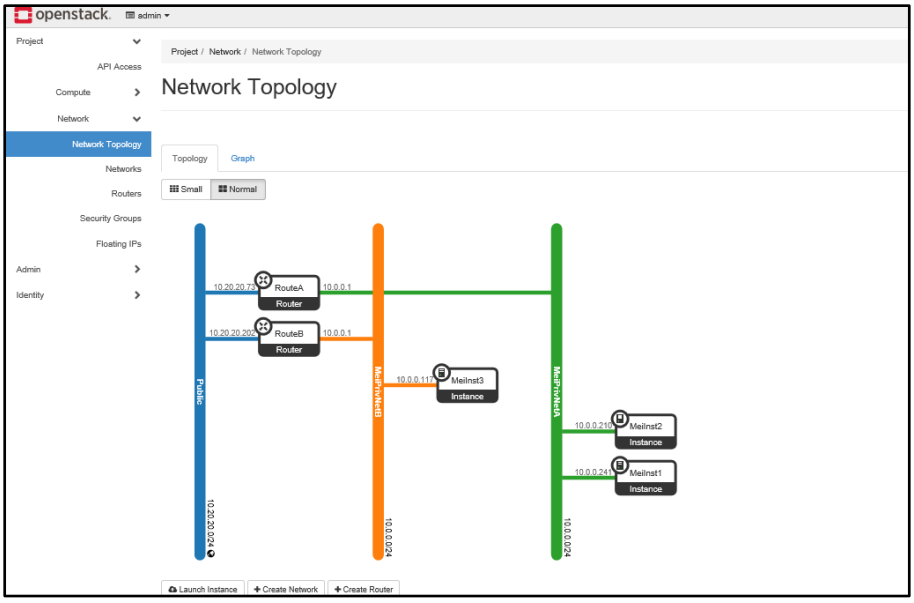

Figure.2.1

Steps to create these instances are:

2.1 **Select Compute > Instances > Launch Instance.**

2.2 As per Figure 2.2, a screen with specifications have to be filled up for each instances. Firstly, create instance's name such as MeiInst1 and keep all as default and then click next.

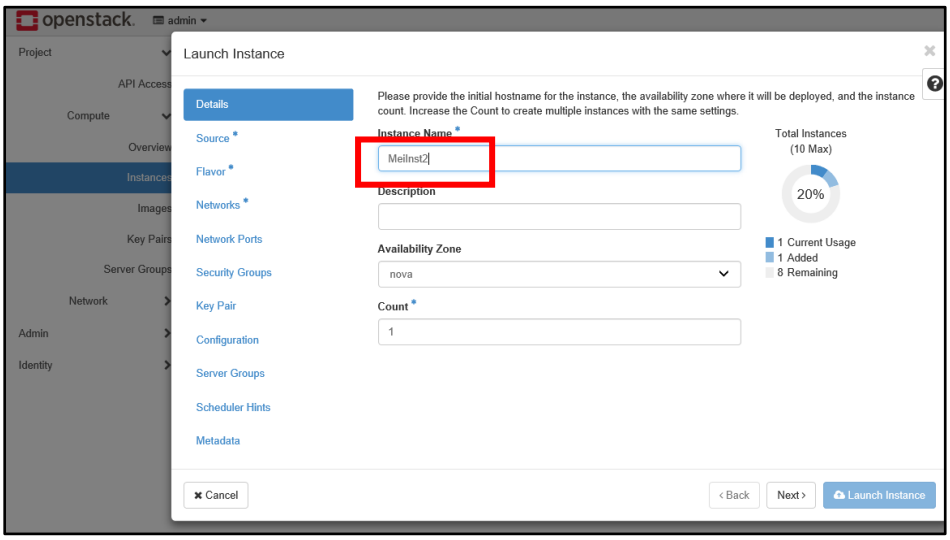

Figure.2.2

2.3 Then **cirros** as the default of Linux operating system was selected by clicking up arrow for the **Source** specification as per Figure 2.3. Then click next button.

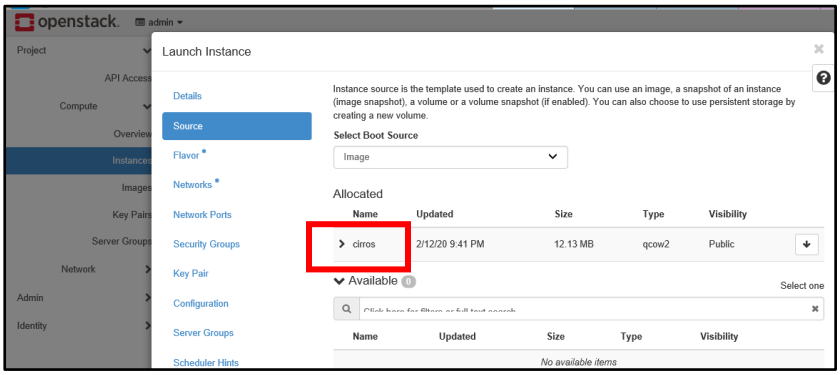

Figure.2.3

2.4 This followed by the specification of **Flavor** which system with **1 CPU, 512MB of RAM, 1GB of disk** was selected by clicking up arrow as per Figure 2.4. Then click next button. Due Openstack installed with Ministack, there were not available Linux Cirros operating system as per assignment required 1 CPU, 64MB of RAM, 1GB of disk.

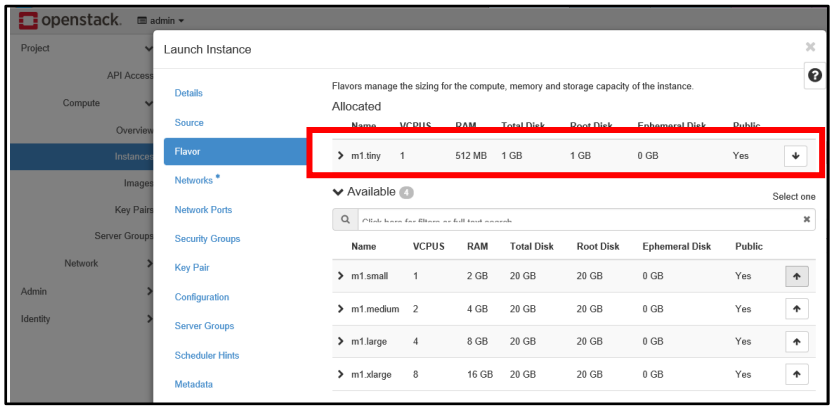

Figure.2.4

2.5 Networks specification for MeiInst1 and MeiInst2 was clicked for up arrow of MeiPrivNetA, whereas MeiInst3 was connected to MeiPrivNetB as Figure.2.5 which both networks are not public visible. Then click next button.

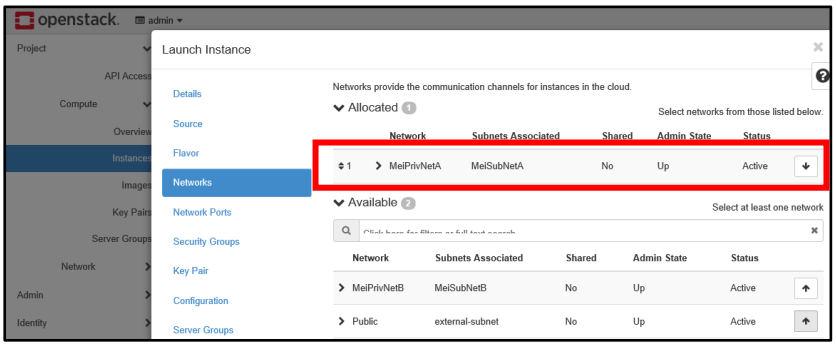

Figure.2.5

2.6 On the specification on Security Groups, **default** and **Mei Security group** were selected to rule the instances as per Figure.2.6. Mei Security group creation will be shown in Task3.

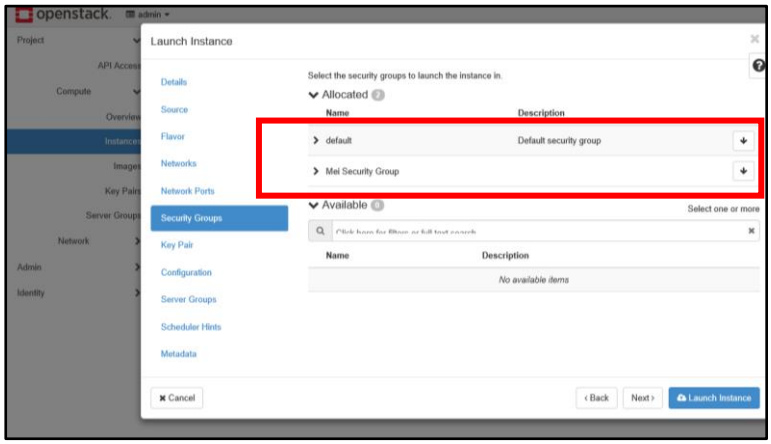

Figure.2.6

2.6 On the specification on Key pair, each instances were specified to each key as shown in Figure.2.7, such as **Mei_Akey** was selected for **MeiInst1**, **Mei_BKey** for **MeiInst2** and lastly **Mei_CKey** for **MeiInst3** as Figure.2.8. Key Pair creation steps shown in Task3.

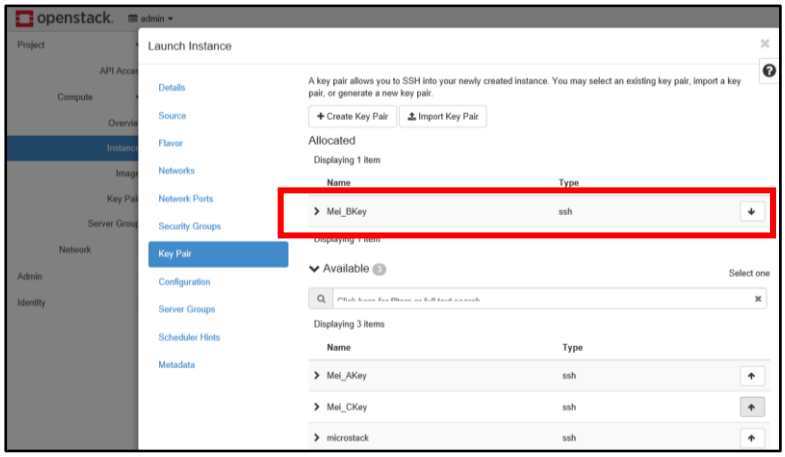

Figure.2.7

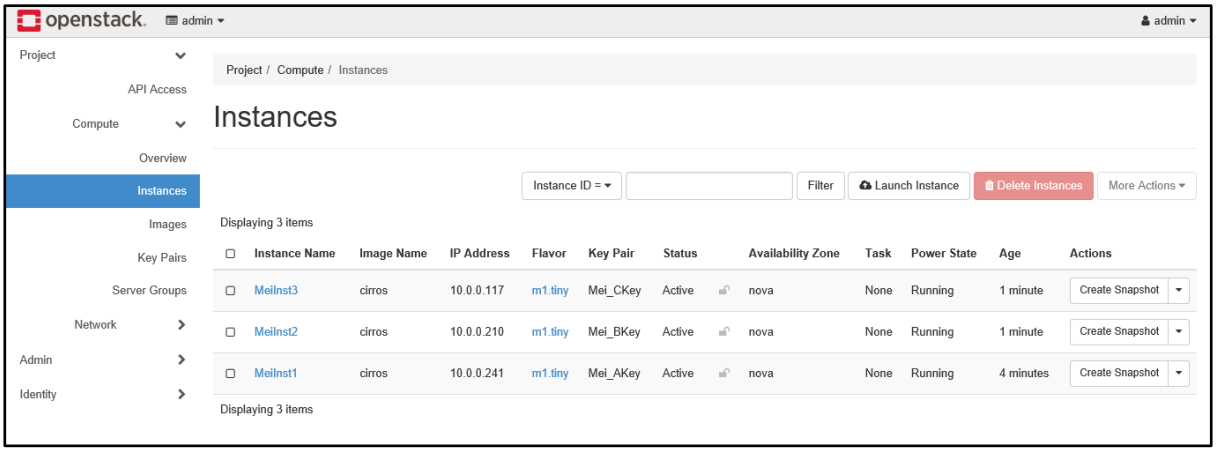

Figure.2.8

Task 3 – Instances and Networks

As per Figure 2.1, first and second instances MeiInst1 and MeiInst2 were connected to MeiPrivNetA, whereas third instance MeiInst3 was connected to MeiPrivNet2. This were done in steps of **Task 2** when **Launching instances** creation **to specify the networks** as per Figure.3.

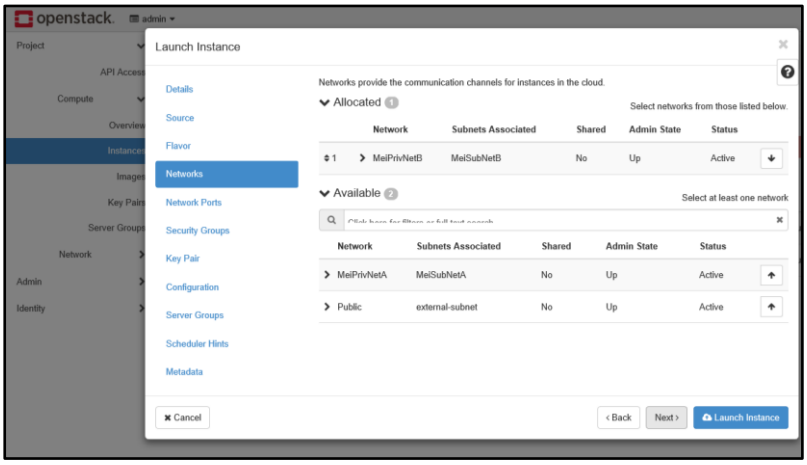

Figure.3

Task 4 – Managing Security

Steps to create security groups and key pairs are:

4.1 A security network of **Mei Security Group** as in Figure.4.1 was built with specific rules. By select **Network > Security Groups > + Create Security Group.**

4.2 Upon creation of the group select the **manage rules** and this will lead you to the screen of Figure.4.1. Then click on **+ Add Rule** button to add **ingress** direction specifies the inside network traffic for by enabling **ICMP** (for ping) and **SSH.**

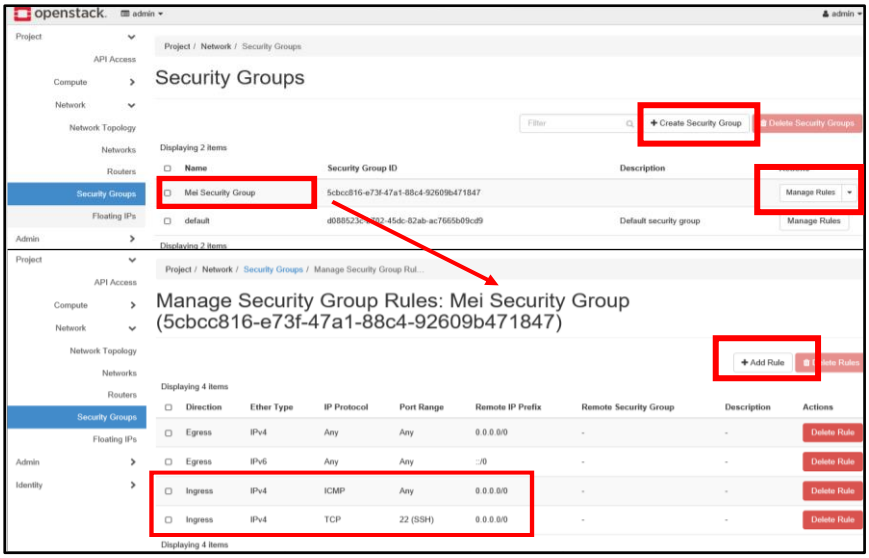

Figure.4.1

4.3 From **Compute > Key pairs > + Create Key Pairs**.

3 keys were created which are Mei_AKey, Mei_BKey and Mei_CKey as below example of screenshot Figure.4.2 These keys were save in the local machine.

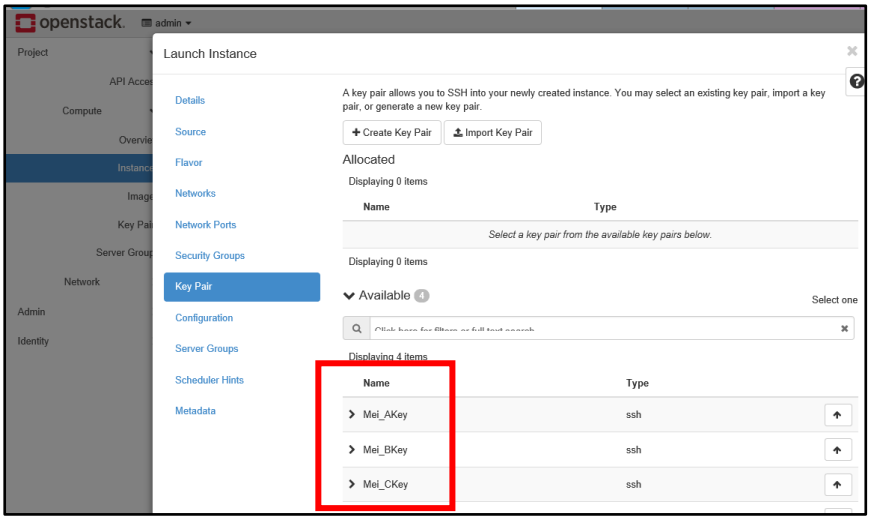

Figure.4.2

Task 5 – Associate of Floating IP

Upon creations of instances as Figure 5.1, the were private IP address associated to each instances. MeiInst1 is associated with IP address 10.0.0.241, MeiInst2 with 10.0.0.210 and MeiInst3 with 10.0.0.117. Then to associate floating IP address to access instances via public network.

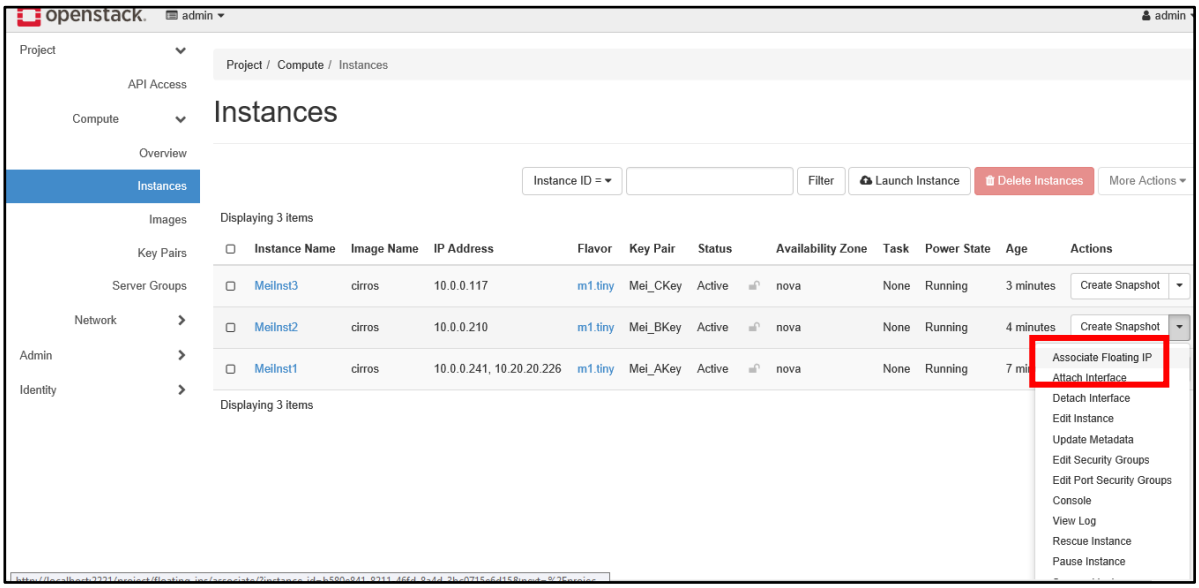

Figure.5.1

Steps to associate floating IP for instances are:

5.1 By clicking the rex box as Figure.5.1 will bring you to screen Figure.5.2 and then click **Associate** button.

5.2 Figure.5.2 shown the Allocating Floating IP screen which Pool filled with Public and to click **Allocate** button.

5.3 Steps 5.1 and 5.2 were repeated for each instance to allocation of the floating IP which will be final shown in Figure.5.3.

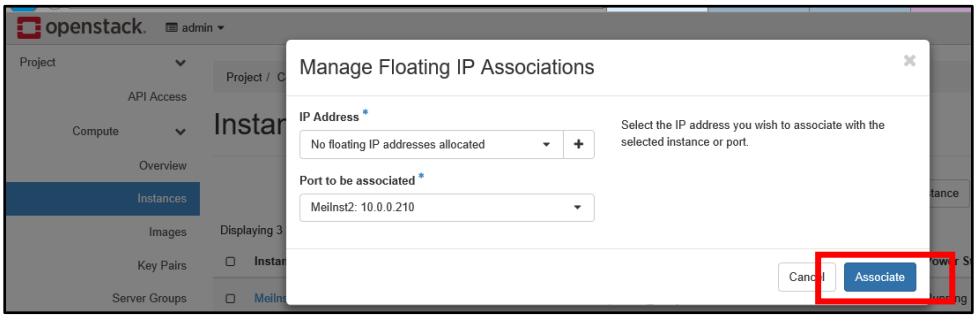

Figure.5.1

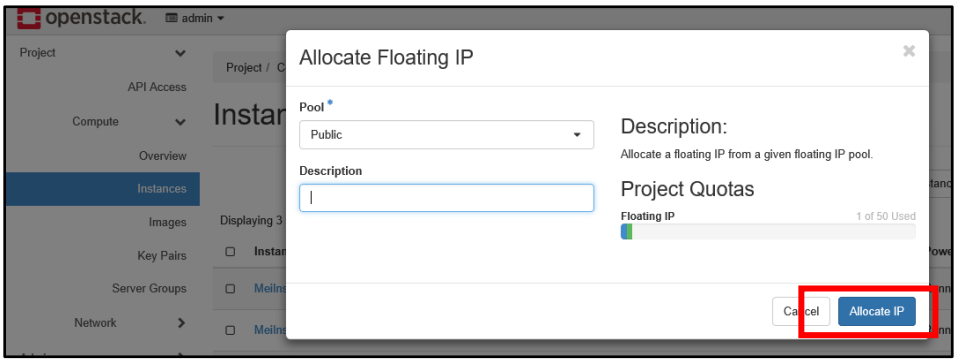

Figure.5.2

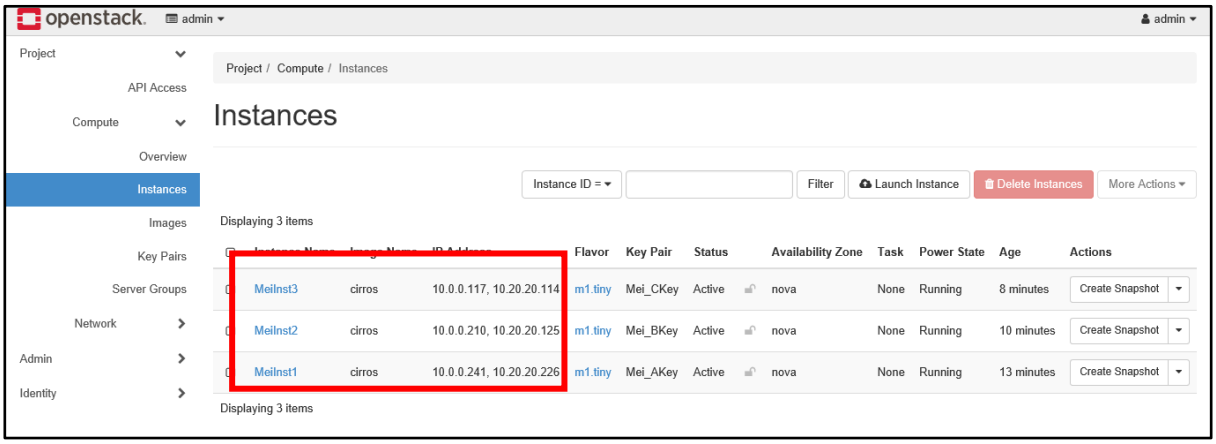

Figure.5.3

Mei Tan Le Ping, lt528@live.mdx.ac.uk
M00724895

Task 6 – Connecting Instances

SSH connection to the server via puTTY, then Ping of the IP addresses of each instances provided the floating IP address as Figure.6. Ping were succeeding as the Mei security group selected had been added with ICMP rule as in Figure.4.1 of Task4.

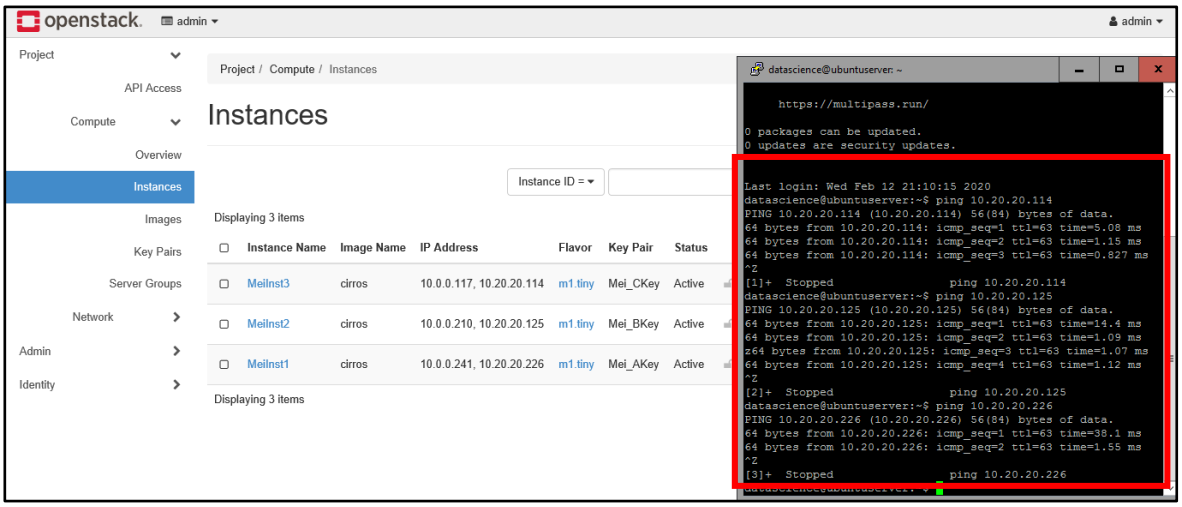

Figure.6

Task 7 – Run Instance via Remote Desktop

Compute > Instances > click the instances > Log were accessed to acquire the password of each instances such as with login username cirros (password 'gocubsgo') as credentials. This will be used in puTTY to connect SSH to each instances via floating IP address with command of the highlighted in red box "ssh cirros@10.20.20.226" as an example as Figure.7.

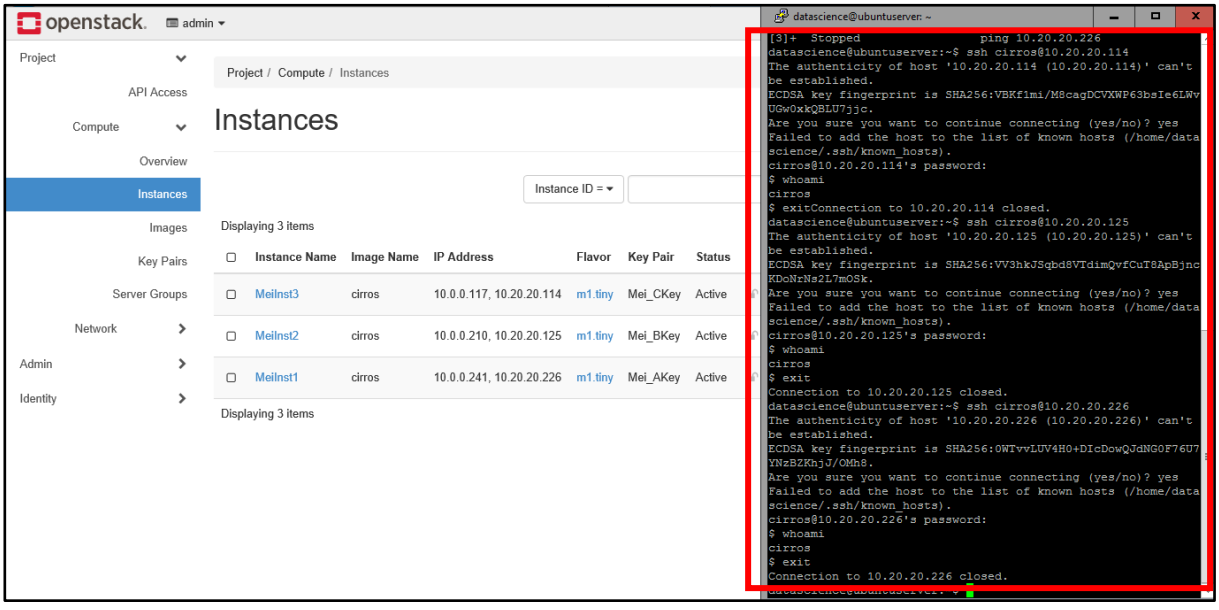

Figure.7

Task 8 – Create New Folder for Instance 1 and Check permission

As per Figure.8, upon SSH connected via the **floating IP address** (10.20.20.226) 0f instance **MeiInst1** as highlighted in red box, **MeiMyFolder** is created with command **mkdir**, whilts command **ls- l** was used to execute the permission checking of the folder.

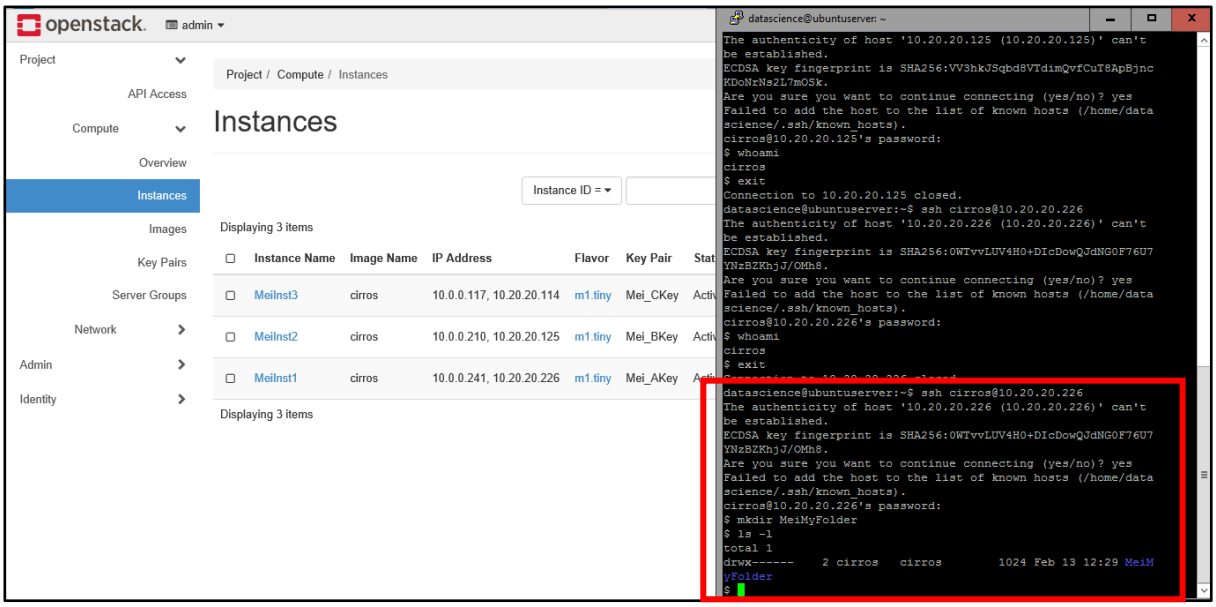

Figure.8

Task 9 – Python File creation

As continuous from Task 8 creation of Folder of MeiMyFolder in MeiInst1, a Python file with name **MeiMyProgram.py** was created with command **application of vi** as per in Figure.9.1. Then click enter will bring to screen as in Figure9.2 left screen which to create simple instruction of print ("Hello World"). Figure9.2 right screen shown the succeed of creation of the python file in instance 1, MeiInst1.

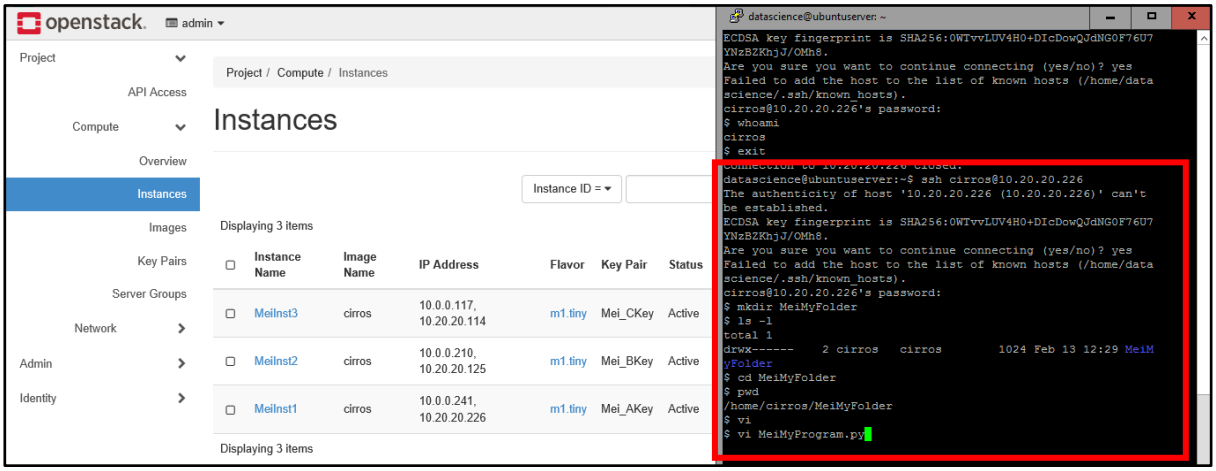

Figure.9.1

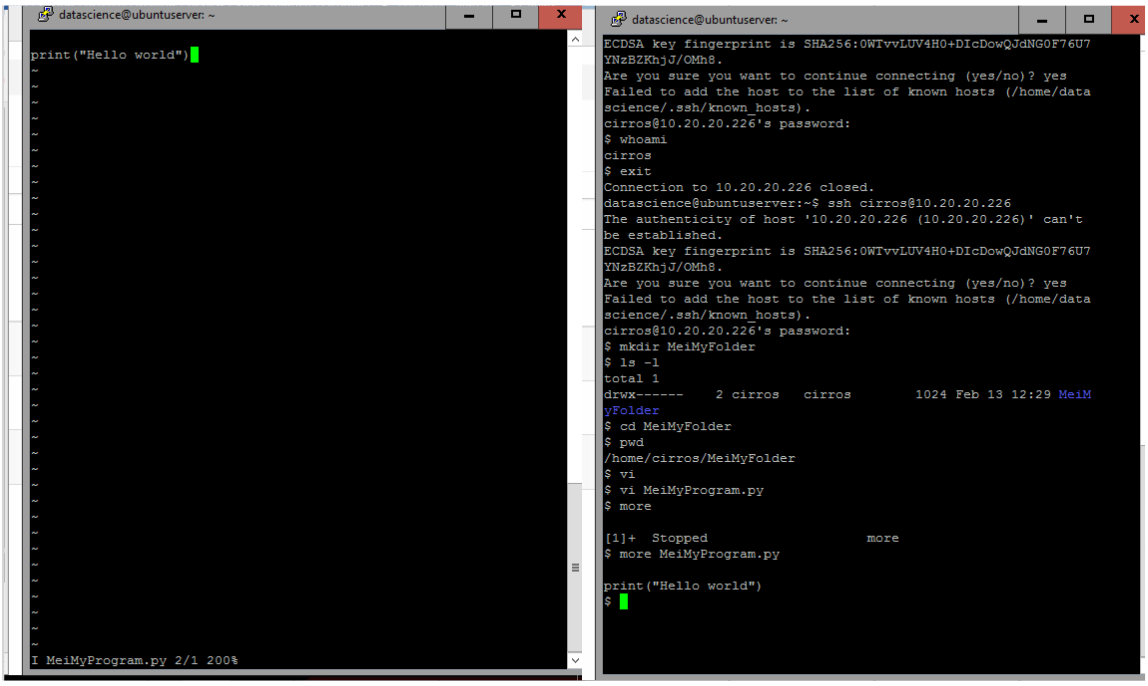

Figure.9.2

Task 10 – Copy of created file in Instance 1 to Inst2 and Inst3

As MeiInst1 and MeiInst2 are connected via the same private network of MeiPrivNetA, thus copy of file from MeiInst1 to MeiInt2 can be done via command of **scp** then **directory of folder in MeiInst1** and follow by **private IP address of MeiInst2** as per shown in Figure.10.1

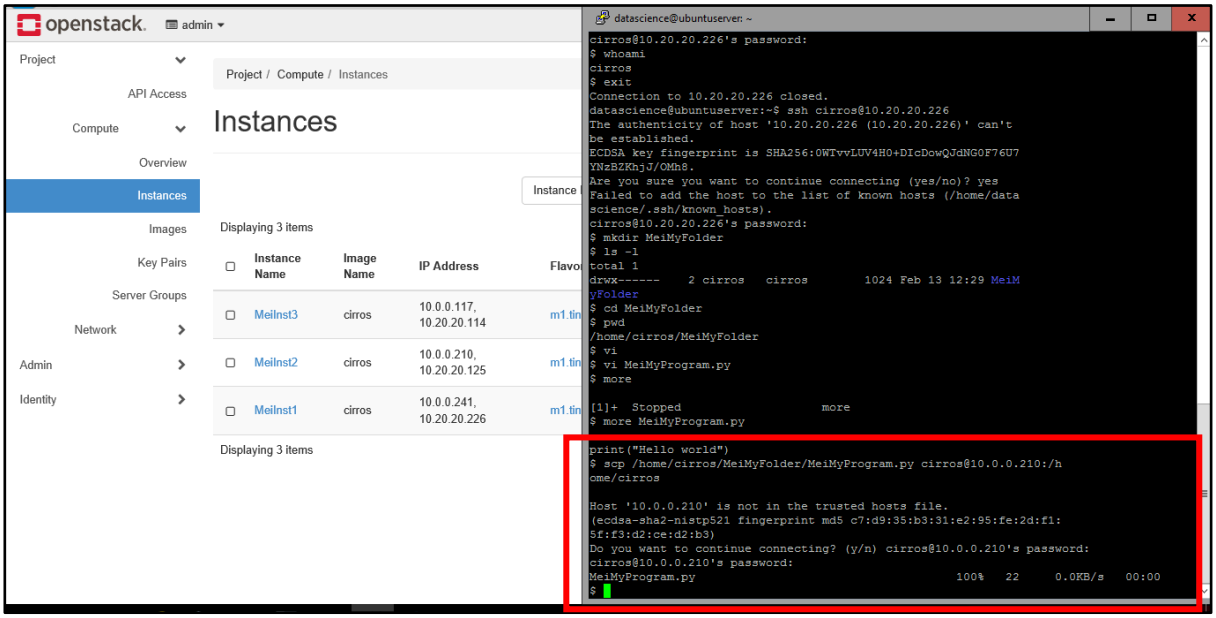

Figure.10.1

Whilts MeiInst3 was connected to another private network of MeiPrivNetB, it was not success to copy file from MeiInst1 directly to MeiInst3 via private IP address of 10.0.0.117. Thus it has to be done via the connecting of floating IP address of 10.20.20.114 which via the public network as shown in Figure.10.2. This is due to the private IP adress is likely to to be used to access the instance by other instances in private networks whilts floating IP address would be used to access the instance from public networks.

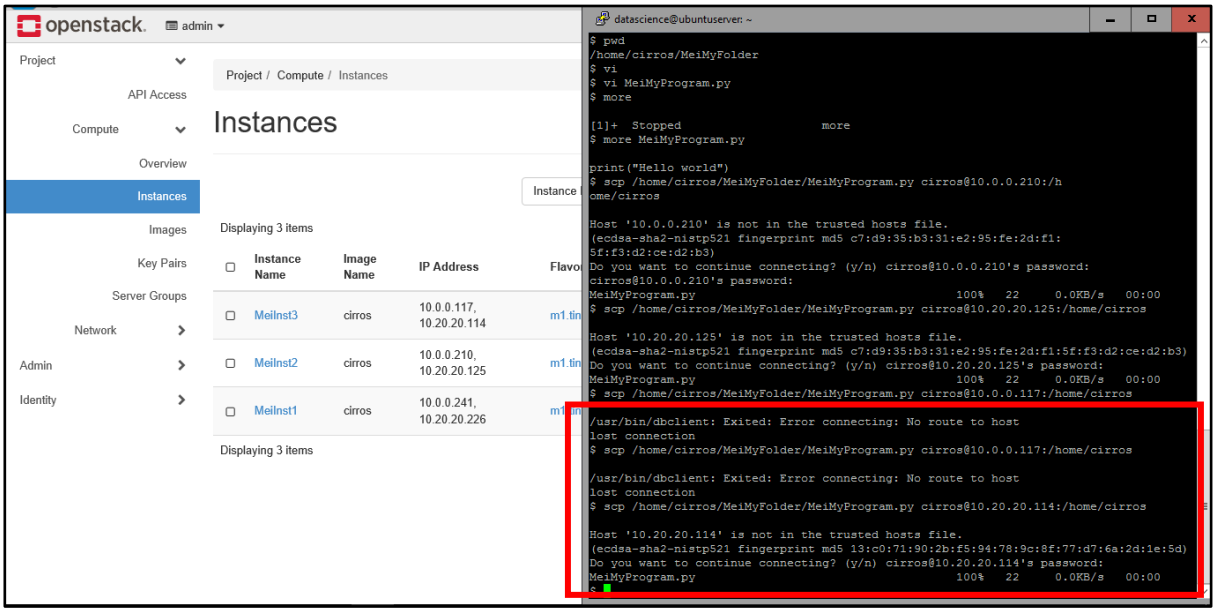

Figure.10.2