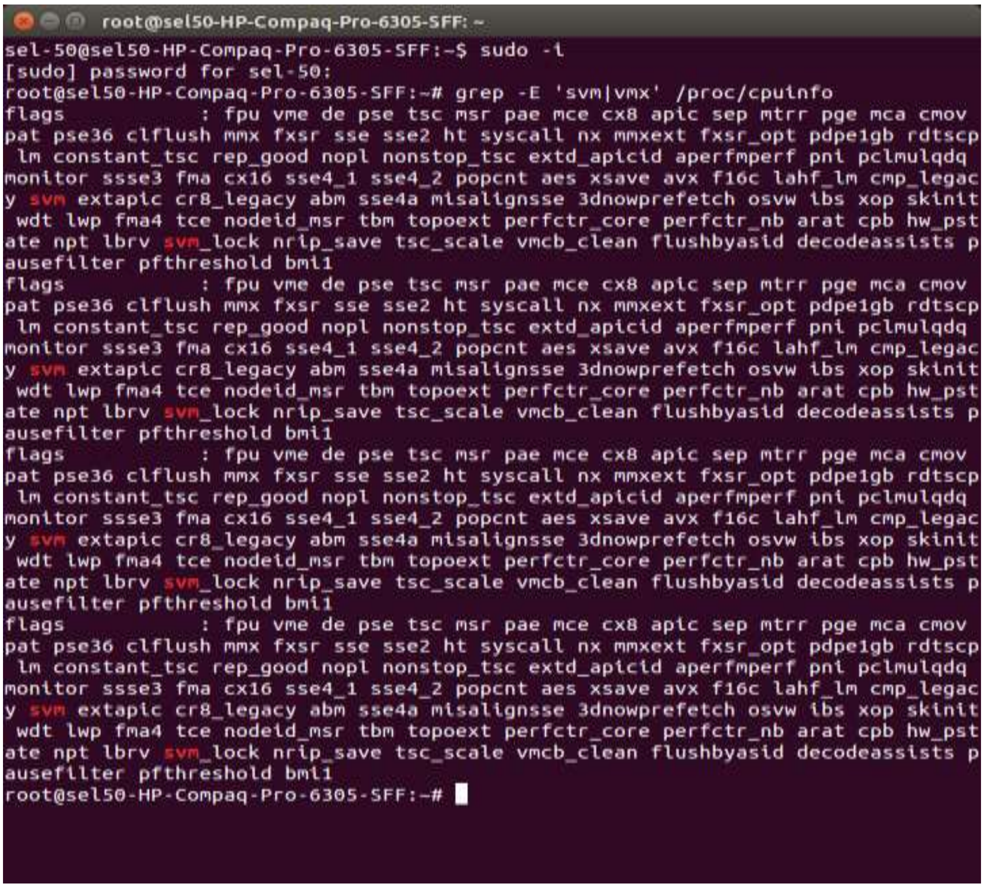
INSTALLATION OF OPENNEBULA

1. Switch the login into root user privilege mode by the command
   a. **sudo -i**
   b. When asked for password, give sel@123
2. Check the system whether it supports virtualisation or not by the command
   a. **grep –E ‘svm|vmx’ /proc/cpuinfo**

      Output must be like this



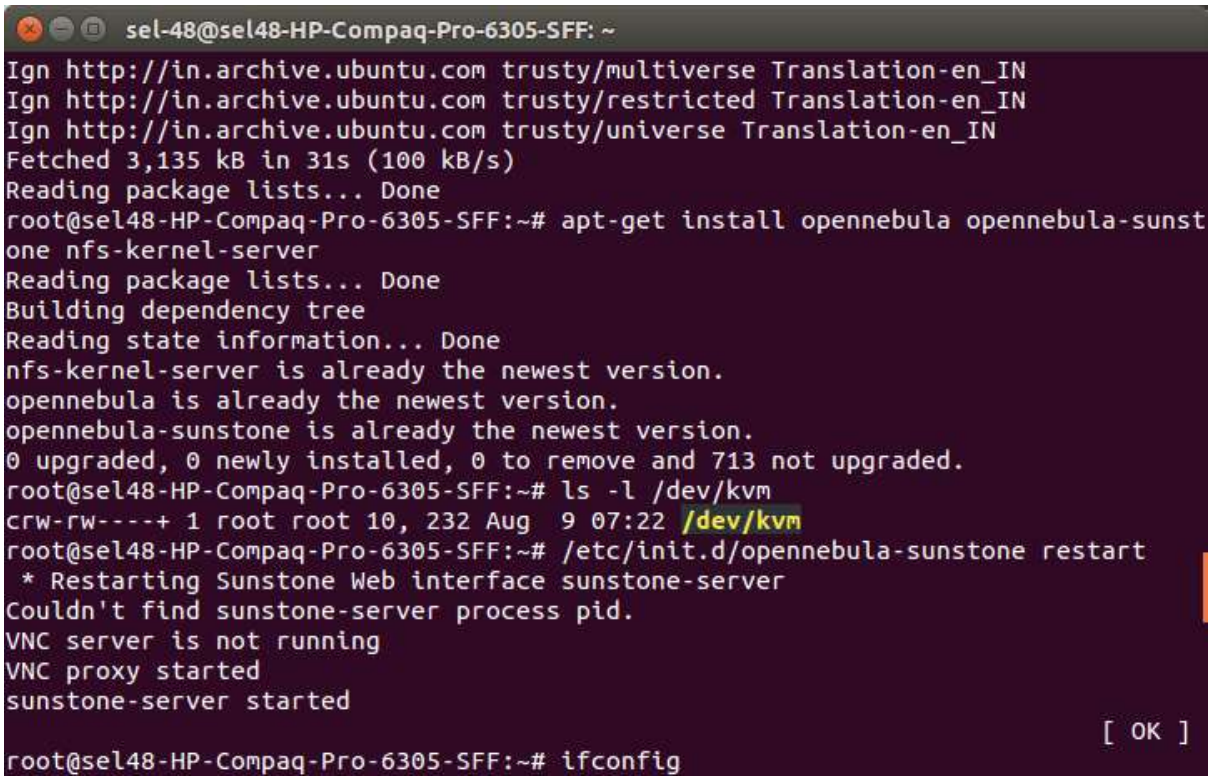
The red lettered svm proves that the system supports virtualisation.

Installation of the frontend:

3. **apt-get update** [sudo is not required as we are already in root mode]
4. The command to install the packages required for frontend are:

a. **apt-get install opennebula opennebula-sunstone nfs-kernel-server**

5. To ensure that the packages have been correctly
   a. The command is : **ls –l /dev/kvm**
6. Sunstone listens only in the loopback interface by default for security reasons. To change it **gedit /etc/one/sunstone-server.conf** and change :host: 127.0.0.1 to :host: 0.0.0.0.
7. Restart the sunstone server –
   /etc/init.d/opennebula-sunstone restart
   a. The output should be as follows:

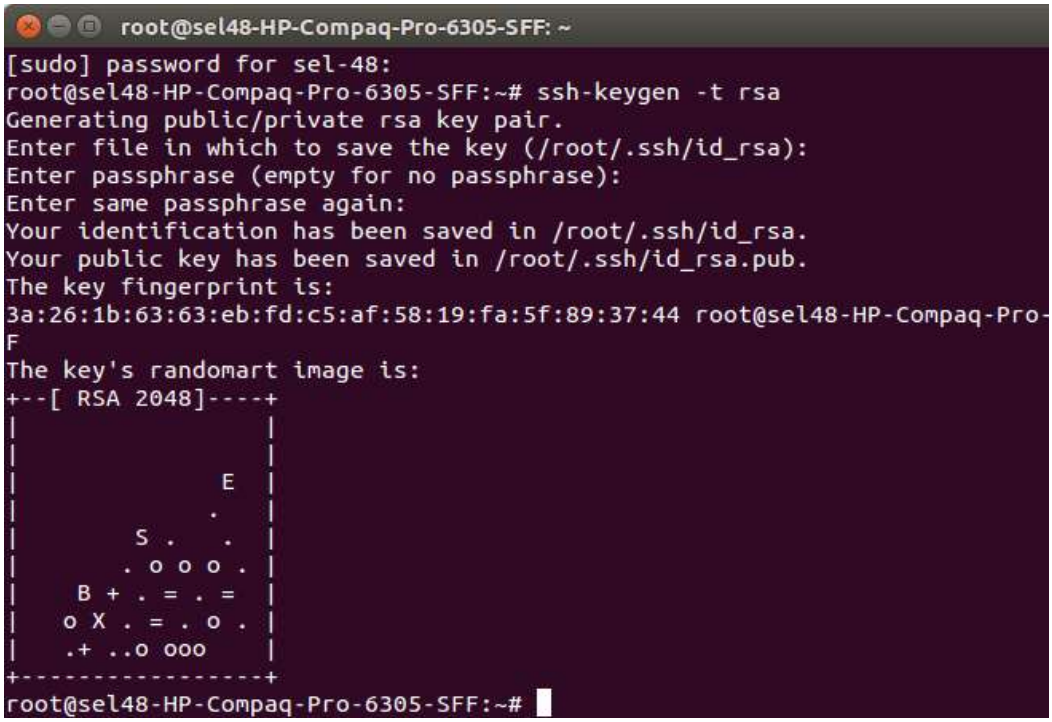


Generation of keys:

8. After ensuring that you are still in root mode, type the following command in the terminal. This command creates a hidden folder named .ssh in the root's home folder. In that a provate and public key will be created.
   //doubt whether "su –oneadmin" should be given or not. Please refer!
   ssh-keygen –t rsa
   Click "enter" for all subsequent queries the terminal asks.

At the end of this step, the following files should have been created in .ssh folder - **id_rsa and id_rsa.pub**

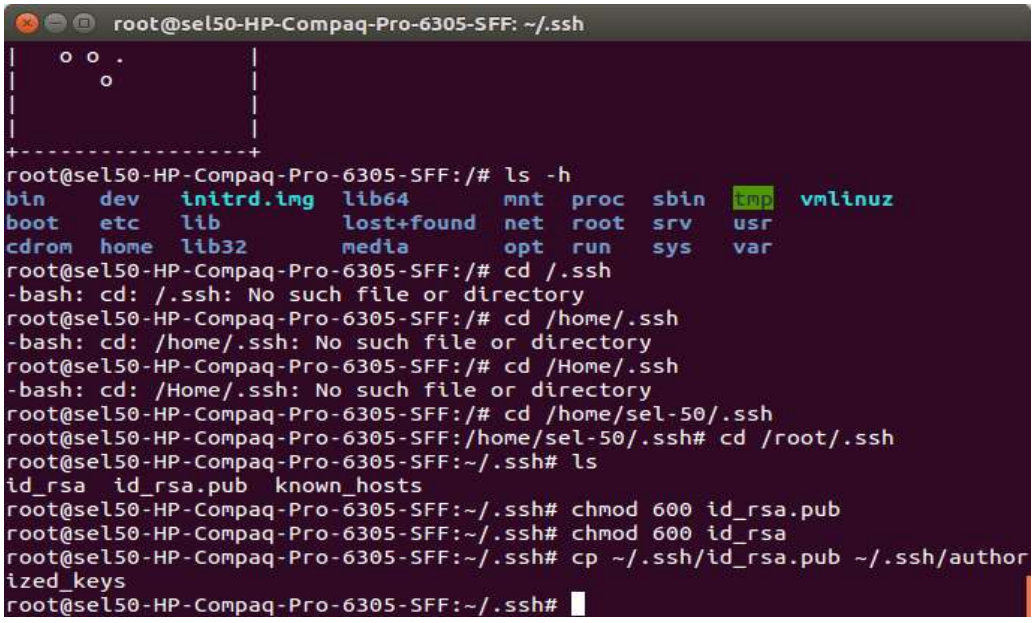
The output would like :



9. Now we must copy the public key into a file named “authorized_keys”. The same can be done as

cd /root/.ssh
chmod 600 id_rsa.pub
cp id_rsa.pub authorized_keys

The output should be:

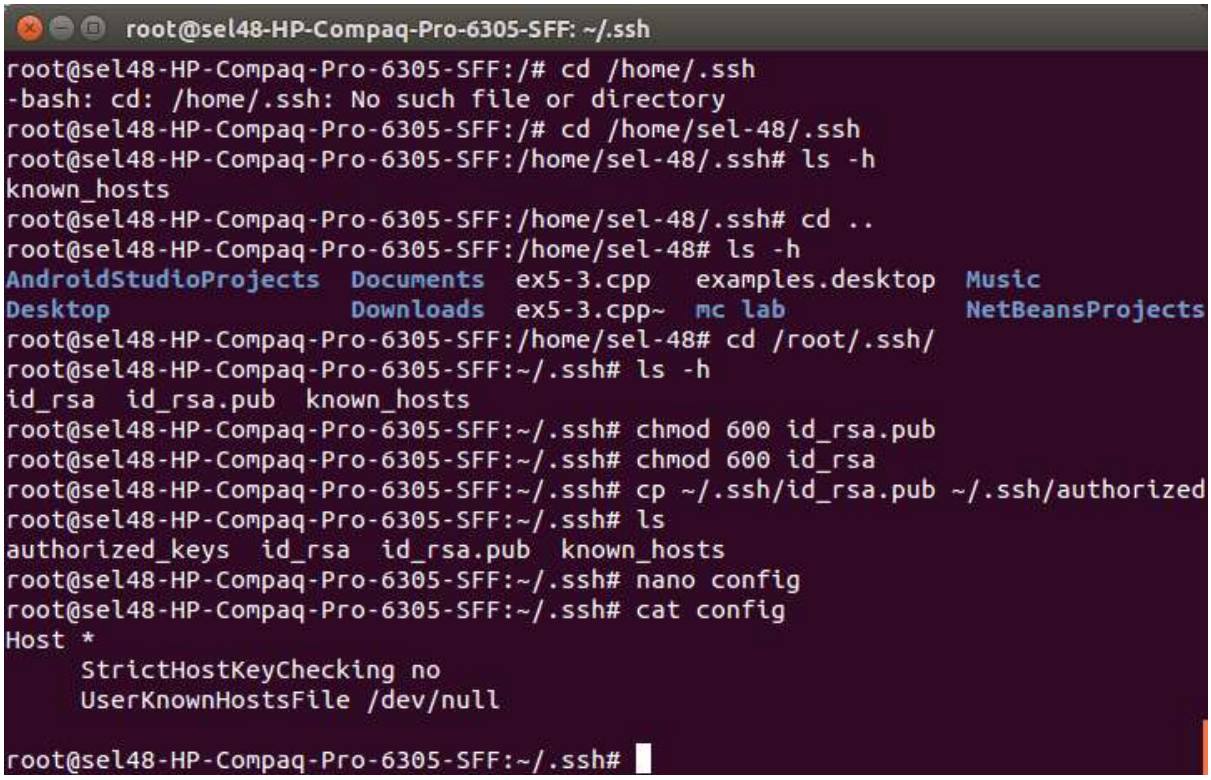


10. The restrictions to the public key should be listed in a file named "config". So to create a file and write contents into it,
    a. Give **gedit config.** Once gedit opens, type the following contents and save.
    b. For systems where gedit is not working, give **nano config**. Nano is another text editor , but within the terminal. Preserve the intendation in the following text as such.
        i. Host *

            StrictHostKey Checking no

            UserKnownHostsFile ./dev/null

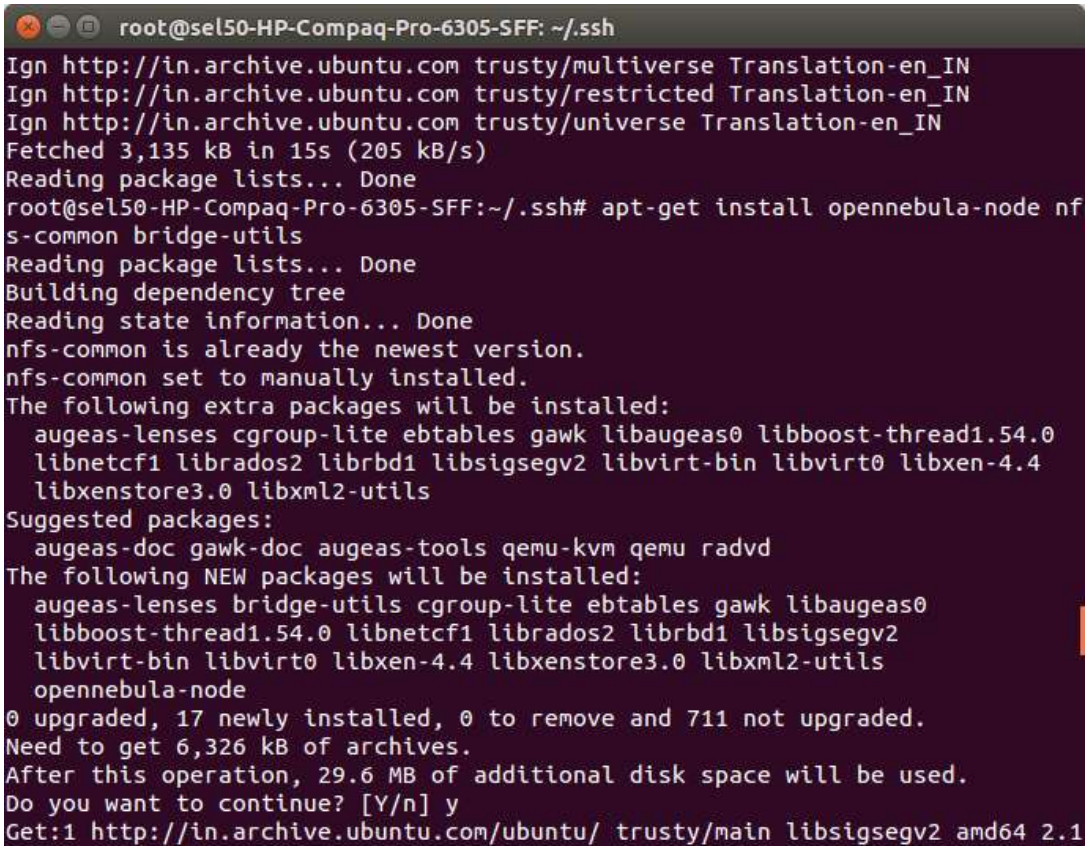
Give **Ctrl+O** to save in nano. It will ask you to confirm the name of the file being written. Click Enter. Give **Ctrl+X** to eturn to main terminal. The contents can be verified by giving **cat config.** The output is:



Installation at the nodes:

11. **apt-get update**
    apt-get install opennebula-node nfs-common bridge-utils

The output should be



Once it is successfully done, proceed to next step.

☆ By this time, go to browser and type **localhost:9869** in URL bar and see if you are getting the home page of OpenNebula.

Configuring the server interface at node

12. Go to interfaces.d folder by typing the command

cd /etc/network/interfaces.d

There create a file named **eth0.config**

The following lines are from the website intended to be writtenin this config file, but with changes.

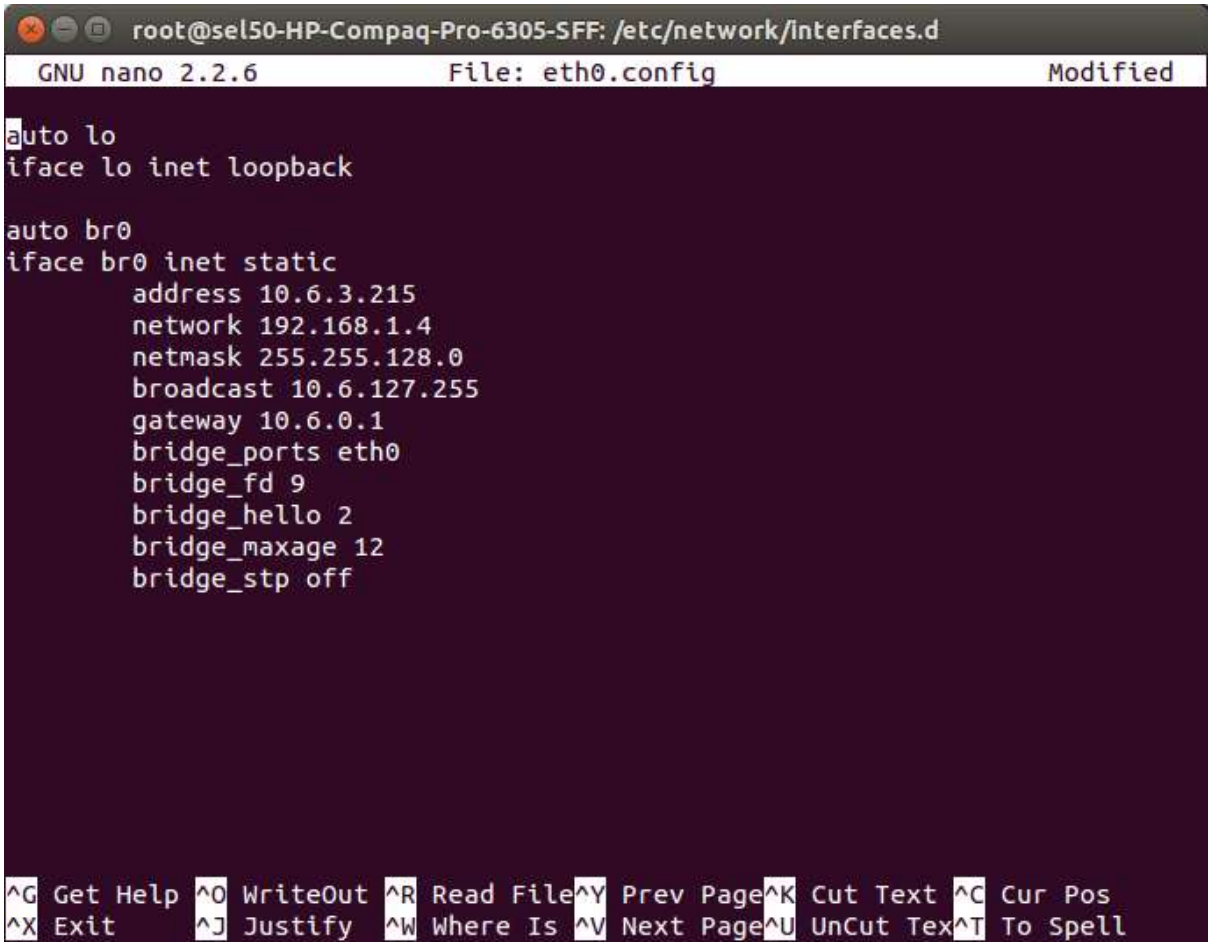
```
auto lo
iface lo inet loopback
```

auto br0
iface br0 inet static
address 192.168.0.10
network 192.168.0.0

netmask 255.255.255.0
broadcast 192.168.0.255
gateway 192.168.0.1
bridge_ports eth0
bridge_fd 9
bridge_hello 2
bridge_maxage 12
bridge_stp off

- ➔ Change address to our IP address.
- ➔ Network should be changed to 192.168.1.4
- ➔ Netmask, broadcast,gateway must be changed
- ➔ Note: To change the above information, the information is available in the terminal by typing **ifconfig**

The output will be:



13. After these changes, restart the network –

 # /etc/init.d/networking restart