MPI CLUSTER ON A LAN (MPICH2)

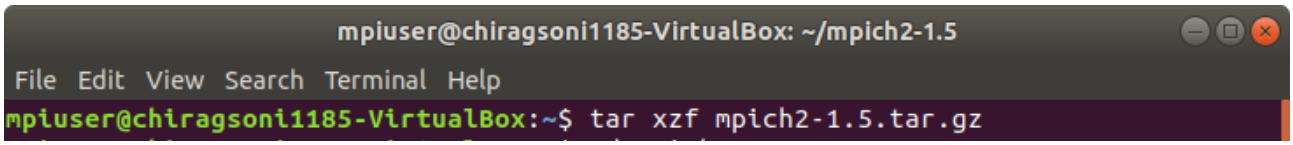
PREREQUISITE:
a) You should be able to **ping between your devices**(laptops/PC's).

b) If working on Virtual Machines, then select **Bridged Network Adapter** under network settings.

1)INSTALLING MPICH2:

a) Download the latest version of mpich2 from "mpich.org".

b) Unpack the tar file using the following command:



NOTE: If your tar doesn't accept the z option, use:

gunzip mpich2-1.5.tar.gz
tar xf mpich2-1.5.tar

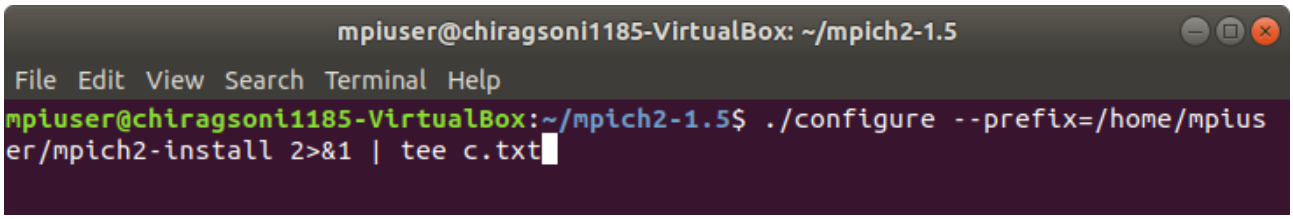
c) Move to the toplevel directory after unpacking the files:

d) Choose an installation directory, say /home/<USERNAME>/mpich2-install, which is assumed to non-existent or empty. It will be most convenient if this directory is shared by all of the machines where you

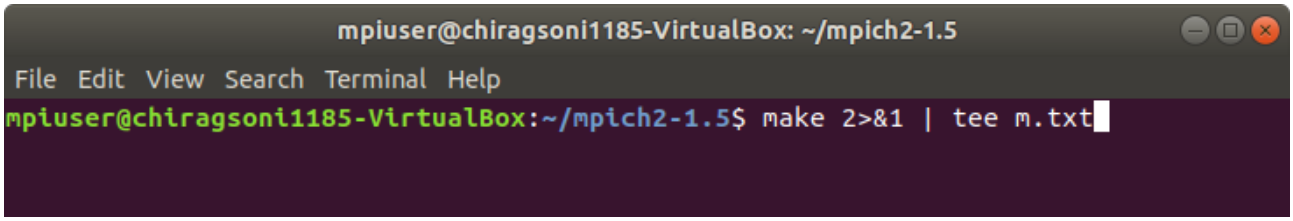
intend to run processes. If not, you will have to duplicate it on the other machines after installation.

e) Configure the MPICH2 specifying the installation directory:

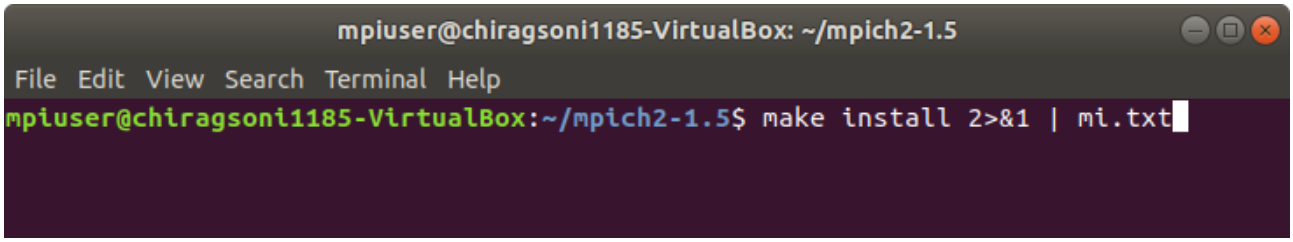
(**NOTE:** If "FORTRAN" is not installed in your machine then either install it or diable it in this step.)



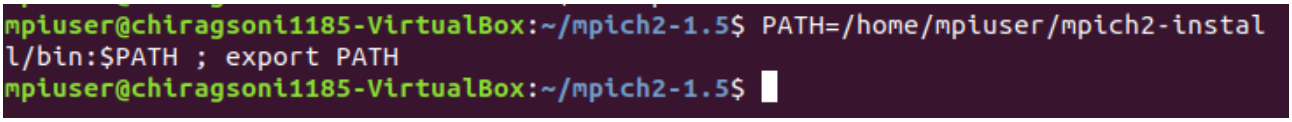
f) Build MPICH2:



g) Install the MPICH2 commands:

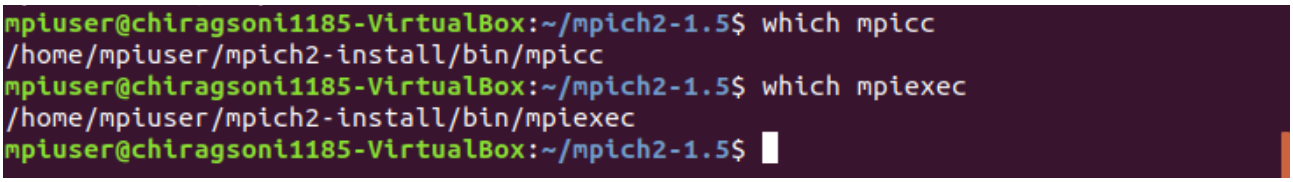


h) Add the bin subdirectory of the installation directory to your path in your startup script:

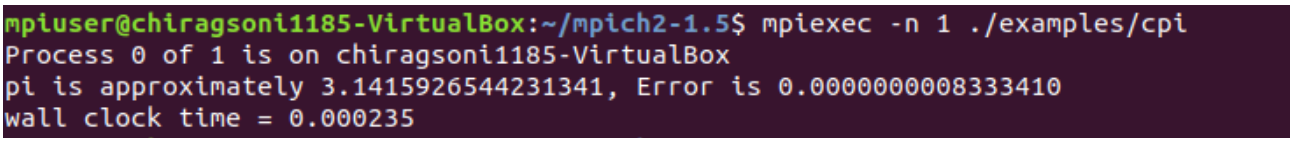


IMPORTANT NOTE: The install directory has to be visible at exactly the same path on all machines you want to run your applications on. This is typically achieved by installing MPICH2 on a shared NFS file-system. If you do not have a shared NFS directory, you will need to manually copy the install directory to all machines at exactly the same location.

i) At this point we check whether everything is in order by executing the following commands:

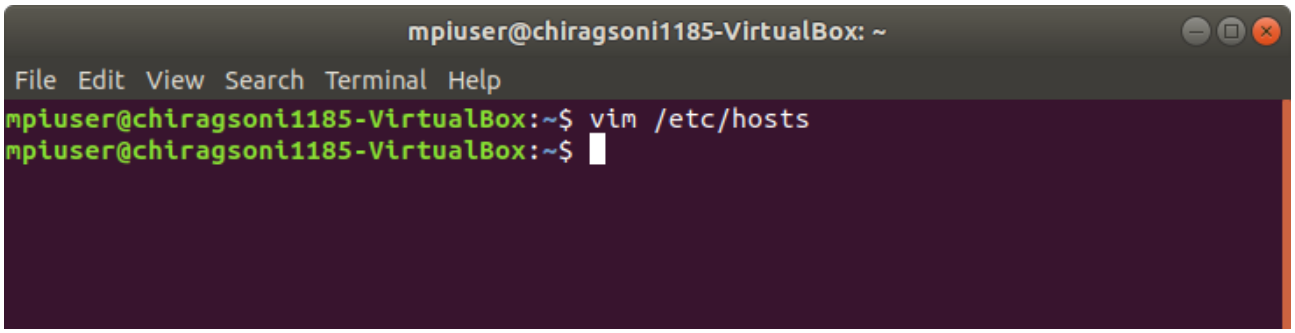


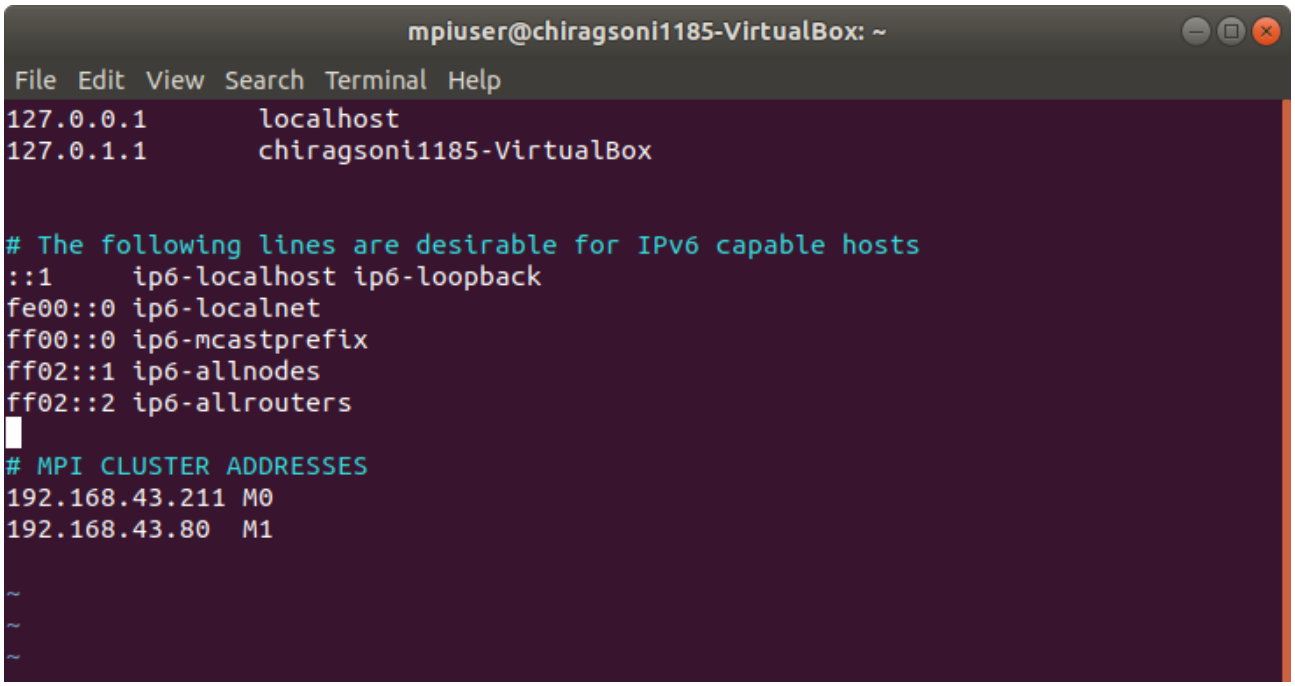
ii) Executing the example program provided in the mpich2 package:



2) EDIT THE "/etc/hosts" FILE:

Edit the above mentioned file so that there in no need to remember the ip-addresses of every client every time. We can assign aliases to the ip-addersses so that we can refer the client using the aliases.

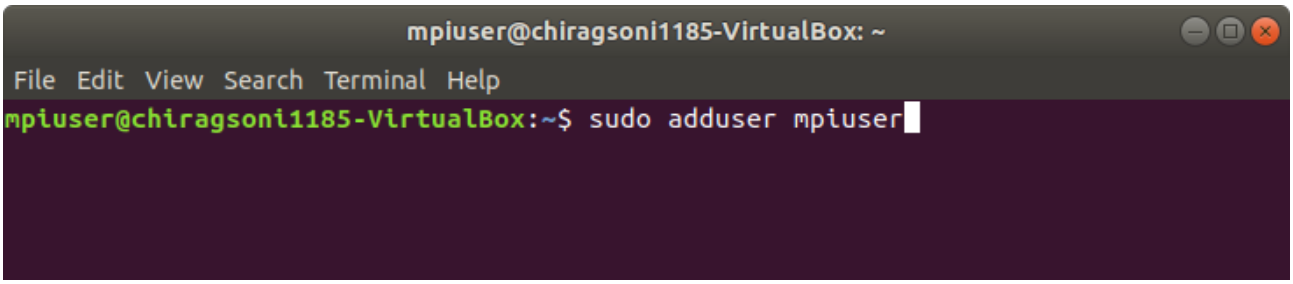




Here M0,M1 are aliases for the master and client machines respectively.

3) ADD ANOTHER USER

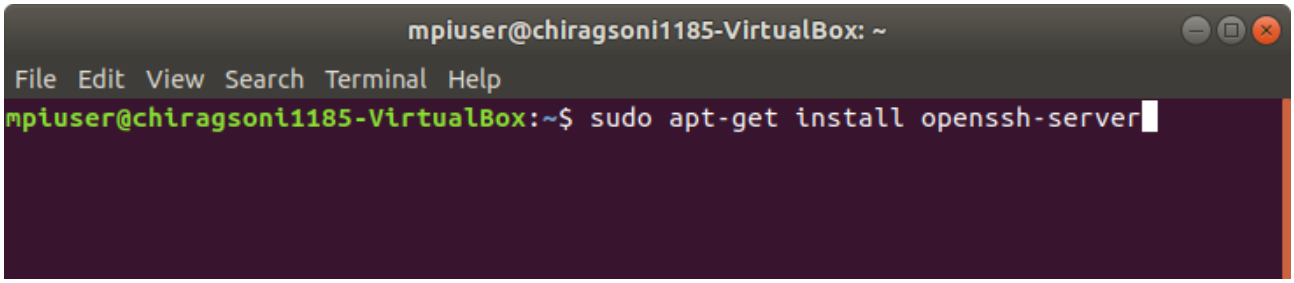
We add another user , here named “mpiuser” , such that we can have a common user in all the the nodes to keep things simple.



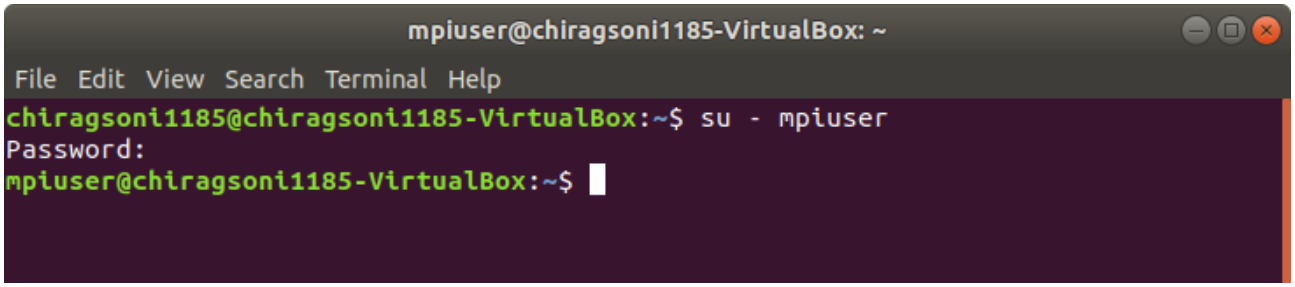
PS: Do not use “useradd” instead of “adduser” as it does not create a seperate home for different users.

4) SETTING UP SSH:

The machines are going to talk over the ssh.



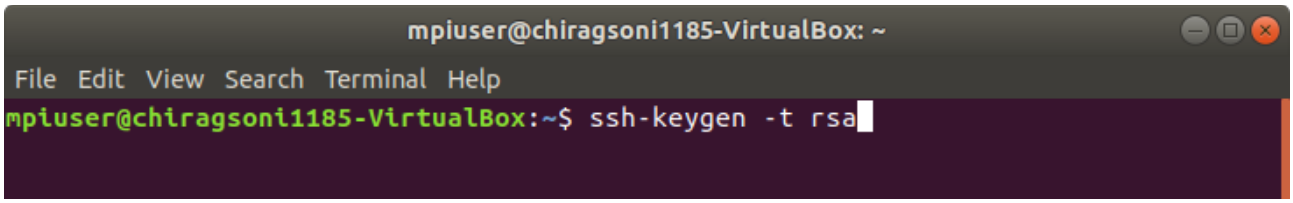
5) SWITCH TO THE NEWLY CREATED USER



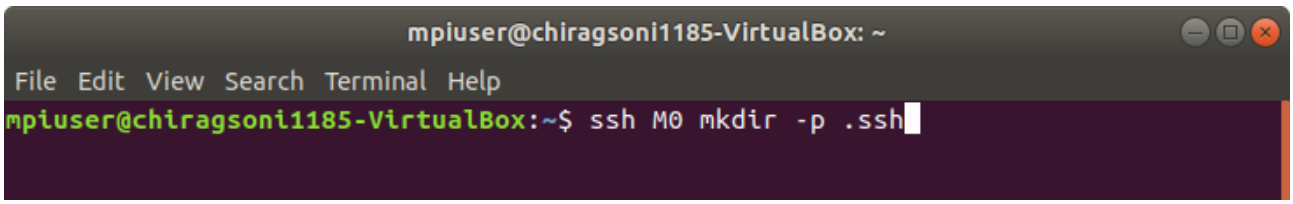
Since the ssh server is already installed, you must be able to login to other machines by ssh **username@hostname**, at which you will be prompted to enter the password of the username. To enable more easier login, we generate keys and copy them to other machines' list of authorized_keys.

6) PASSWORDLESS LOGIN:

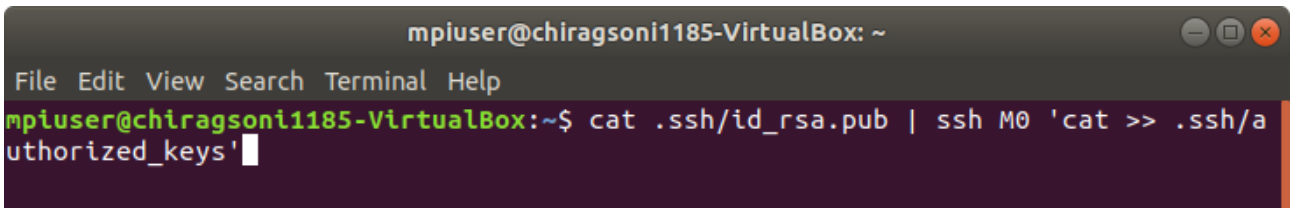
a) Generate a pair of authentication rsa keys using the ssh:



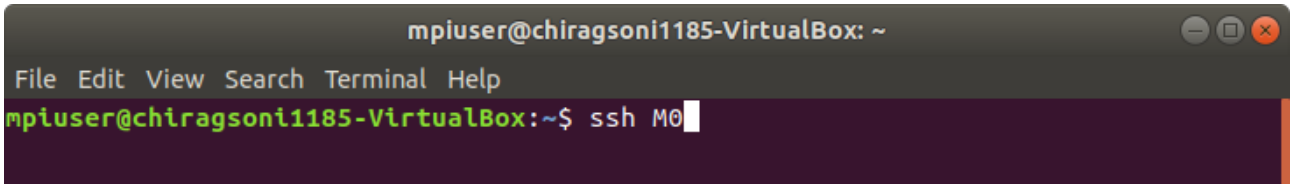
b) Use ssh to create a directory ~/.ssh as user on any Client(here M0) .
(**Note:** The directory may already exist, which is fine):



c) Append Master's new public key to user@Client:.ssh/authorized_keys and enter users's password one last time:



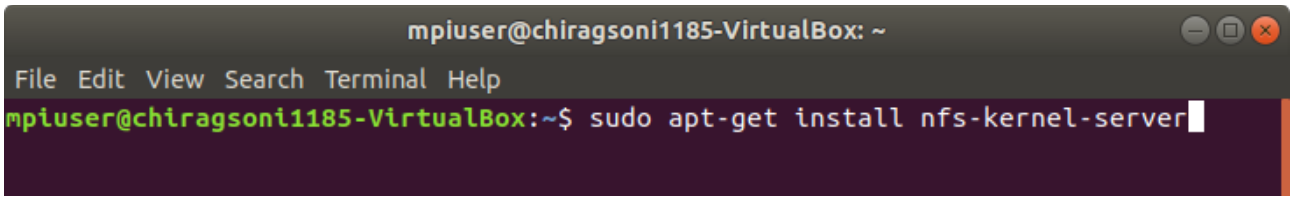
d) The next login to the client will be passwordless.



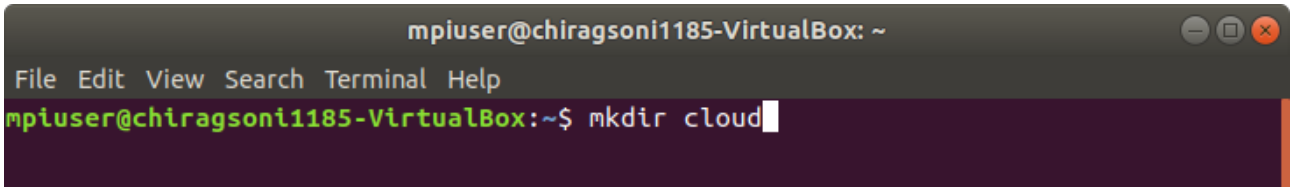
7) INSTALLING NFS-SERVER:

You share a directory via NFS in **master** which the **client** mounts to exchange data.

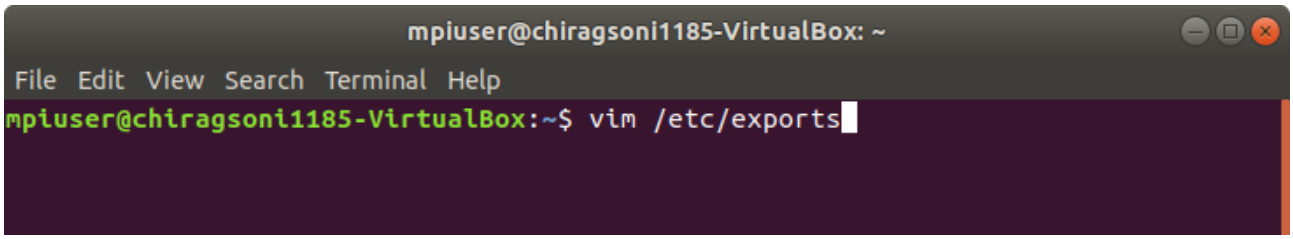
a) Insrtall the required packages:

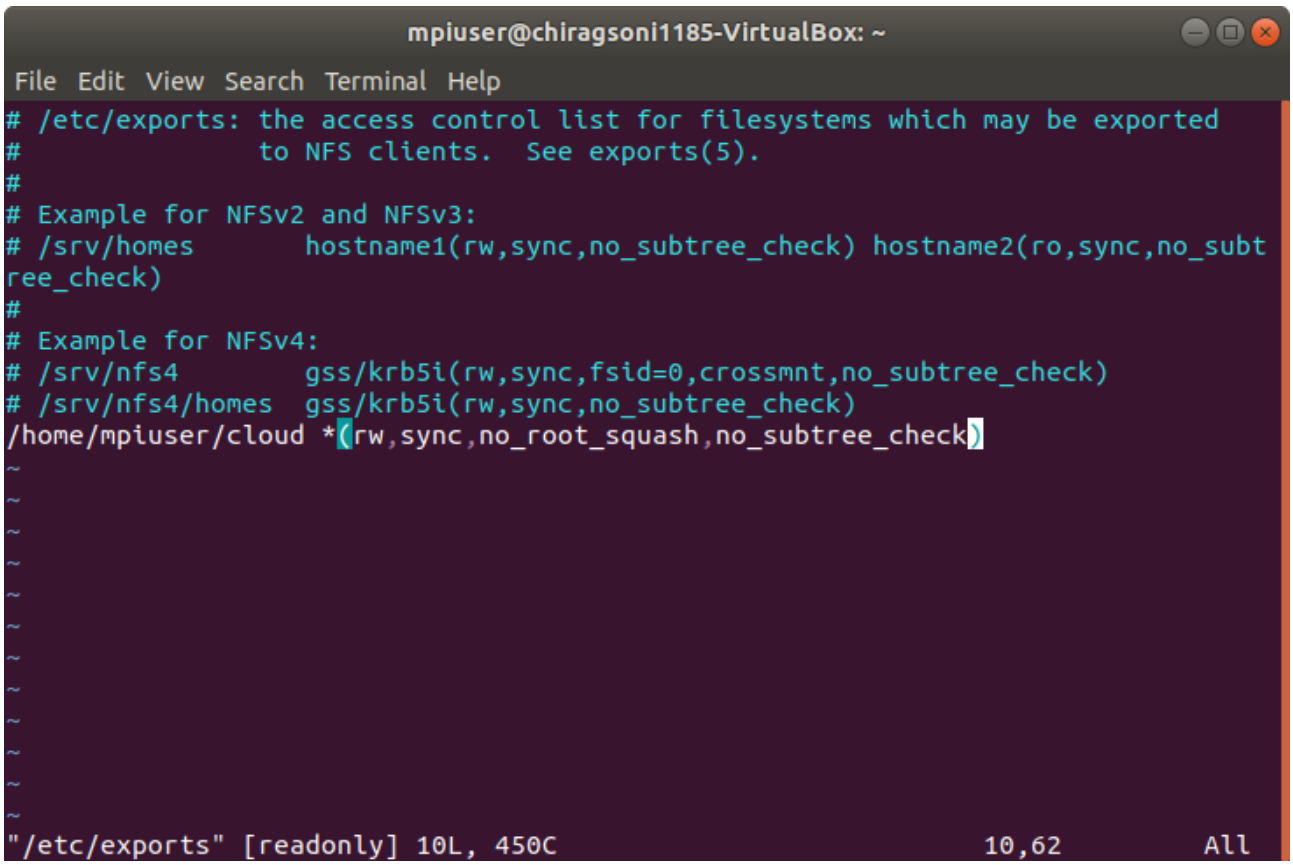


b) Create a common folder(here cloud) which we will share across the network.



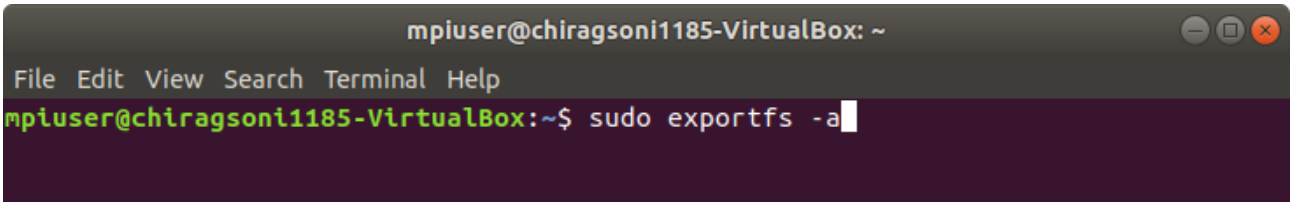
c) To export the cloud directory, we will have to create an entry in the /etc/exports file.



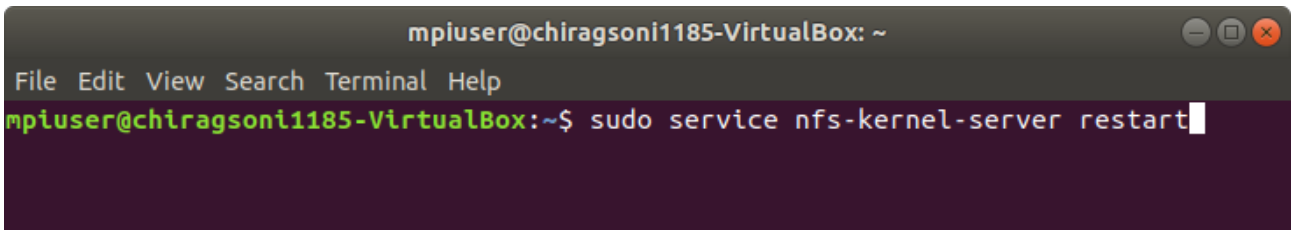


PS: Here instead of * in the entry in the file you can add specific ip addresses

d) After making the entry in the /etc/exports file, we need o export the file system using **exportfs -a**

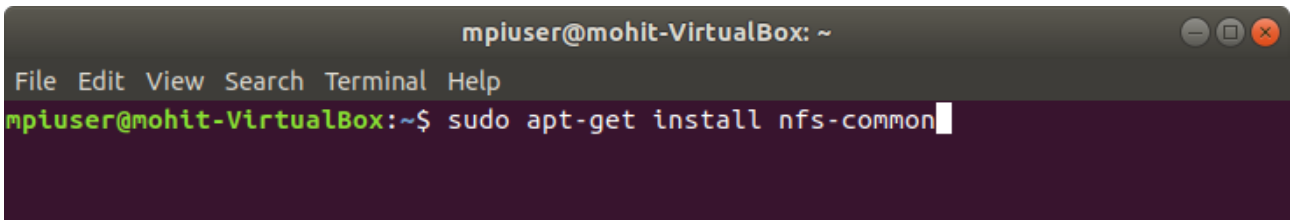


e) After exporting the file system , restart the nfs server.

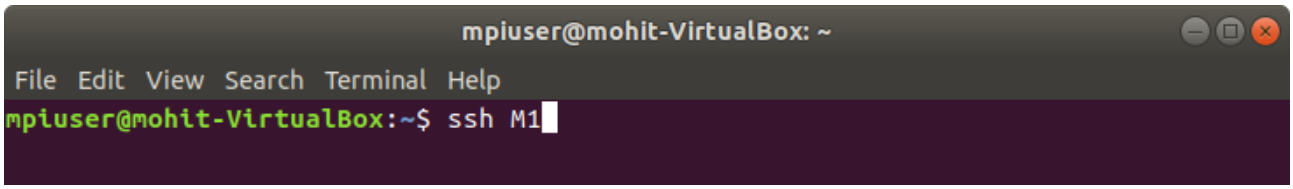


8) NFS-CLIENT:

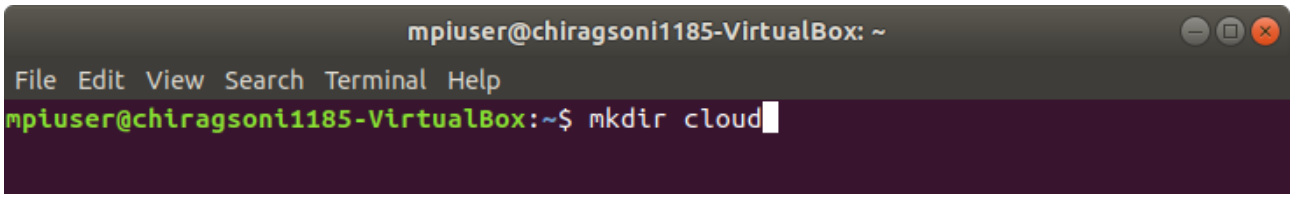
a) Install the required packages:



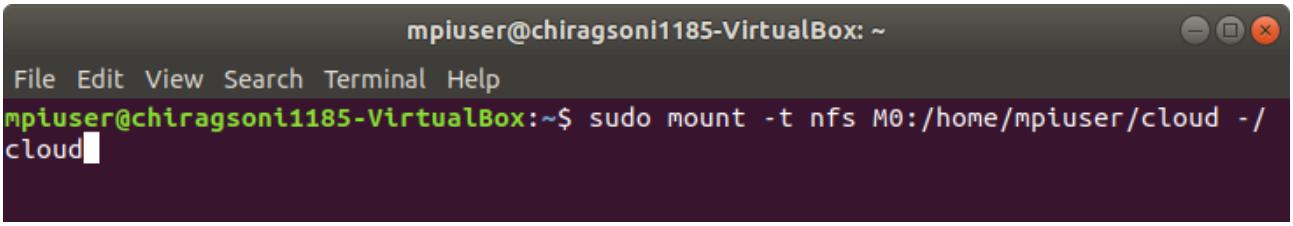
b) Now, we need to create a directory with the same name as the directory created in the previous step in the client's machine. Therefore we first login to the client's system.



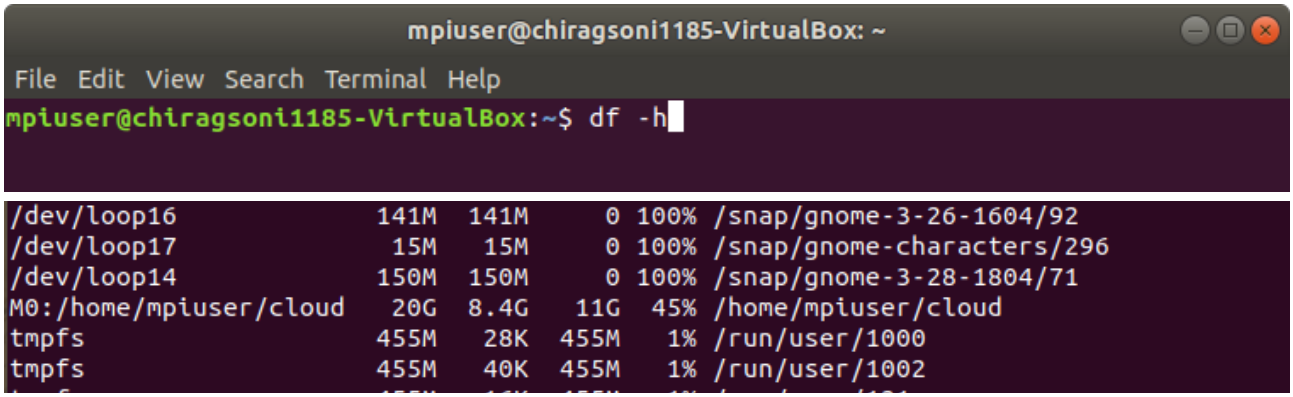
c) Create the directory:



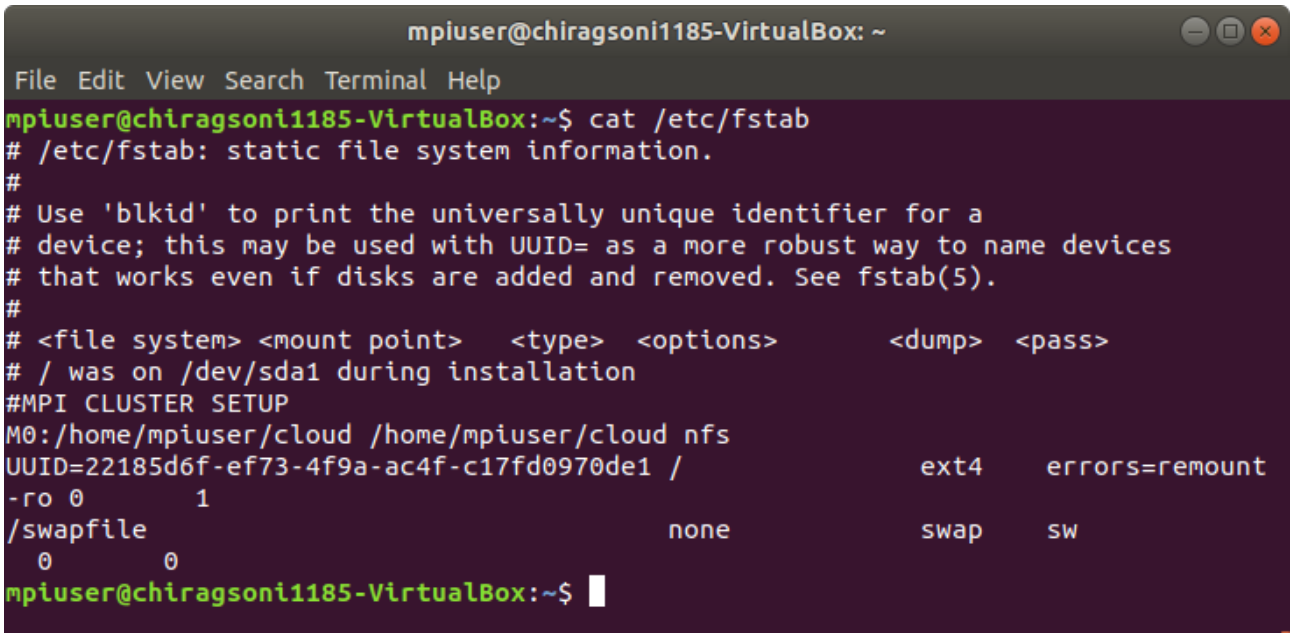
d) Mounting the shared directory:



e) Checking the mounted directories:



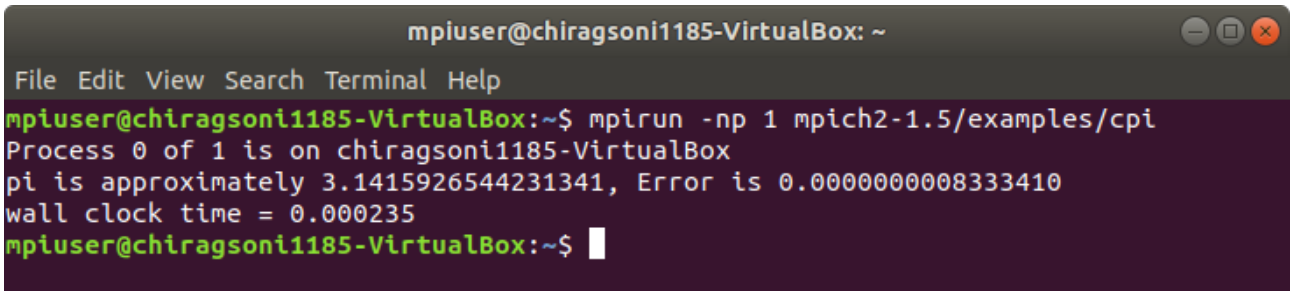
f) To make the mount permanent so you don't have to manually mount the shared directory everytime you do a system reboot, you can create an entry in your file systems table - i.e., /etc/fstab file like this:



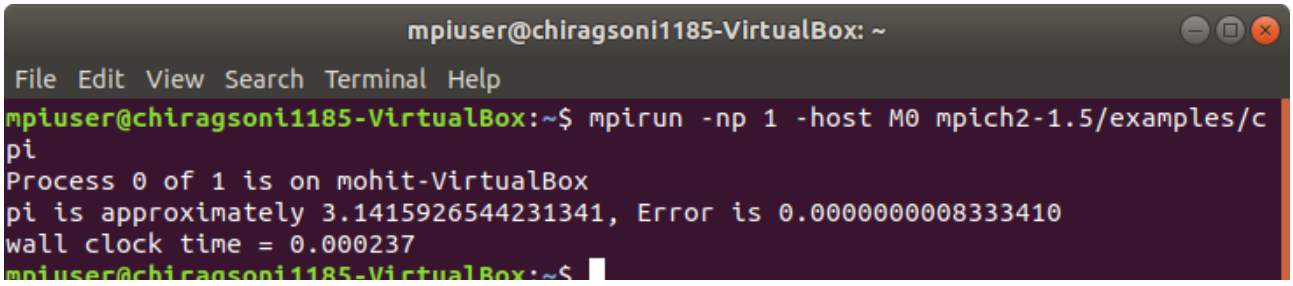
9) RUNNING THE MPI PROGRAMS:

We will just run the sample program provided in the MPICH2 installation for now.

a) Running it on your own machine:



b) Running it on the master's machine:



References:

https://mpitutorial.com/tutorials/running-an-mpi-cluster-within-a-lan/
https://www.mpich.org/
https://www.linode.com/docs/security/authentication/use-public-key-authentication-with-ssh/