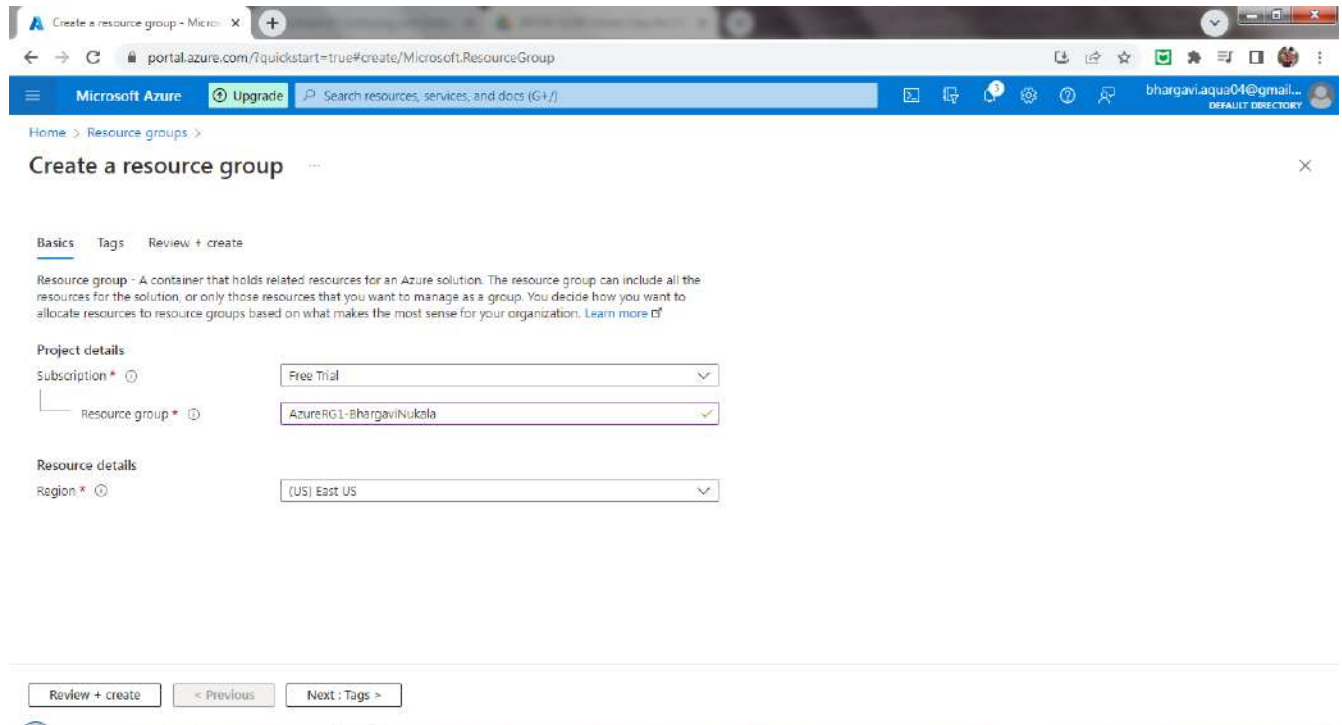
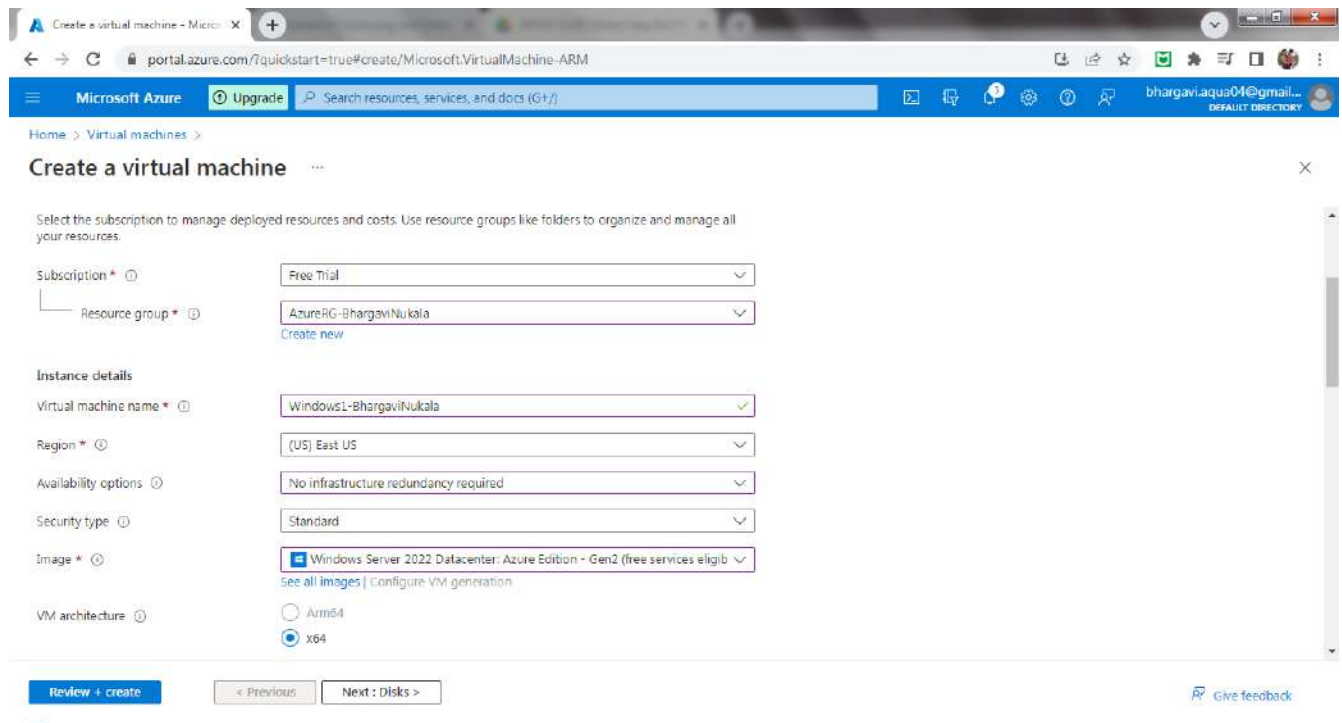


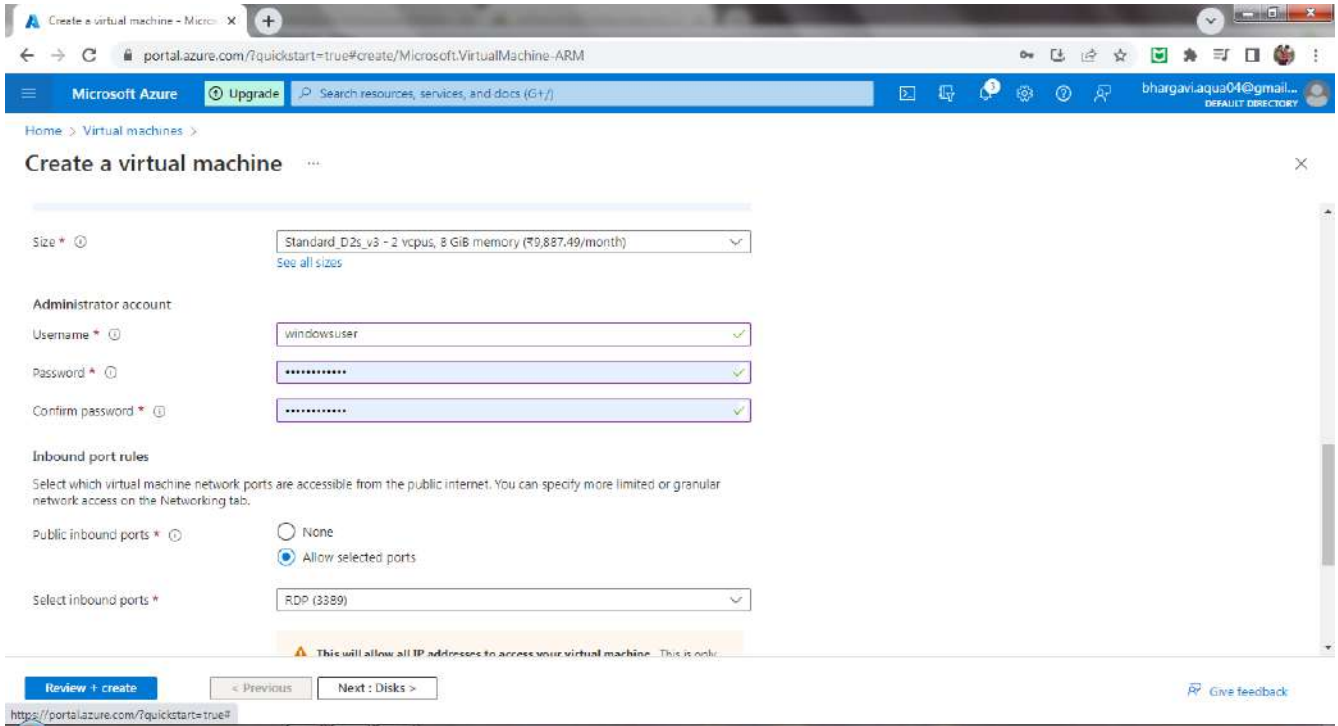
Resource Group: Create a resource group to associate for all resources to be created in Azure



Create a Windows Virtual Machine under above created the ResourceGroup.

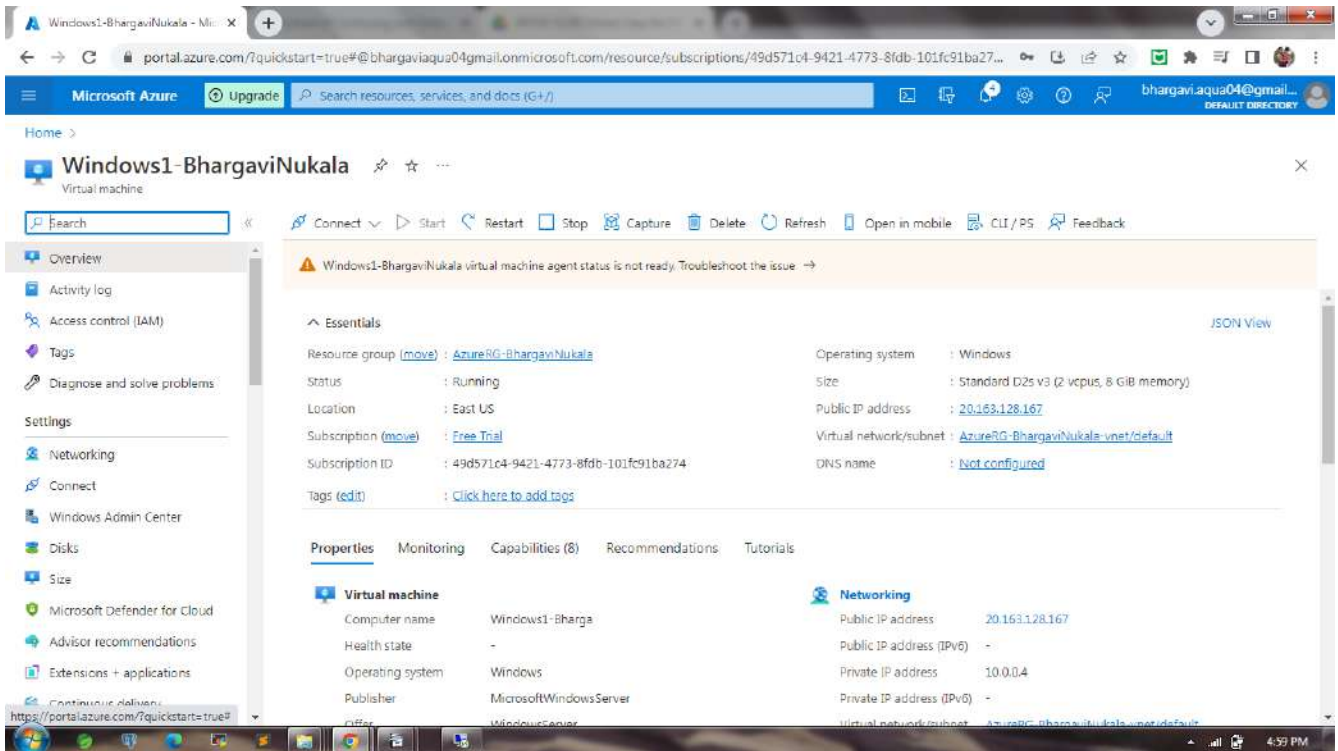


We should mention the username and password to connect to VM while creation only.



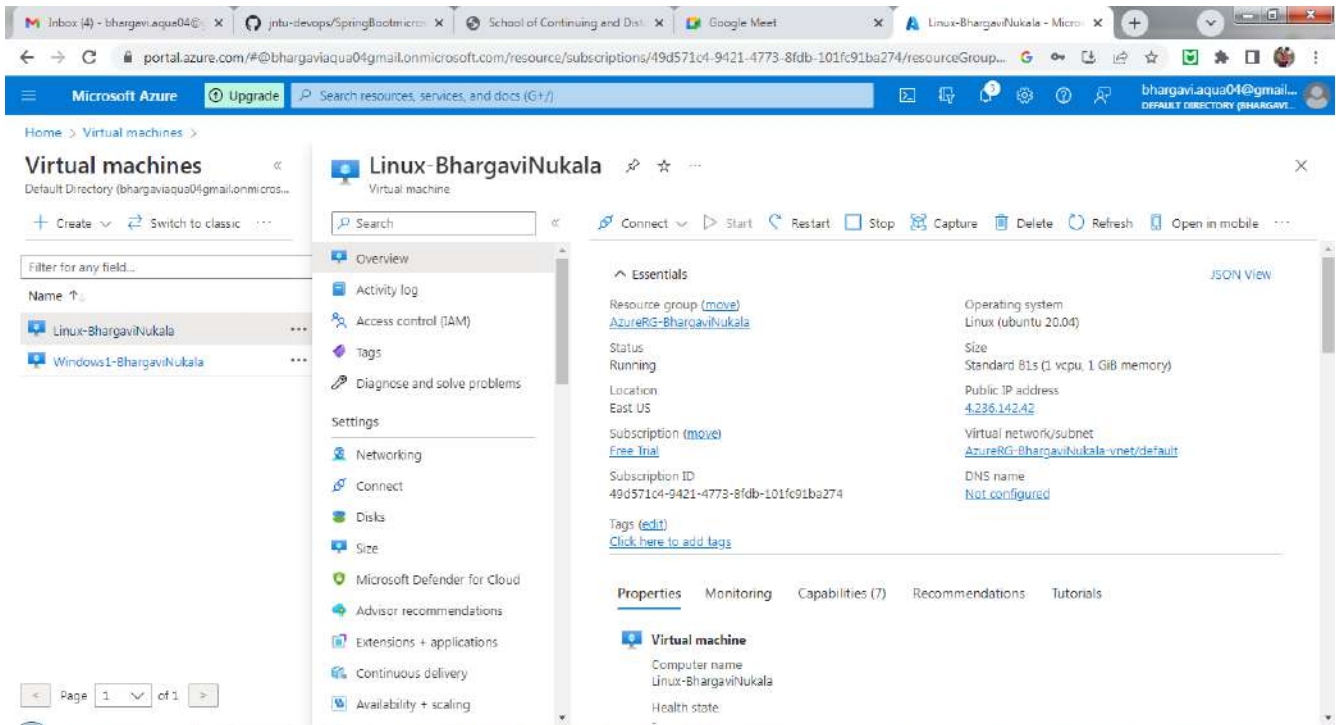
In Networking tab, check the “Delete IP address when VM is deleted” check box and see that Public IP is set.

Virtual Machine is created.



We connect to Windows machine using Remote Desktop Connection.

Creation of a Linux Virtual Machine

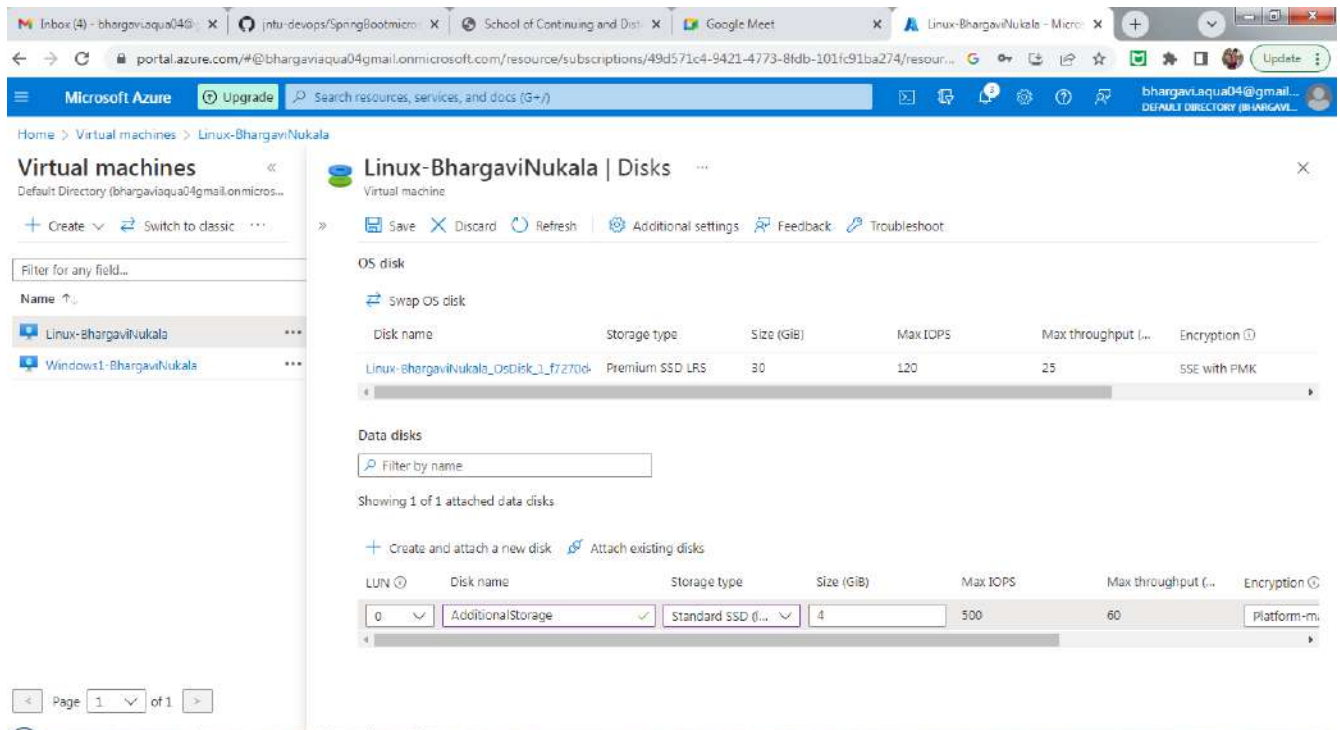


Connect to Linux machine using “ssh username@publicIP”

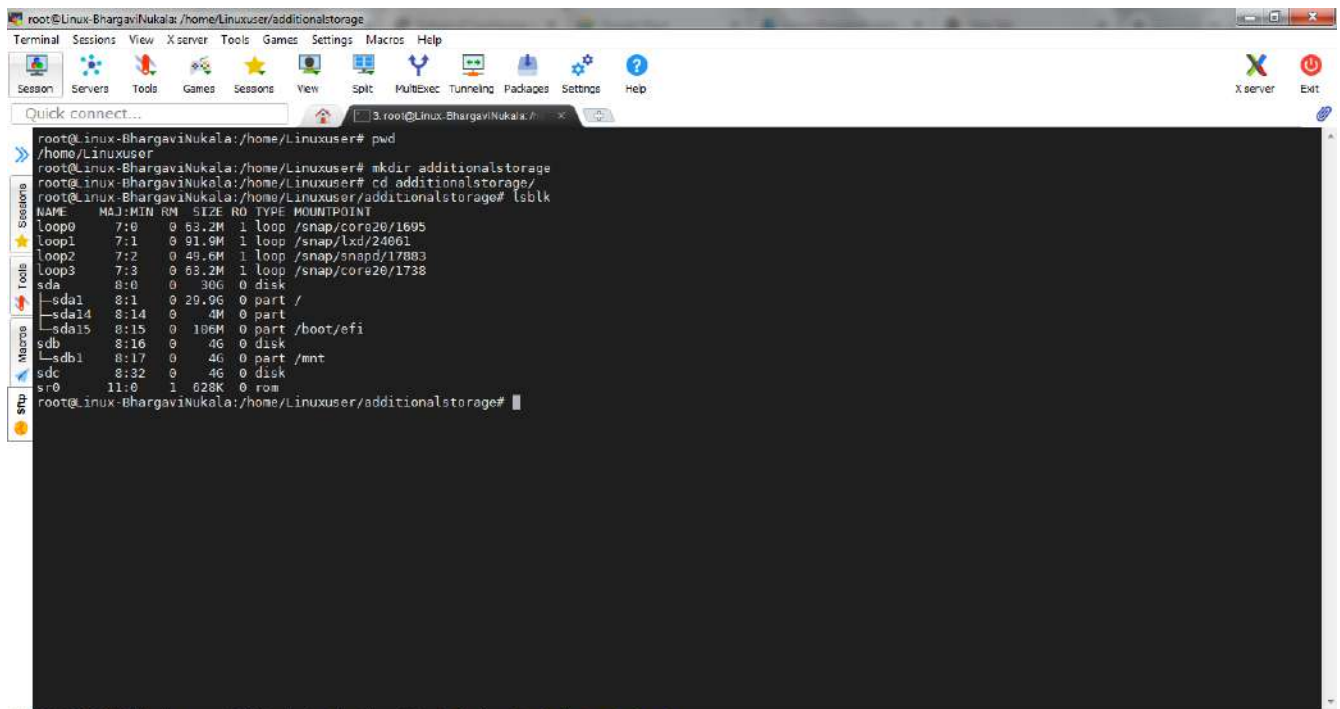


Creating extra storage disks and mount a filesystem

In VM, click on Disks, and create a additional disk. Attach it to the VM and save.



Additional Storage “sdc” available on Linux VM and shows not mounted



Now mount the storage using commands
\$mkdir additionalstorage
\$mkfs .ext4 /dev/sdc
\$mount -t ext4/dev/sdc additionalstorage/

```

root@Linux-BhargaviNukala: /home/Linuxuser
Terminal Sessions View Xserver Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultExec Tunneling Packages Settings Help
Quick connect...
loop2 7:2 0 49.6M 1 loop /snap/snapd/17883
loop3 7:3 0 63.2M 1 loop /snap/core20/1738
sda 8:0 0 30G 0 disk
├─sda1 8:1 0 29.9G 0 part /
├─sda14 8:14 0 4M 0 part
├─sda15 8:15 0 100M 0 part /boot/efi
├─sdb 8:16 0 4G 0 disk
├─sdb1 8:17 0 4G 0 part /mnt
├─sdc 8:32 0 4G 0 disk
└─sr0 11:0 1 628K 0 rom
root@Linux-BhargaviNukala: /home/Linuxuser# mkfs.ext4 /dev/sdc
mkfs2fs 1.45.5 (07-Jan-2020)
Discarding device blocks: done
Creating filesystem with 1048576 4k blocks and 262144 inodes
Filesystem UUID: 3866ac12-4ee5-4b3d-9f70-93e180405373
Superblock backups stored on blocks:
32768, 98304, 163840, 229376, 294912, 819200, 884736
Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
root@Linux-BhargaviNukala: /home/Linuxuser# mount -t ext4 /dev/sdc additionalstorage/
root@Linux-BhargaviNukala: /home/Linuxuser# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
loop0 7:0 0 63.2M 1 loop /snap/core20/1695
loop1 7:1 0 91.9M 1 loop /snap/lxd/24061
loop2 7:2 0 49.6M 1 loop /snap/snapd/17883
loop3 7:3 0 63.2M 1 loop /snap/core20/1738
sda 8:0 0 30G 0 disk
├─sda1 8:1 0 29.9G 0 part /
├─sda14 8:14 0 4M 0 part
├─sda15 8:15 0 100M 0 part /boot/efi
├─sdb 8:16 0 4G 0 disk
├─sdb1 8:17 0 4G 0 part /mnt
├─sdc 8:32 0 4G 0 disk /home/Linuxuser/additionalstorage
└─sr0 11:0 1 628K 0 rom
root@Linux-BhargaviNukala: /home/Linuxuser#

```

Create 10 files in the disk using the command
\$touch {1..10}.txt

```

root@Linux-BhargaviNukala: /home/Linuxuser
Terminal Sessions View Xserver Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultExec Tunneling Packages Settings Help
Quick connect...
sdb 8:16 0 4G 0 disk
├─sdb1 8:17 0 4G 0 part /mnt
├─sdc 8:32 0 4G 0 disk /home/Linuxuser/additionalstorage
└─sr0 11:0 1 628K 0 rom
root@Linux-BhargaviNukala: /home/Linuxuser# df -hTp
df: invalid option -- 'p'
Try 'df --help' for more information.
root@Linux-BhargaviNukala: /home/Linuxuser# df -hTp
Filesystem Type Size Used Avail Use% Mounted on
/dev/root ext4 29G 1.8G 28G 7% /
/devtmpfs devtmpfs 489M 0 489M 0% /dev
tmpfs tmpfs 454M 0 454M 0% /dev/shm
tmpfs tmpfs 91M 984K 90M 2% /run
tmpfs tmpfs 5.0M 0 5.0M 0% /run/lock
tmpfs tmpfs 454M 0 454M 0% /sys/fs/cgroup
/dev/loop8 squashfs 64M 64M 0 100% /snap/core20/1695
/dev/loop1 squashfs 92M 92M 0 100% /snap/lxd/24061
/dev/loop2 squashfs 50M 50M 0 100% /snap/snapd/17883
/dev/sda15 vfat 105M 5.2M 100M 5% /boot/efi
/dev/sdb1 ext4 3.9G 28K 3.7G 1% /mnt
/dev/loop3 squashfs 64M 64M 0 100% /snap/core20/1738
tmpfs tmpfs 91M 0 91M 0% /run/user/1800
/dev/sdc ext4 3.9G 24K 3.7G 1% /home/Linuxuser/additionalstorage
root@Linux-BhargaviNukala: /home/Linuxuser# cd additionalstorage/
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage# touch {1..10}.txt
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage# ls
1.txt 10.txt 2.txt 3.txt 4.txt 5.txt 6.txt 7.txt 8.txt 9.txt lost+found
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage# vi "hello world!" > test.txt
vi "hello world!" > test.txt
Vim: Warning: Output is not to a terminal
[1]+ Stopped vi "hello world!" > test.txt
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage# ls
1.txt 10.txt 2.txt 3.txt 4.txt 5.txt 6.txt 7.txt 8.txt 9.txt lost+found test.txt
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage# vi test.txt
[2]+ Stopped vi test.txt
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage# cd ..
root@Linux-BhargaviNukala: /home/Linuxuser# umount

```

Unmount the additional Storage using the command
\$umount additionalstorage/

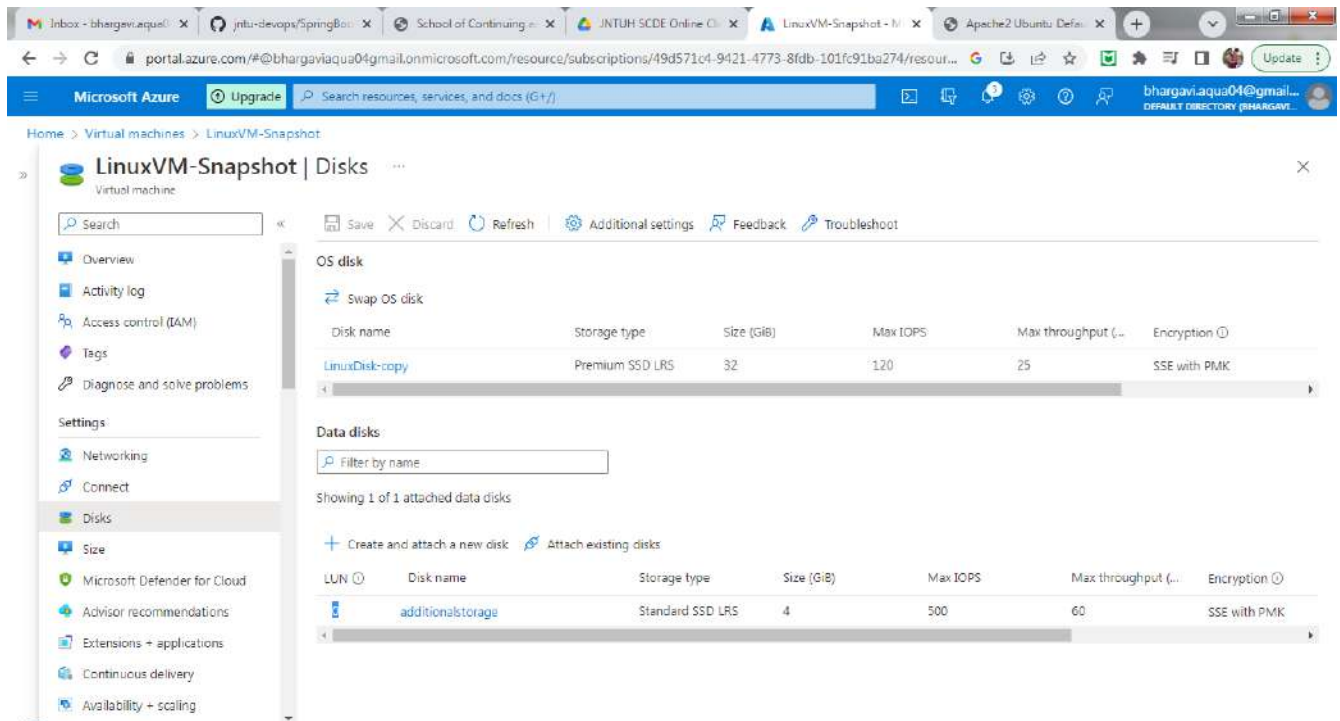
```

root@Linux-BhargaviNukala: /home/Linuxuser
Terminal Sessions View Xserver Tools Games Settings Macros Help
Season Servers Tools Games Sessions View Split Multitex Tunneling Padiages Settings Help
Quick connect... 3.root@Linux-BhargaviNukala: /home/Linuxuser 6.root@Linux-BhargaviNukala: /home/Linuxuser
15 updates can be applied immediately.
To see these additional updates run: apt list --upgradable
New release '22.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
Last login: Tue Dec 13 06:15:10 2022 from 49.37.131.16
Linuxuser@Linux-BhargaviNukala:~$ sudo su
root@Linux-BhargaviNukala:/home/Linuxuser# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
loop0 7:0 0 63.2M 1 loop /snap/core20/1695
loop1 7:1 0 91.9M 1 loop /snap/lxd/24061
loop2 7:2 0 49.6M 1 loop /snap/snappy/17883
loop3 7:3 0 63.2M 1 loop /snap/core20/1738
sda 8:0 0 30G 0 disk
├─sda1 8:1 0 29.9G 0 part /
├─sda14 8:14 0 4M 0 part
├─sda15 8:15 0 196M 0 part /boot/efi
├─sdb 8:16 0 4G 0 disk
├─sdb1 8:17 0 4G 0 part /mnt
├─sdc 8:32 0 4G 0 disk /home/Linuxuser/additionalstorage
└─sr0 11:0 1 628K 0 rom
root@Linux-BhargaviNukala:/home/Linuxuser# mount additionalstorage/
root@Linux-BhargaviNukala:/home/Linuxuser# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
loop0 7:0 0 63.2M 1 loop /snap/core20/1695
loop1 7:1 0 91.9M 1 loop /snap/lxd/24061
loop2 7:2 0 49.6M 1 loop /snap/snappy/17883
loop3 7:3 0 63.2M 1 loop /snap/core20/1738
sda 8:0 0 30G 0 disk
├─sda1 8:1 0 29.9G 0 part /
├─sda14 8:14 0 4M 0 part
├─sda15 8:15 0 196M 0 part /boot/efi
├─sdb 8:16 0 4G 0 disk
├─sdb1 8:17 0 4G 0 part /mnt
├─sdc 8:32 0 4G 0 disk
└─sr0 11:0 1 628K 0 rom
root@Linux-BhargaviNukala:/home/Linuxuser#

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Now attach the same disk to another VM



Connect to the VM and check the disk is already mounted and the additionalStorage is already available on the disk. Also the 10 files which we created in previous VM are also available in this VM.

```

root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage
Terminal Sessions View Xserver Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MuTBec Tunneling Packages Settings Help
Quick connect...
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage
mkdir: cannot create directory 'additionalstorage': File exists
root@Linux-BhargaviNukala: /home/Linuxuser# ls
additionalstorage
root@Linux-BhargaviNukala: /home/Linuxuser# cd additionalstorage/
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage# ls -ltr
total 0
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage# cd ..
root@Linux-BhargaviNukala: /home/Linuxuser# mount -t ext4 /dev/sdc additionalstorage/
root@Linux-BhargaviNukala: /home/Linuxuser# cd additionalstorage/
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage# ls -ltr
total 32
drwx----- 2 root root 16384 Dec 13 06:30 lost+found
-rw-r--r-- 1 root root 0 Dec 13 06:36 9.txt
-rw-r--r-- 1 root root 0 Dec 13 06:36 8.txt
-rw-r--r-- 1 root root 0 Dec 13 06:36 7.txt
-rw-r--r-- 1 root root 0 Dec 13 06:36 6.txt
-rw-r--r-- 1 root root 0 Dec 13 06:36 5.txt
-rw-r--r-- 1 root root 0 Dec 13 06:36 4.txt
-rw-r--r-- 1 root root 0 Dec 13 06:36 3.txt
-rw-r--r-- 1 root root 0 Dec 13 06:36 2.txt
-rw-r--r-- 1 root root 0 Dec 13 06:36 10.txt
-rw-r--r-- 1 root root 0 Dec 13 06:36 1.txt
-rw-r--r-- 1 root root 13596 Dec 13 07:14 test.txt
root@Linux-BhargaviNukala: /home/Linuxuser/additionalstorage#

```

UNREGISTERED VERSION - Please support Mobaxterm by subscribing to the professional edition here: <https://mobaxterm.mobaxterm.net>

Snapshot:

Create a Linux VM and install apache2.

\$apt update

\$apt install apache2

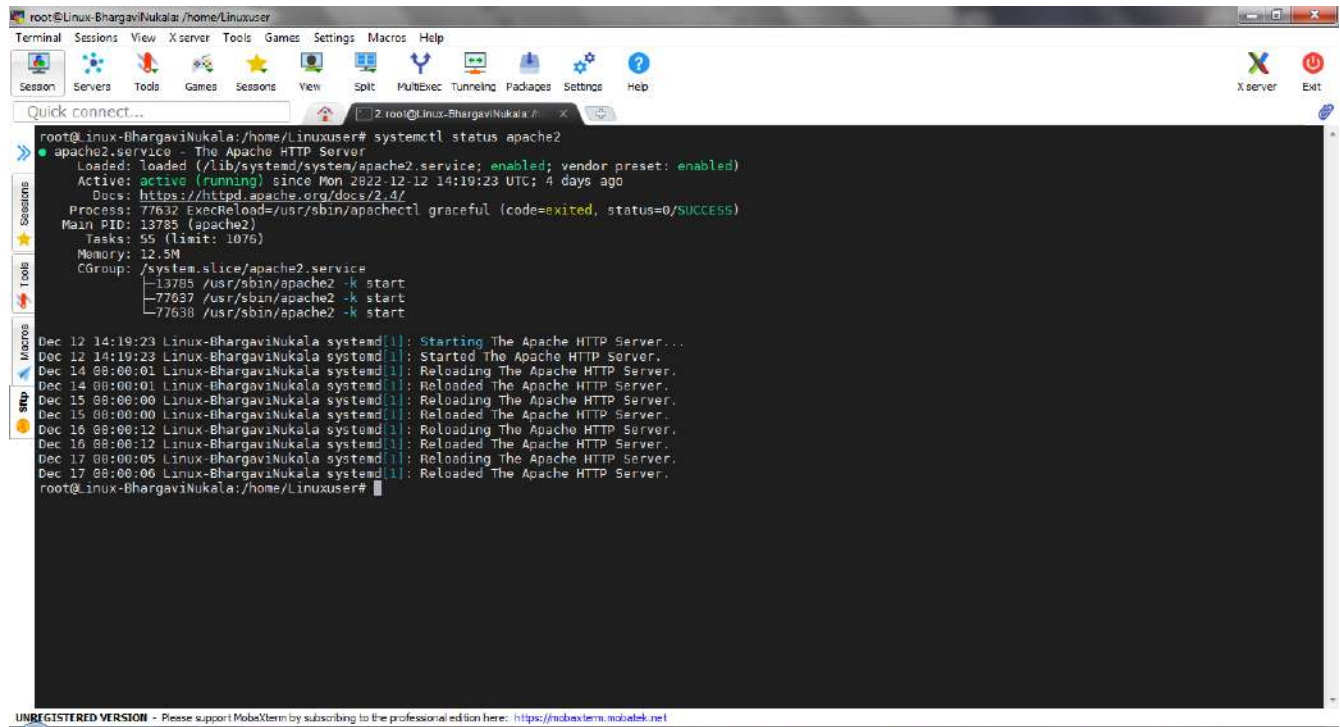
The screenshot shows the Azure portal interface for a virtual machine. The main content area displays the 'Essentials' section for the VM 'Linux-BhargaviNukala'. The 'Properties' tab is selected, showing the following details:

- Resource group:** AzureRG-BhargaviNukala
- Operating system:** Linux (ubuntu 20.04)
- Status:** Running
- Size:** Standard B1s (1 vcpu, 1 GiB memory)
- Location:** East US
- Public IP address:** 4.236.142.42
- Subscription:** Free Trial
- Subscription ID:** 49d571c4-9421-4773-8fdb-101fc91ba274
- Tags:** Not configured

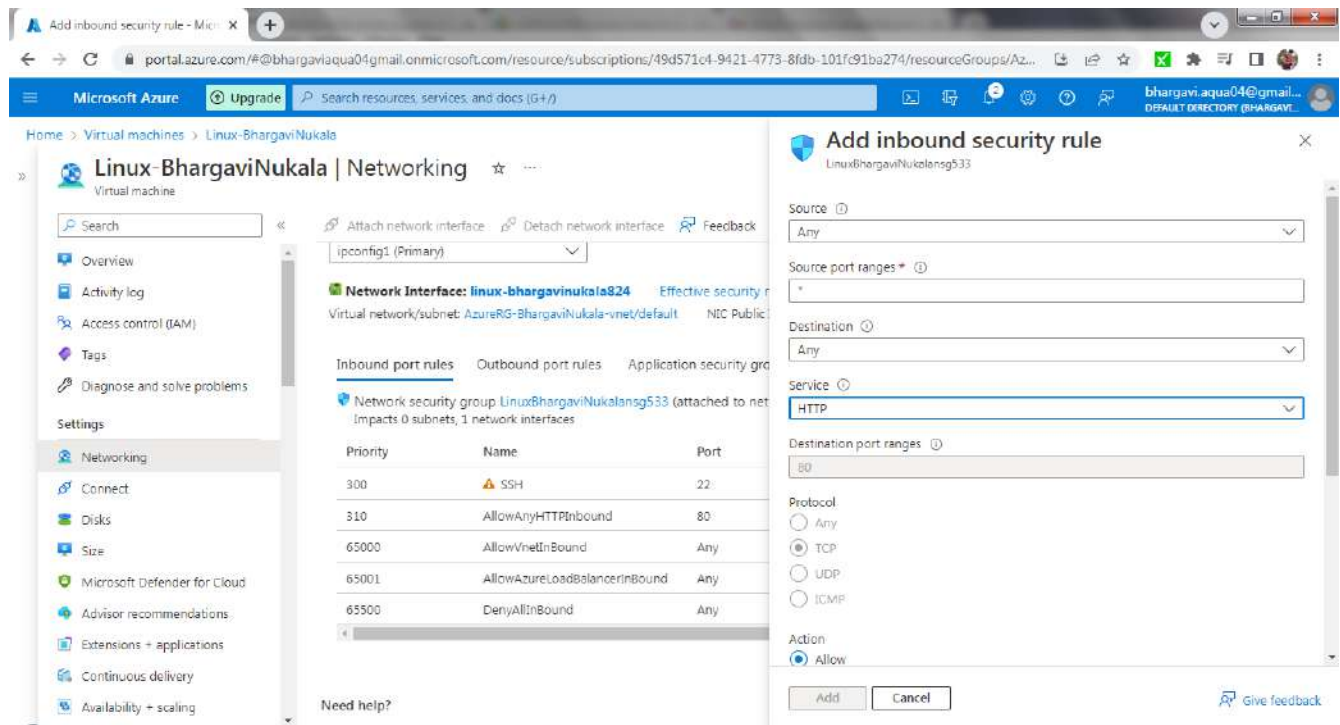
Below the Essentials section, there are two summary cards:

- Virtual machine:**
 - Computer name: Linux-BhargaviNukala
 - Health state: -
 - Operating system: Linux (ubuntu 20.04)
- Networking:**
 - Public IP address: 4.236.142.42
 - Public IP address (IPv6): -
 - Private IP address: 10.0.0.5

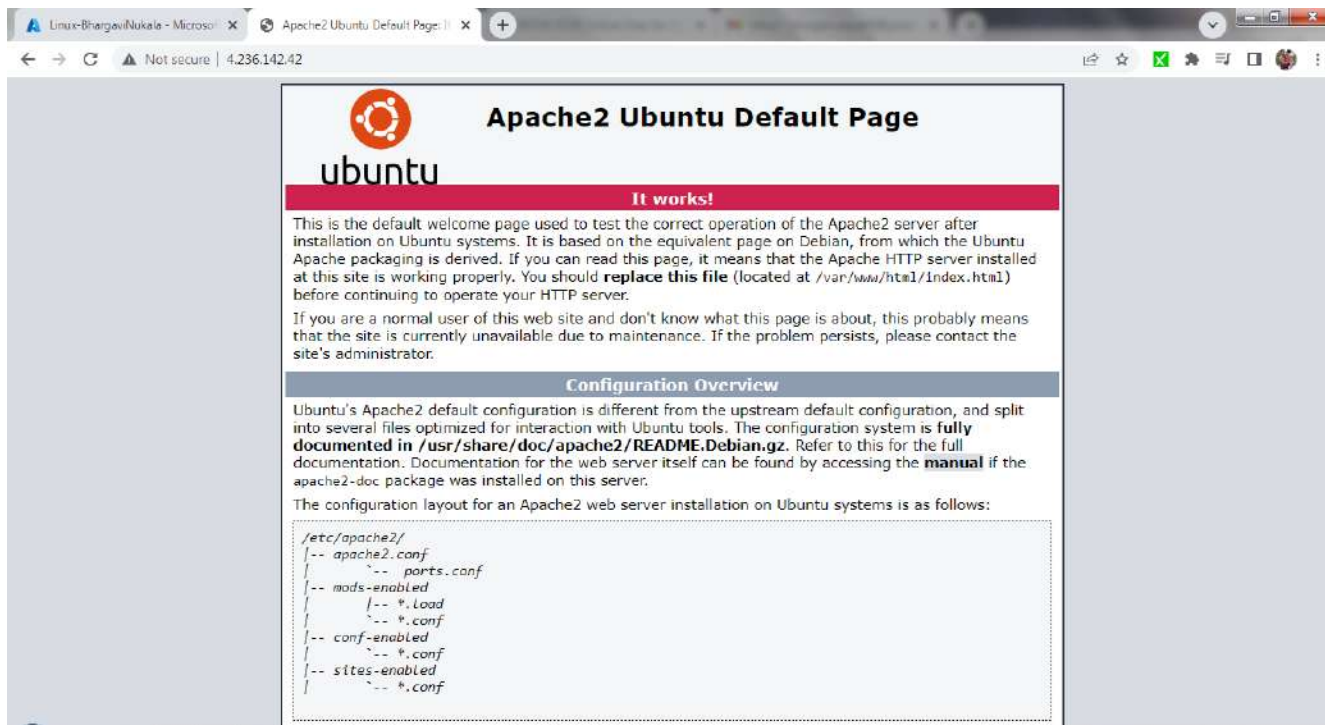
Check the status of apache2 running
\$systemctl status apache2



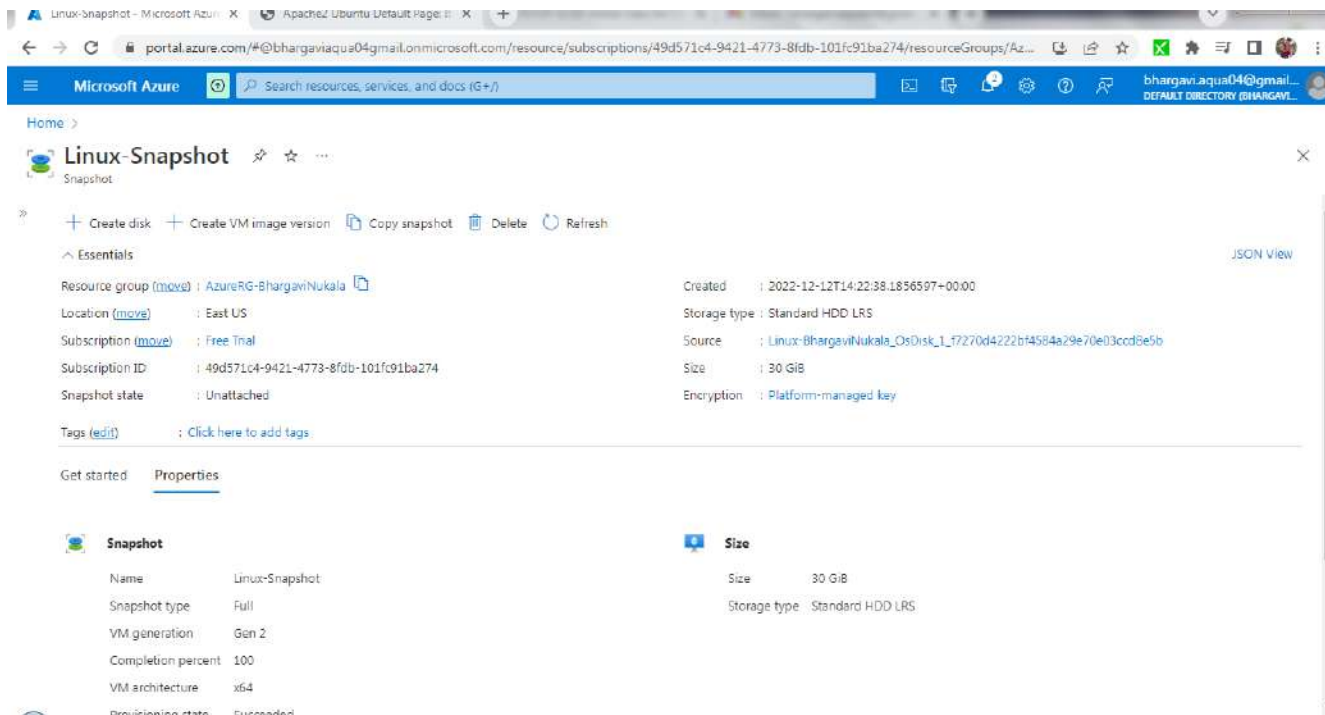
Add the Inbound rule for HTTP-80 port in Networking tab of VM



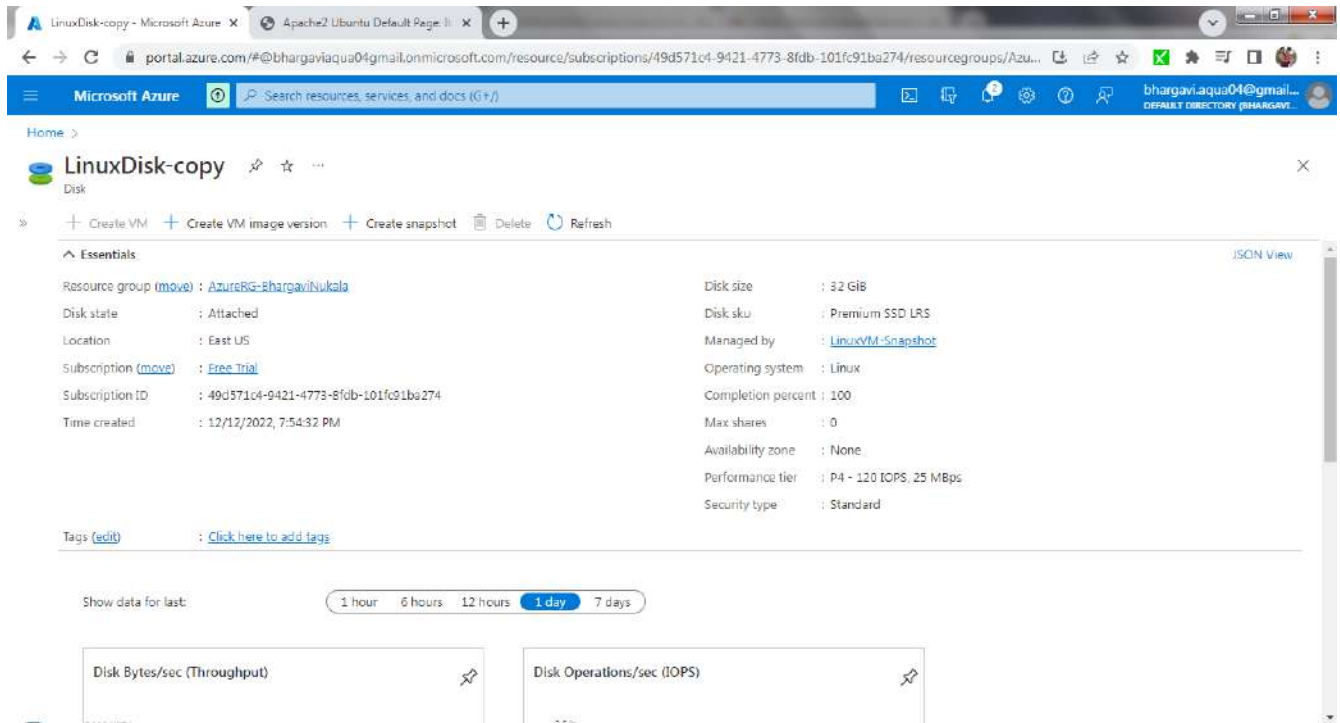
Access the webpage from browser using public IP of VM



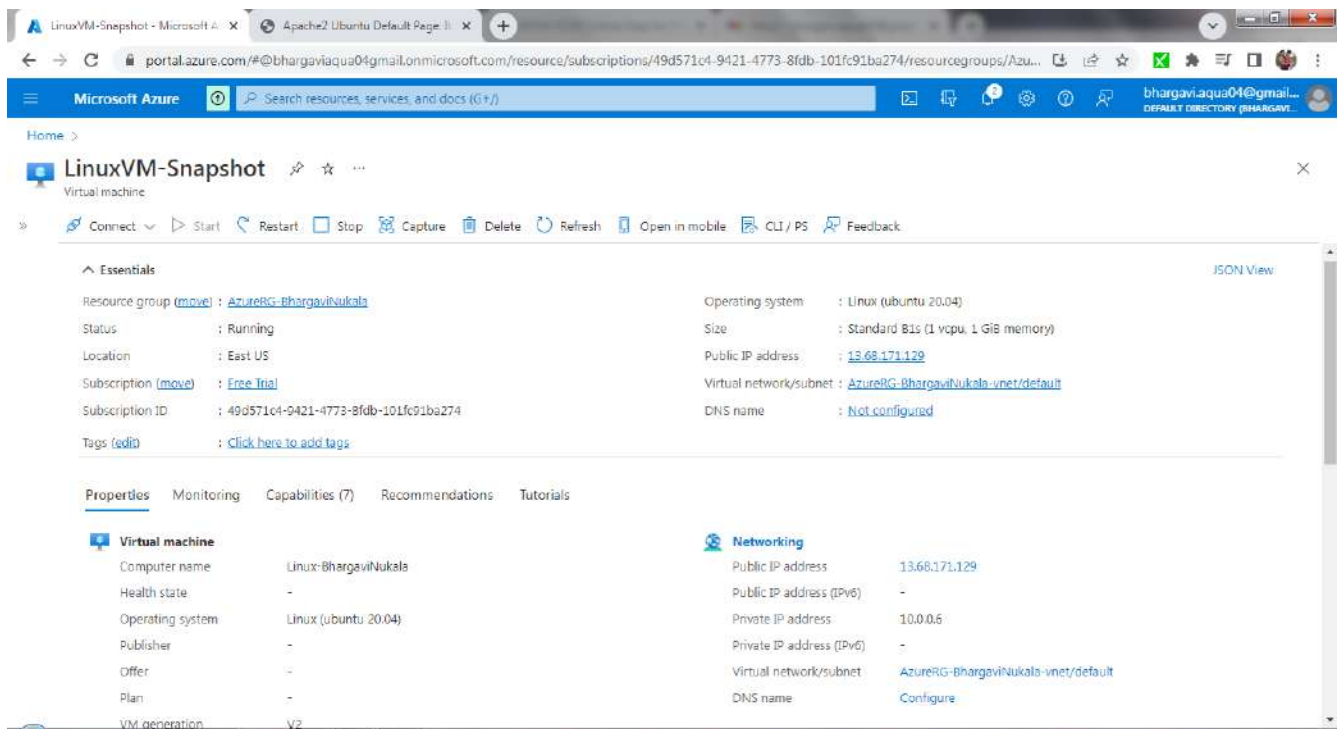
Go to VM->Disks->Create Snapshot of the disk



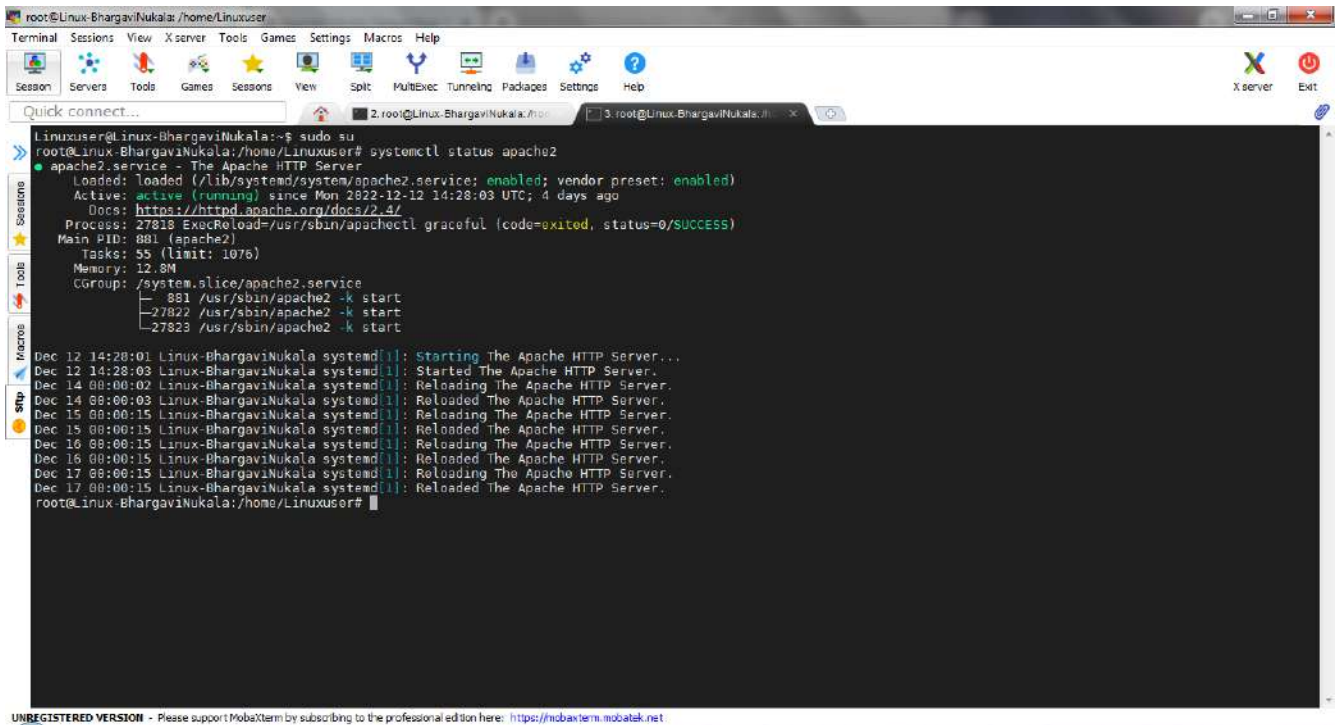
Now create a disk from the created snapshot
Goto Created Snapshot->Create Disk



From the newly created disk-> create a VM
Select the above created disk in OS dropdown box

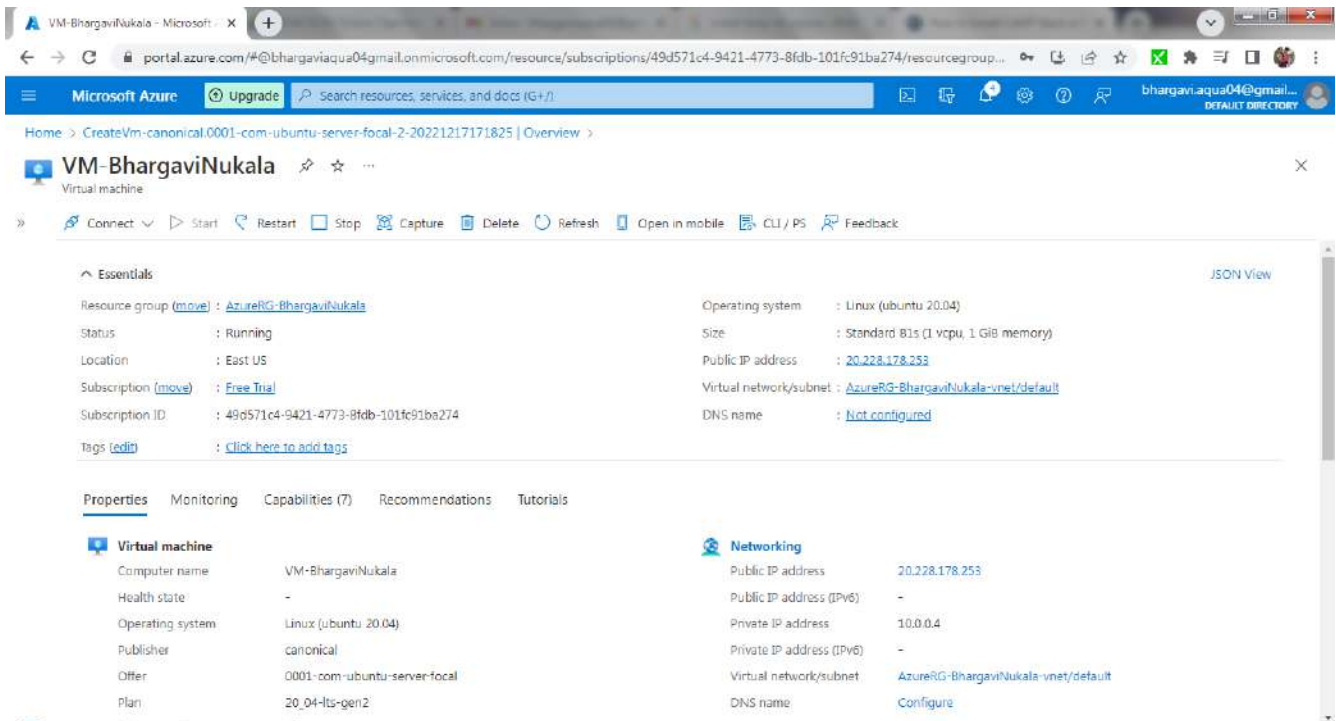


After the new VM is created from the snapshot, check the status of apache2.
We can see that apache2 is already installed.



Machine Image:

Create an Ubuntu machine and install apache2 and php as mentioned in the link <https://www.tecmint.com/install-lamp-with-phpmyadmin-in-ubuntu-18-04/>



Install apache2 and php

```

root@VM-BhargaviNukala:/home/Linuxuser# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2022-12-17 12:02:18 UTC; 28s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 10735 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 10739 (apache2)
    Tasks: 6 (limit: 1076)
   Memory: 12.3M
   CGroup: /system.slice/apache2.service
           └─10739 /usr/sbin/apache2 -k start
             └─10741 /usr/sbin/apache2 -k start
               └─10742 /usr/sbin/apache2 -k start
                 └─10743 /usr/sbin/apache2 -k start
                   └─10744 /usr/sbin/apache2 -k start
                     └─10745 /usr/sbin/apache2 -k start

Dec 17 12:02:18 VM-BhargaviNukala systemd[1]: apache2.service: Succeeded.
Dec 17 12:02:18 VM-BhargaviNukala systemd[1]: Stopped The Apache HTTP Server.
Dec 17 12:02:18 VM-BhargaviNukala systemd[1]: Starting The Apache HTTP Server...
Dec 17 12:02:18 VM-BhargaviNukala systemd[1]: Started The Apache HTTP Server.
root@VM-BhargaviNukala:/home/Linuxuser#

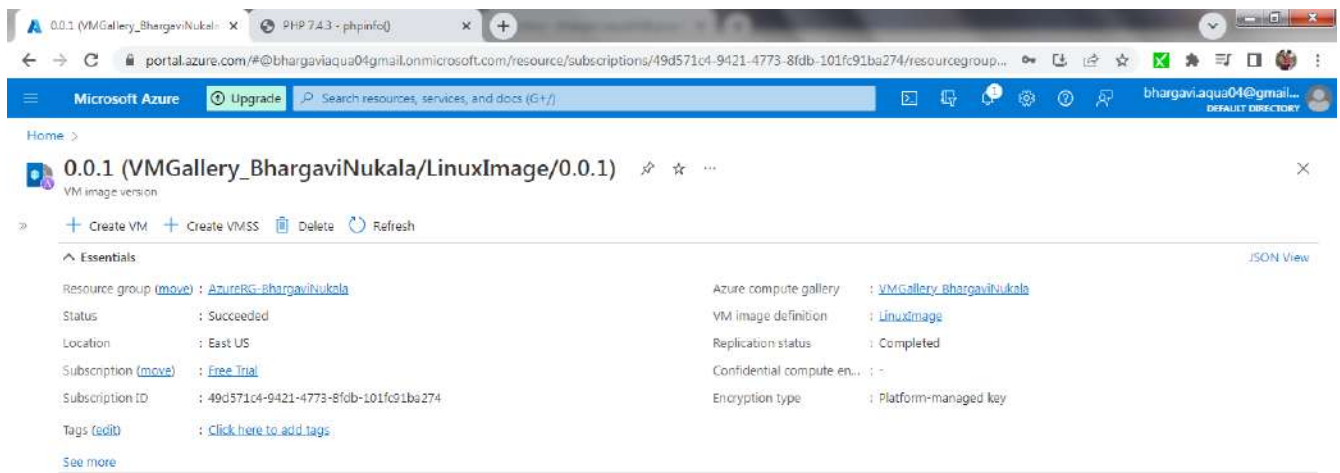
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

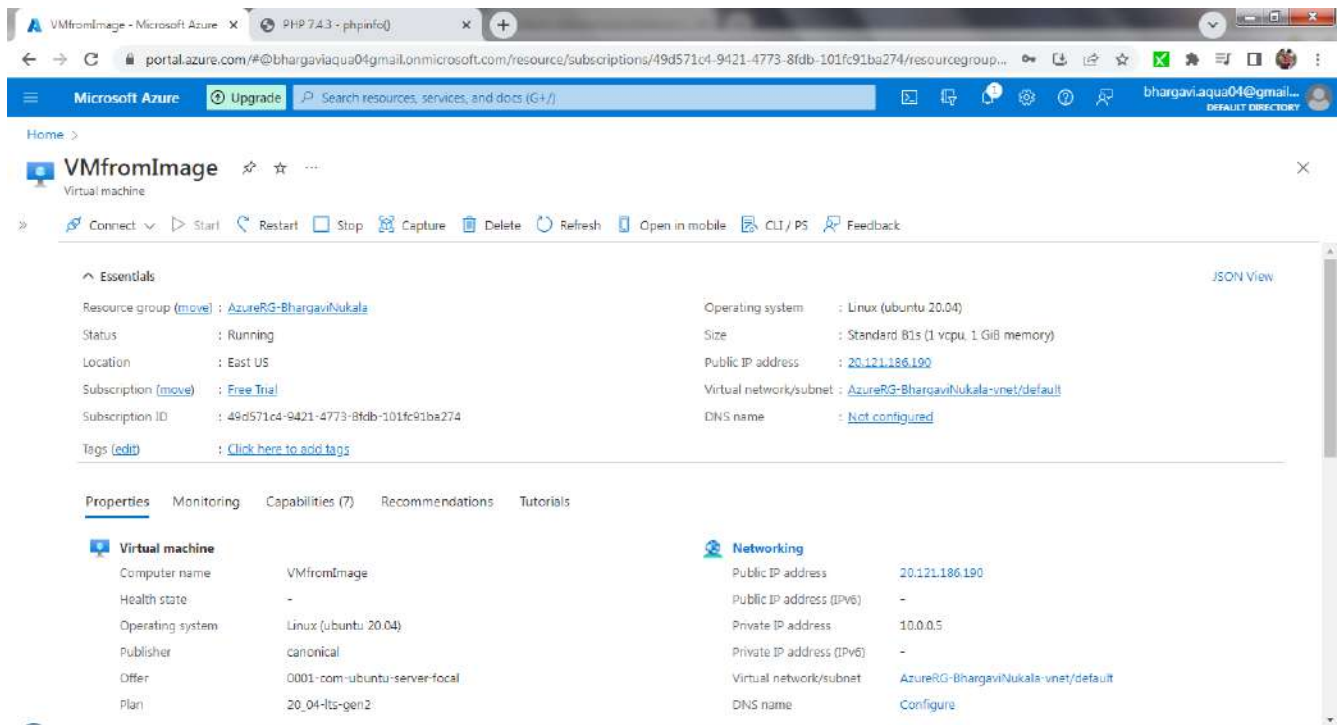
Access the webpage of VM using public IP/info.php



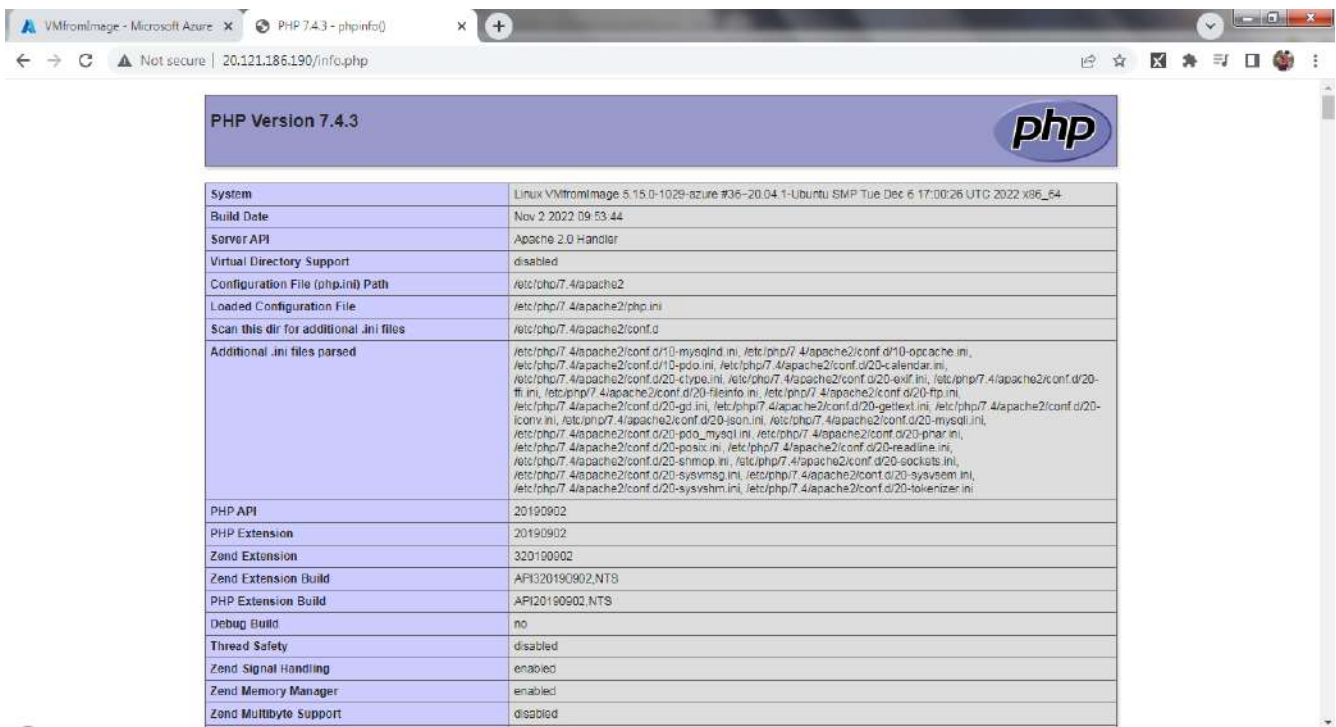
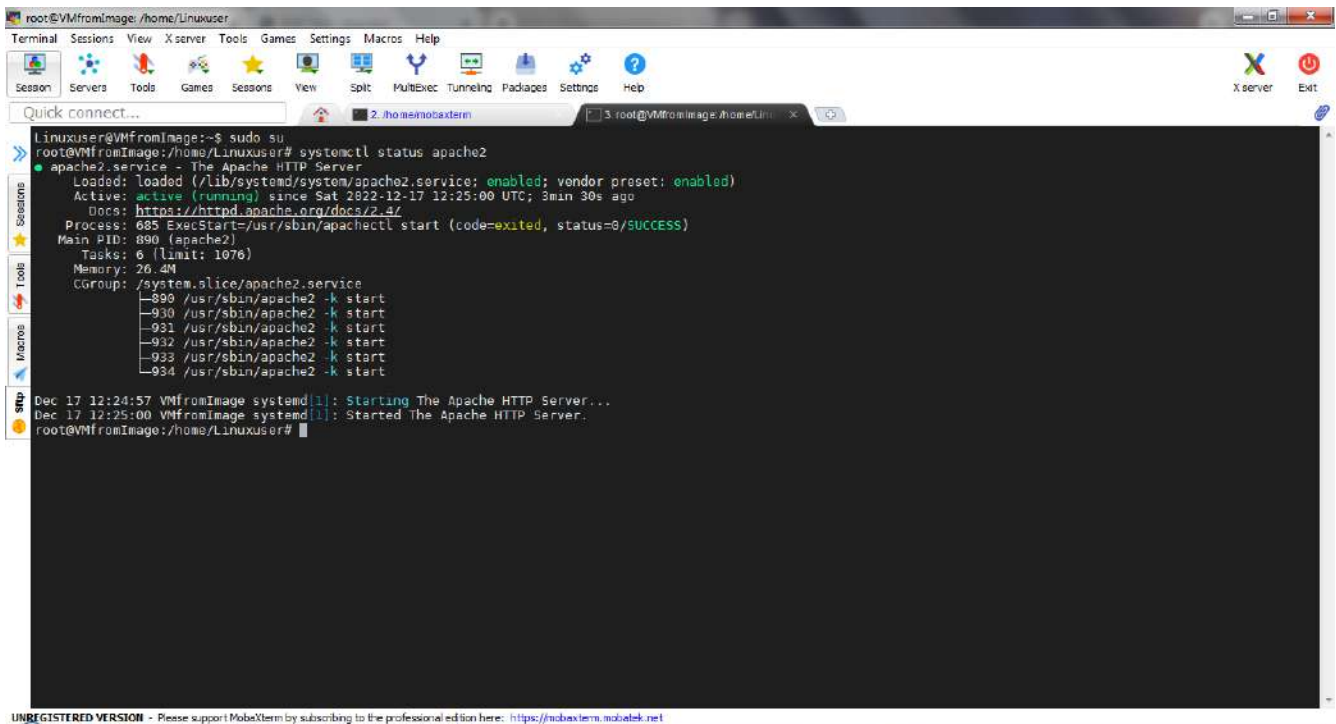
To create an image of the above created VM, goto VM->Capture



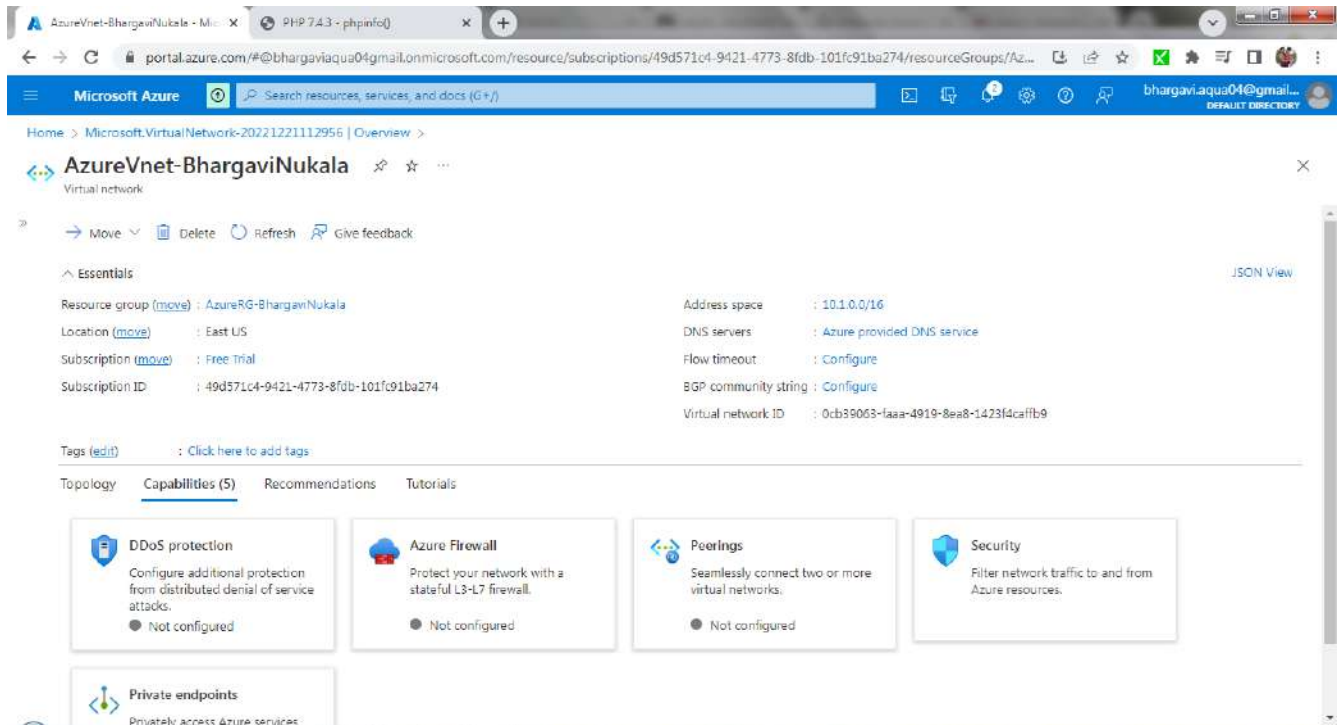
Create a VM from this Image by selecting this image in OS section of VM.



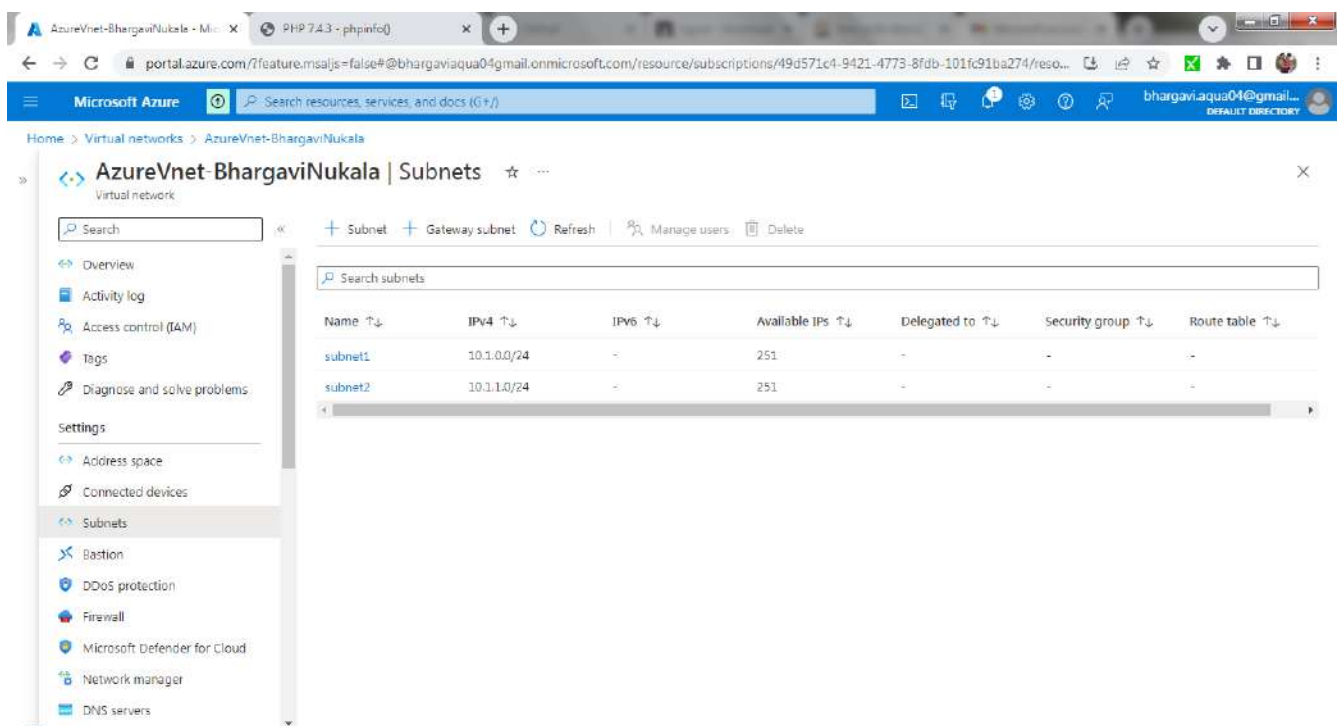
Check that apache2 and php are already installed in this new VM



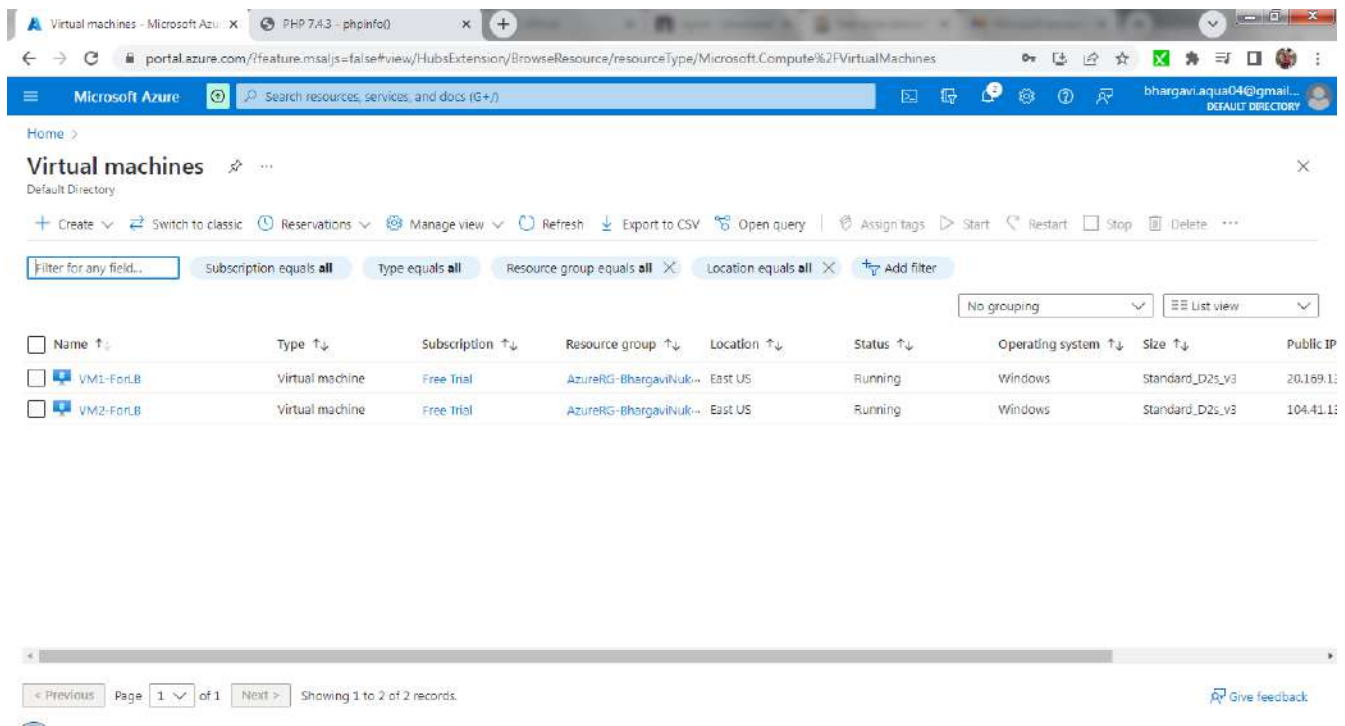
Network Security Group
 Create a Vnet with two subnets—use Virtual Networks to create a Vnet.



In the Vnet, create 2 subnets



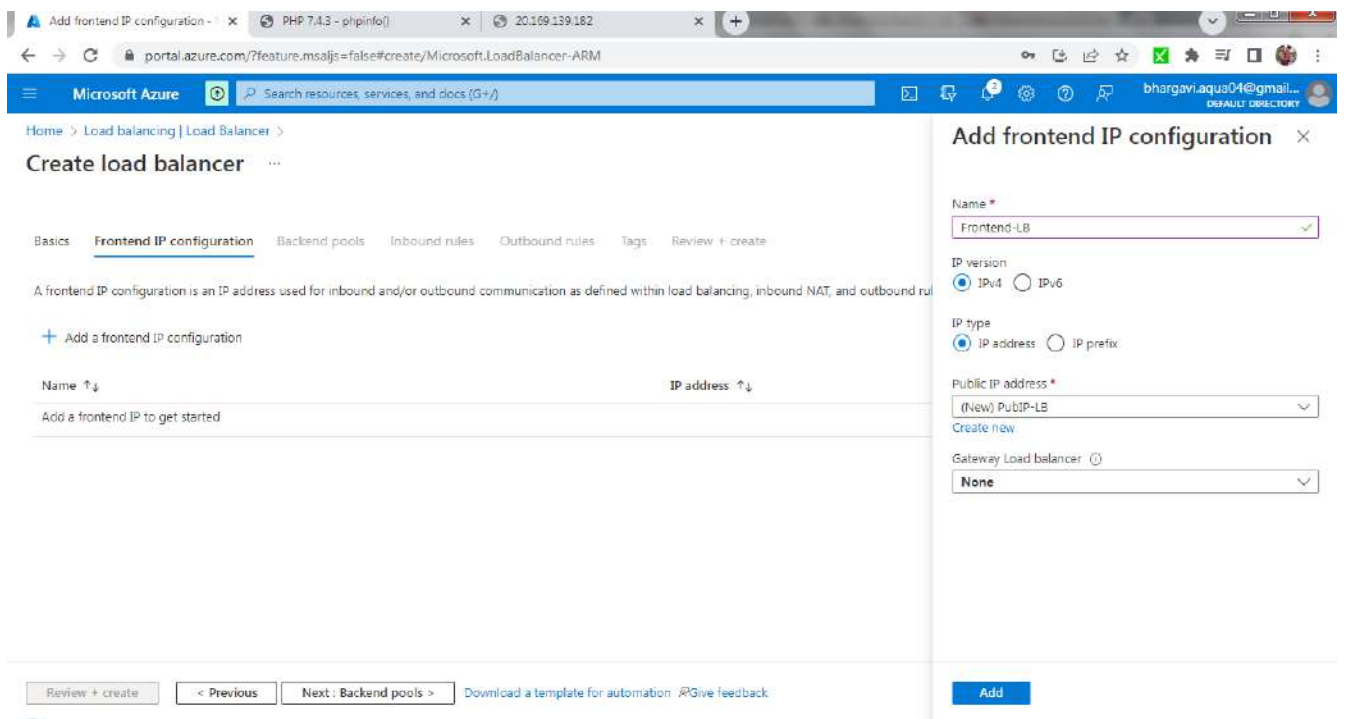
Now create one VM in each subnet.

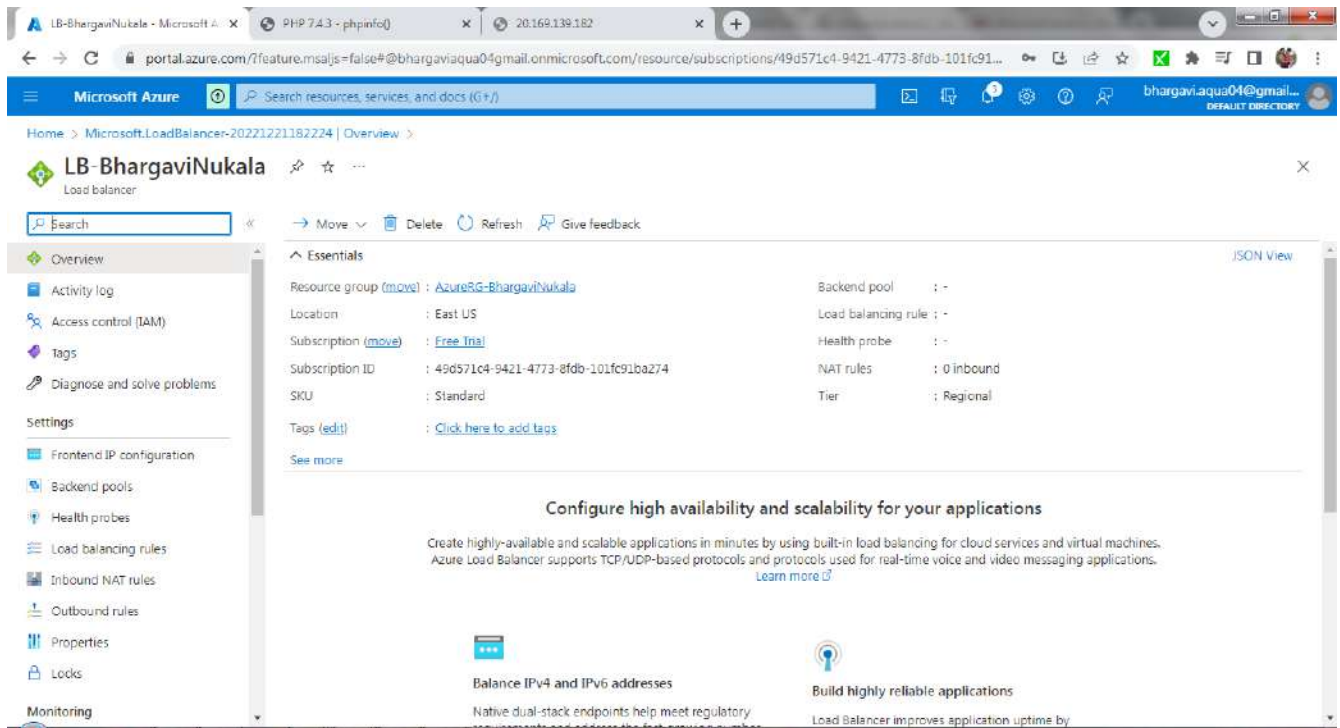


Setup IIS server on both VM s. As we have created Vnet with default Inbound and Outbound rules, with default rules, only Inbound traffic from same Vnet are allowed and from outside is not allowed.

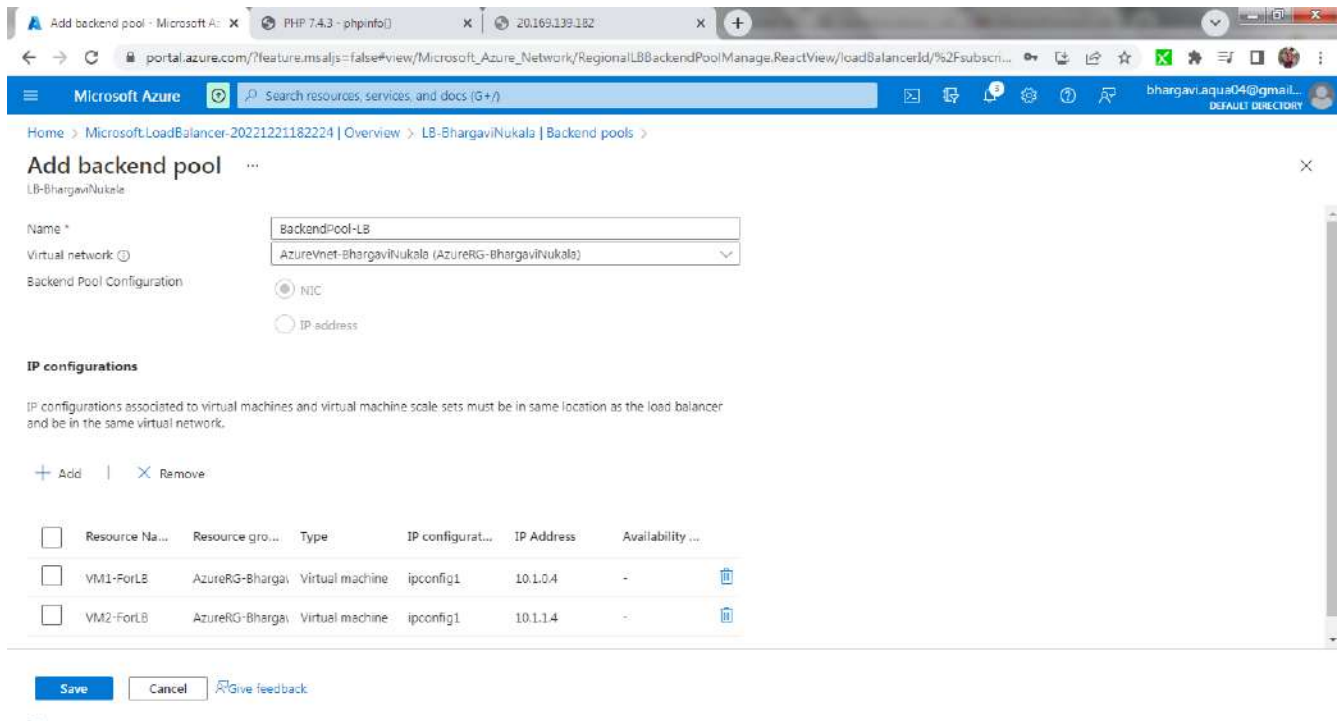
Load Balancer:

Create a Load Balancer by adding a new Frontend Configuration and adding a new Public IP.

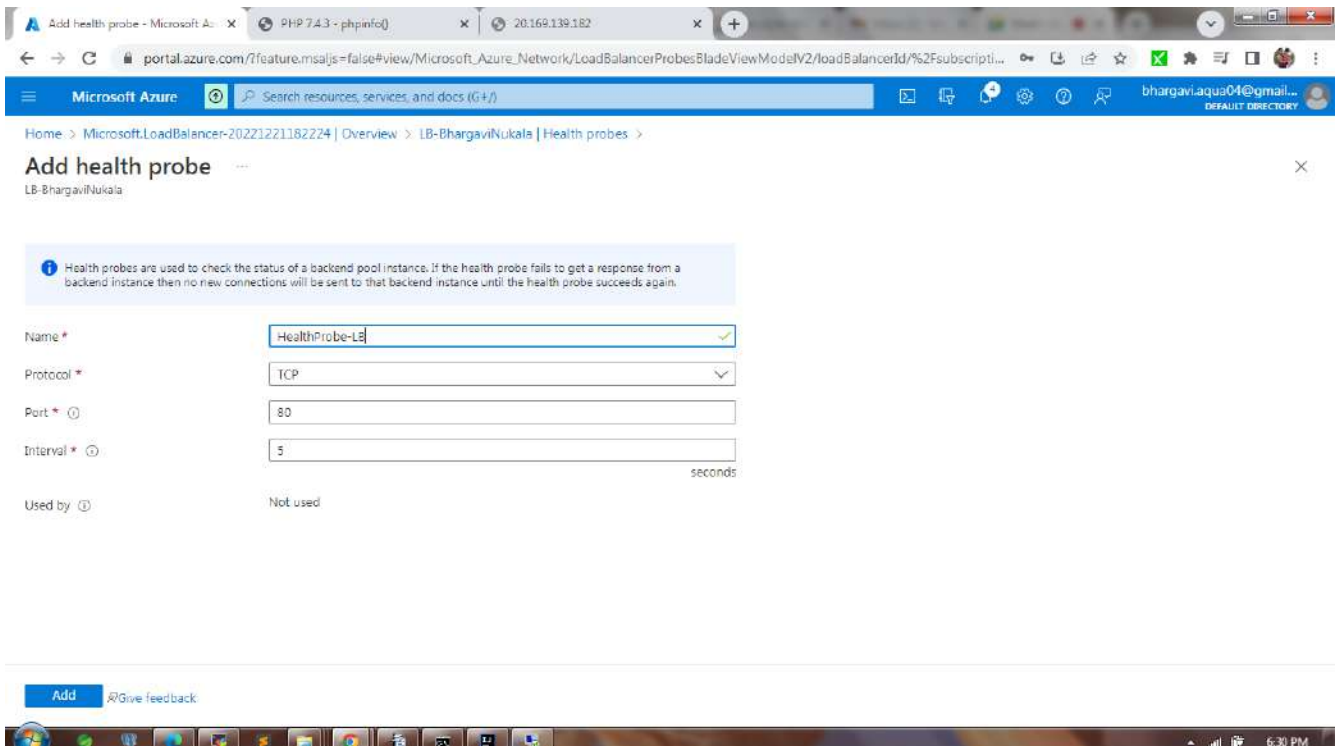




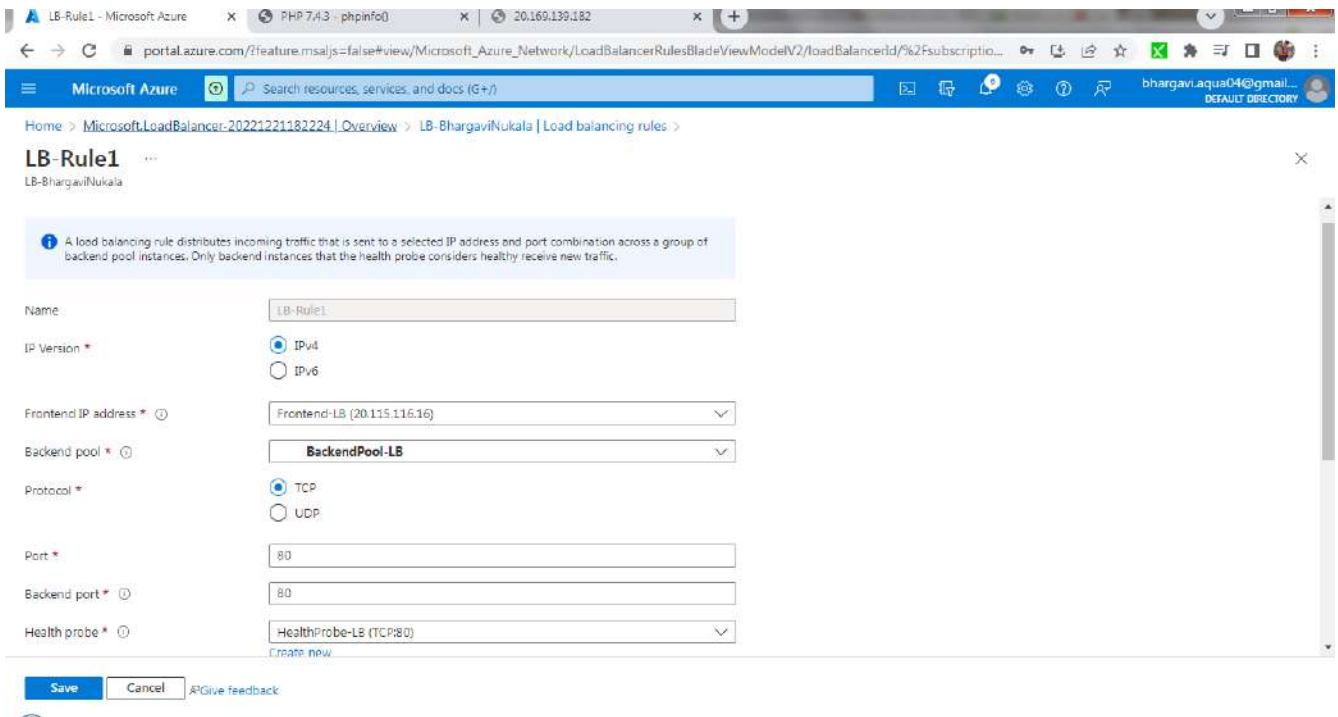
Now add Backend pools for LB. Select the created Vnet and the machines we created in 2 subnets and add them to Backend Pool of LB.



Add a health probe for LB



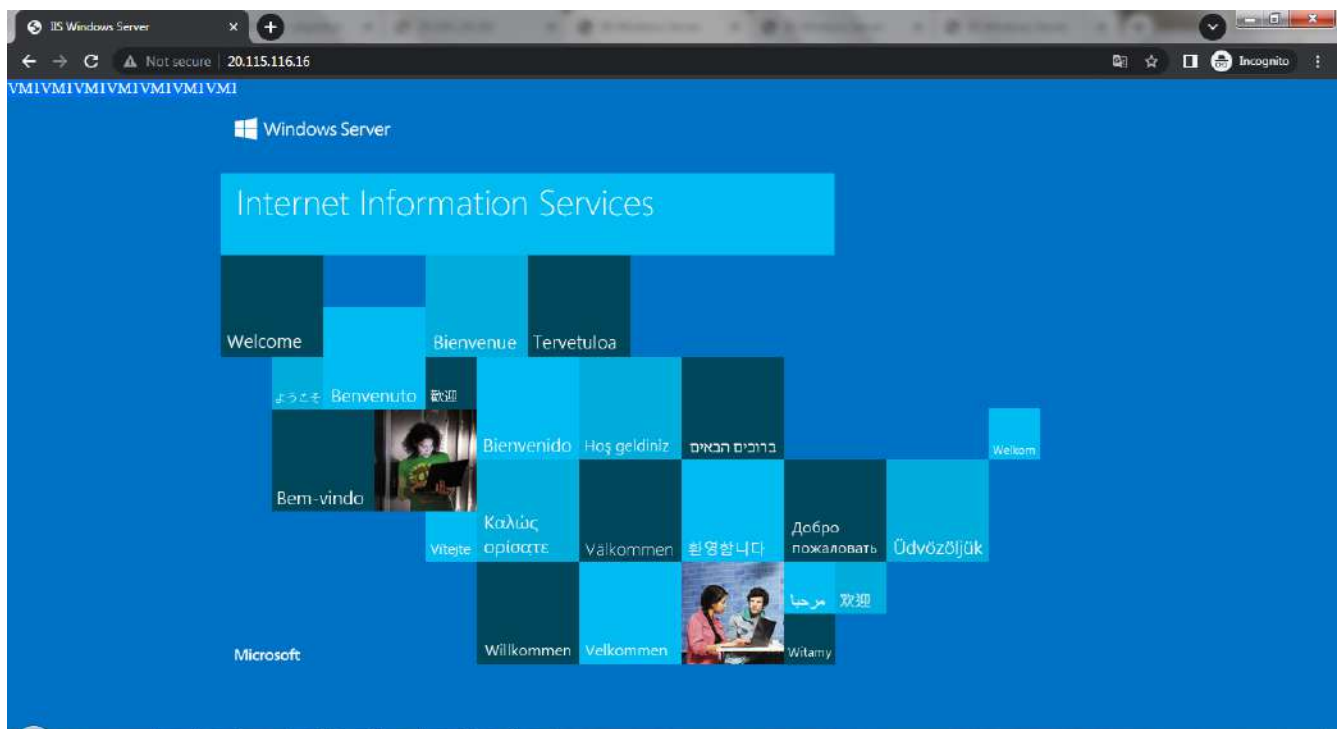
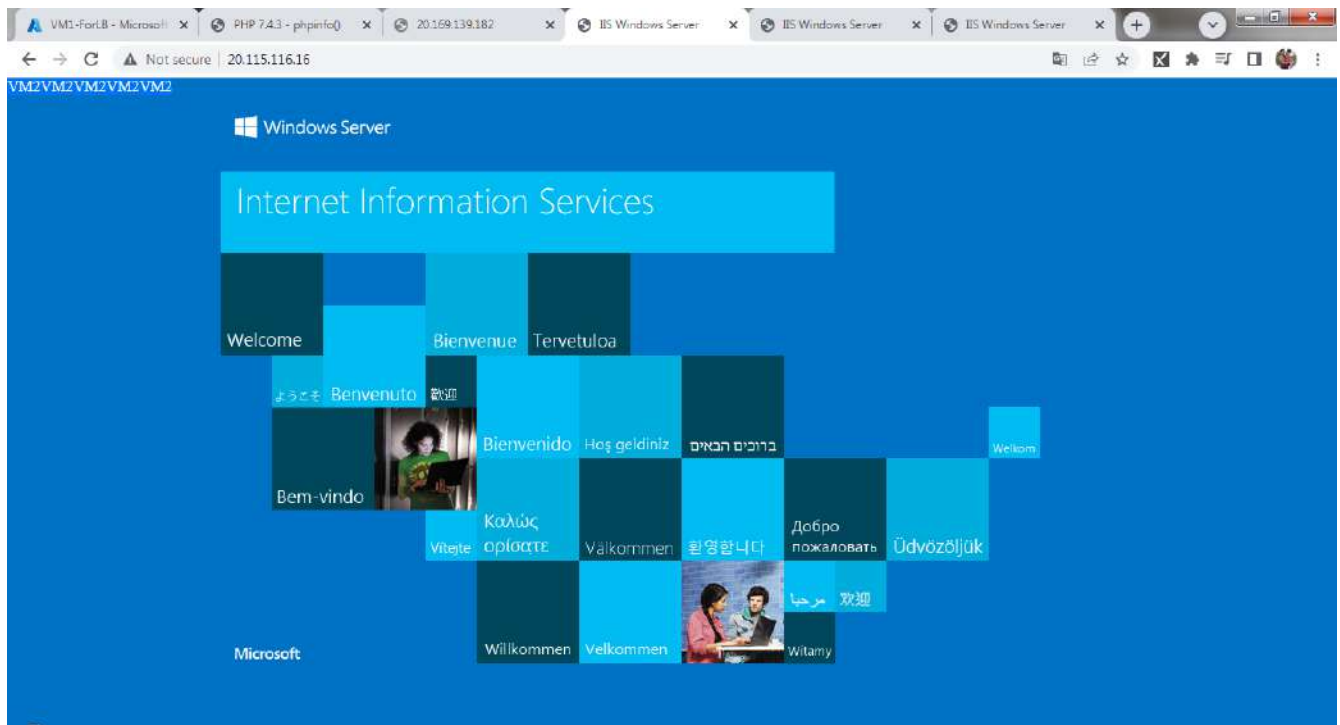
Add Load Balancing Rules by selecting the above created Frontend, Backend pool and health probe



Check that IIS server is installed on two VMs. Make changes in web pages located at C:\inetpub\wwwroot to display VM1 and VM2.

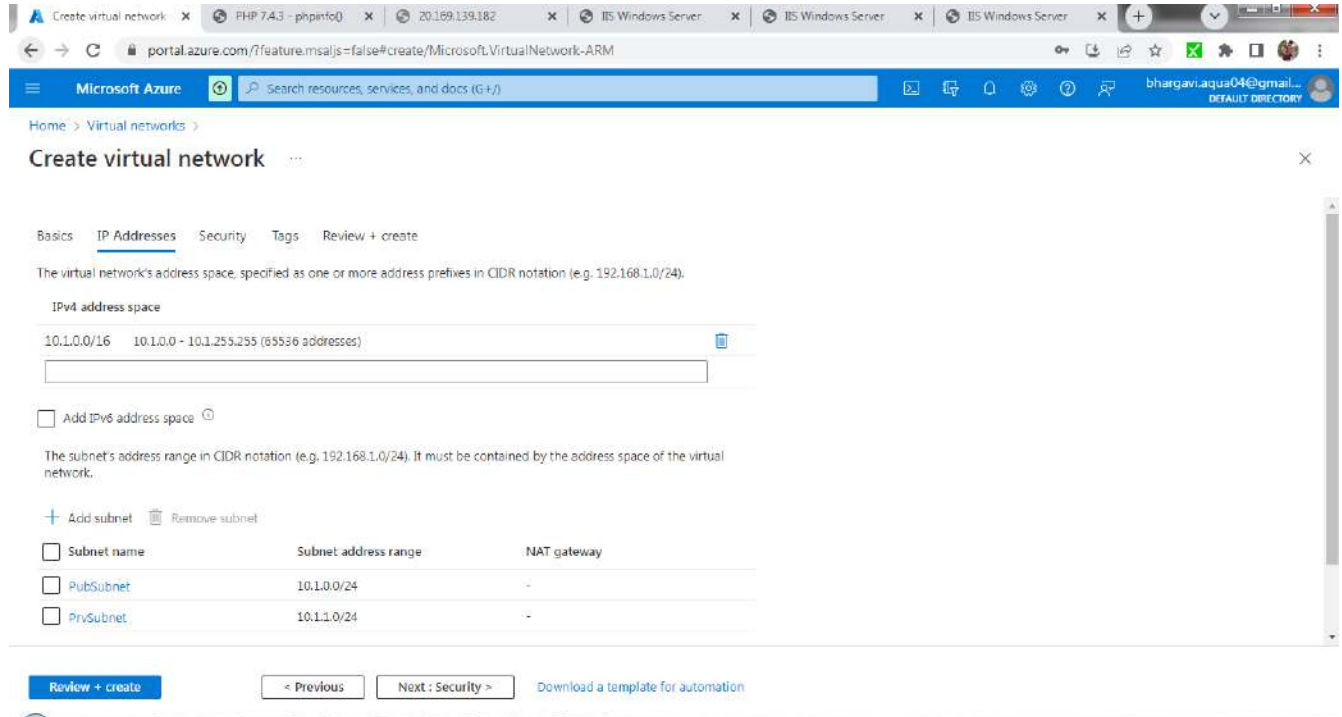
Access webpage from browser using public IP of VM.

As seen, once VM1 is called and once VM2 is called by Load Balancer.

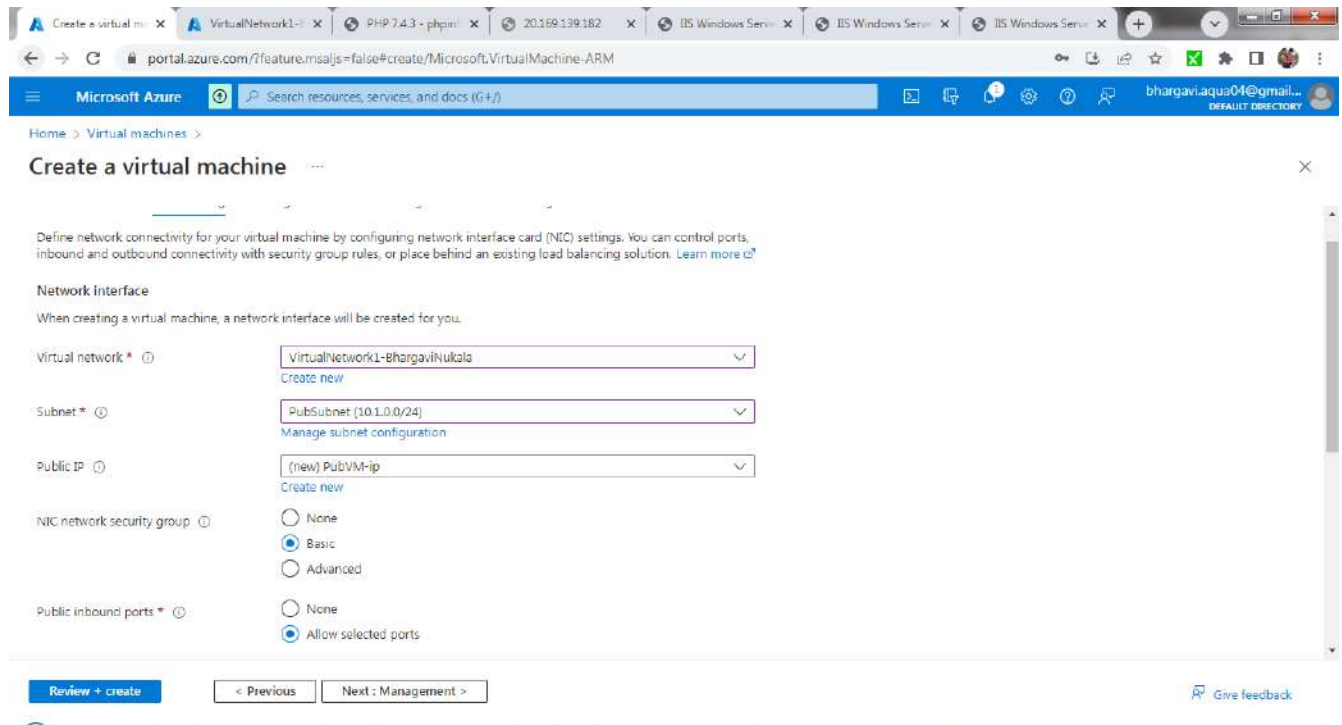


VPC:

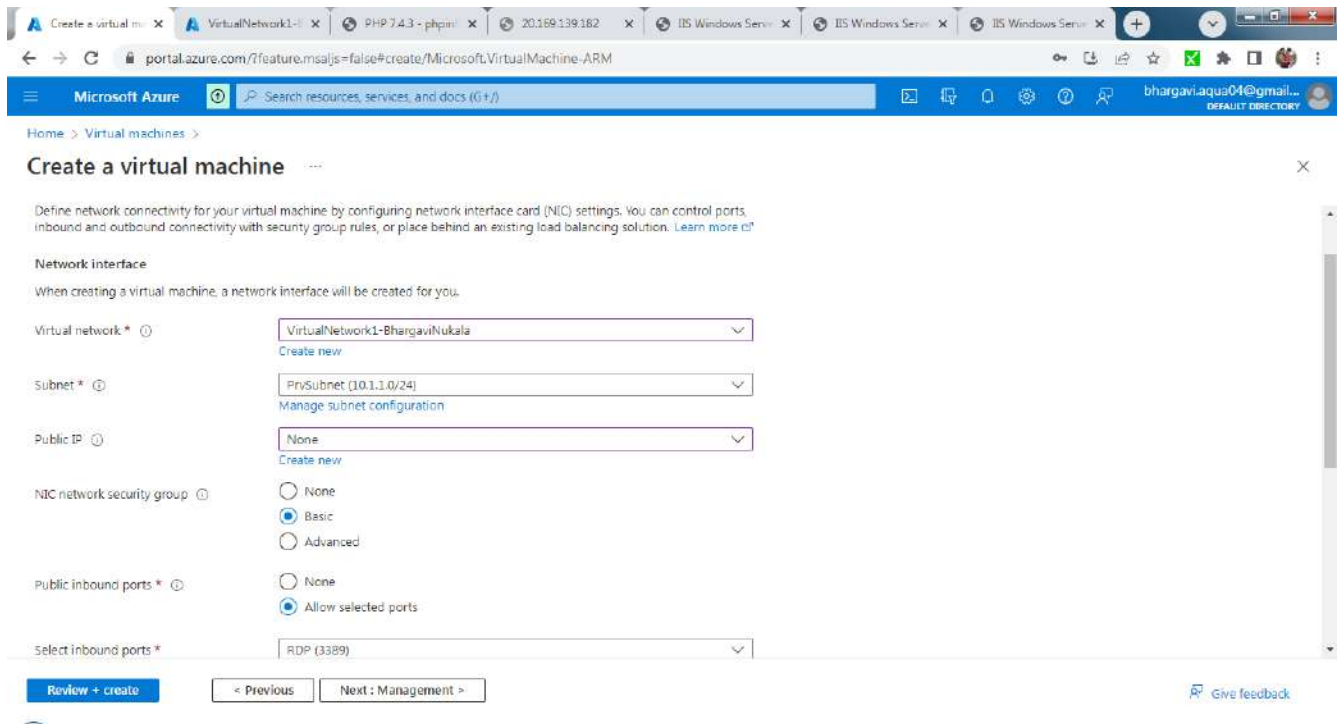
Create a Virtual Network. Add 2 subnets, one public subnet and one private subnet.



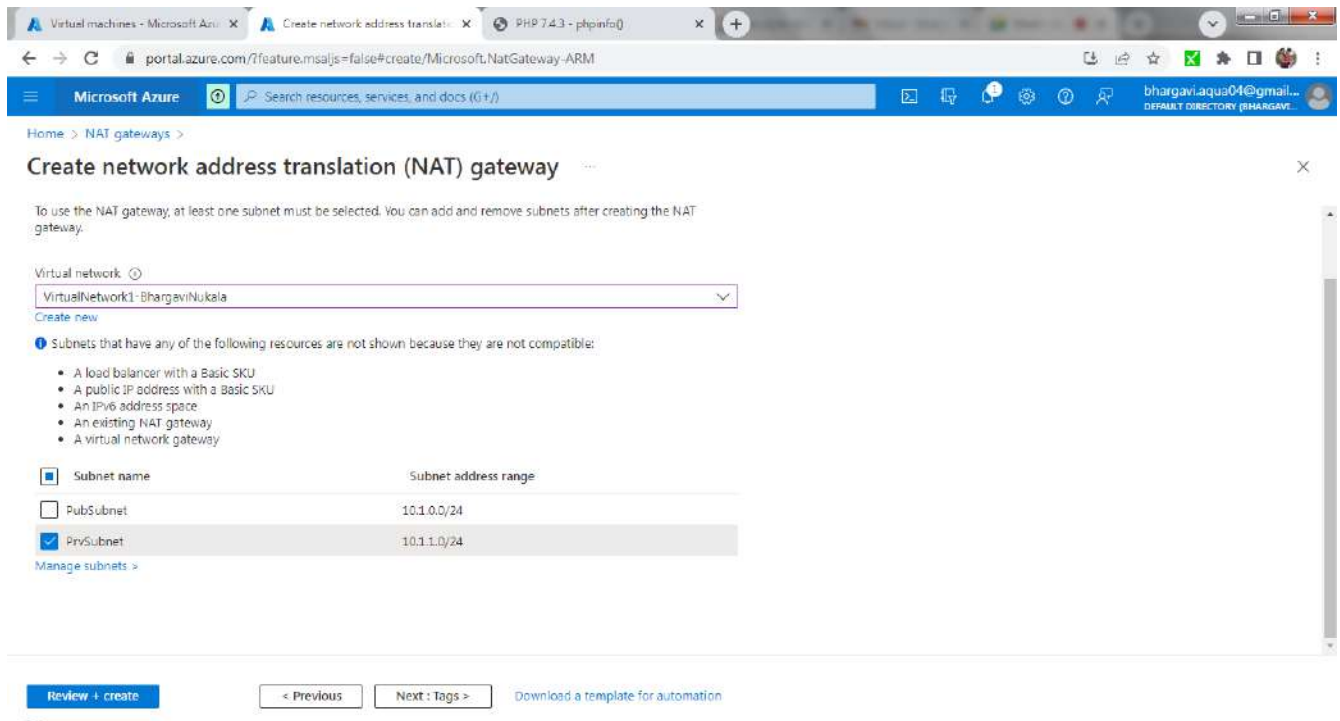
Create 2 Vms, one in Public subnet and one VM in private subnet. Public VM should be created by selecting the public IP.



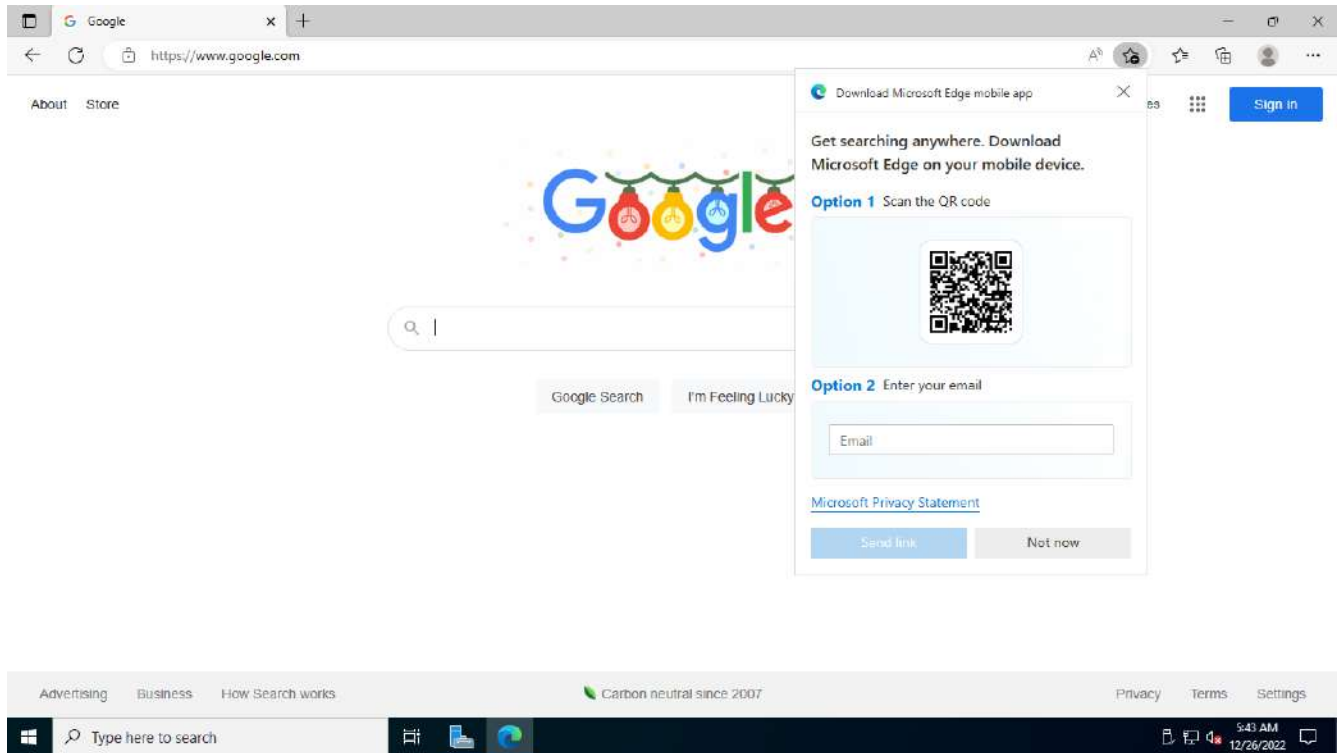
Private VM should be created by not selecting the public IP.



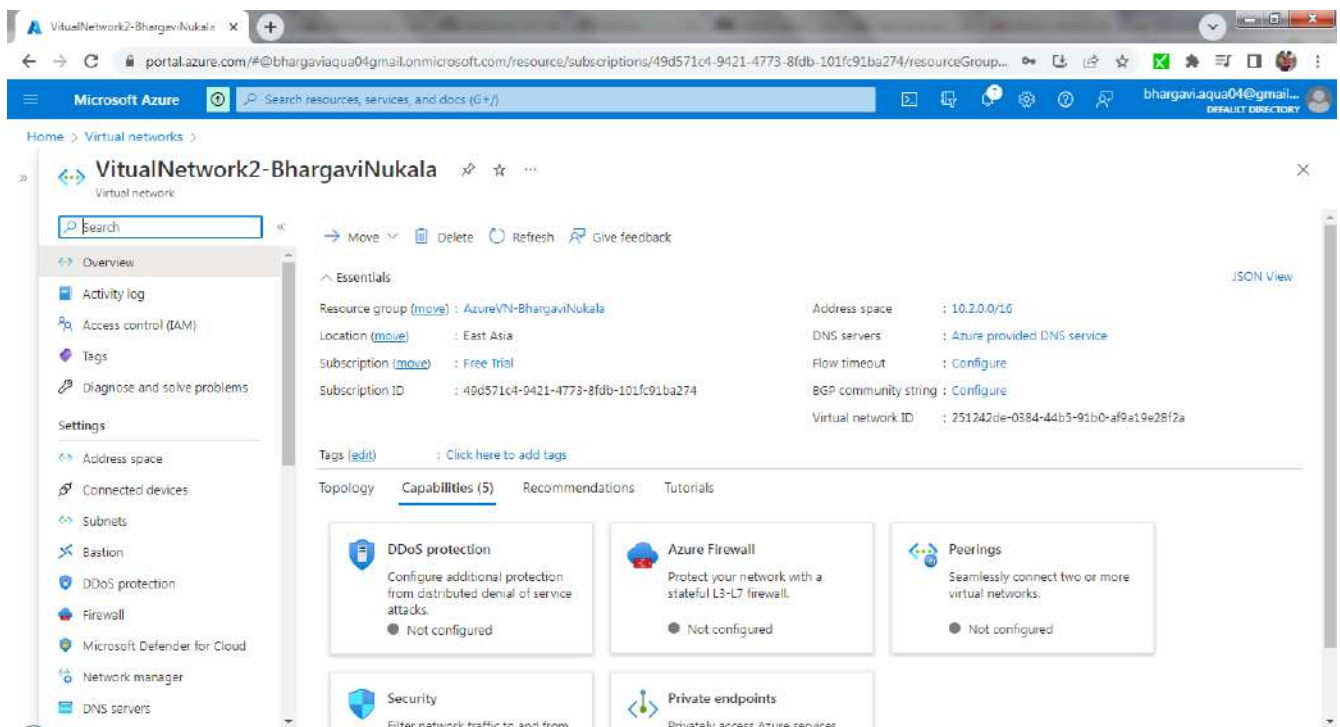
Create a NAT Gateway to get Internet to Private Subnet. Select the Virtual Network and the private subnet where you want Internet.



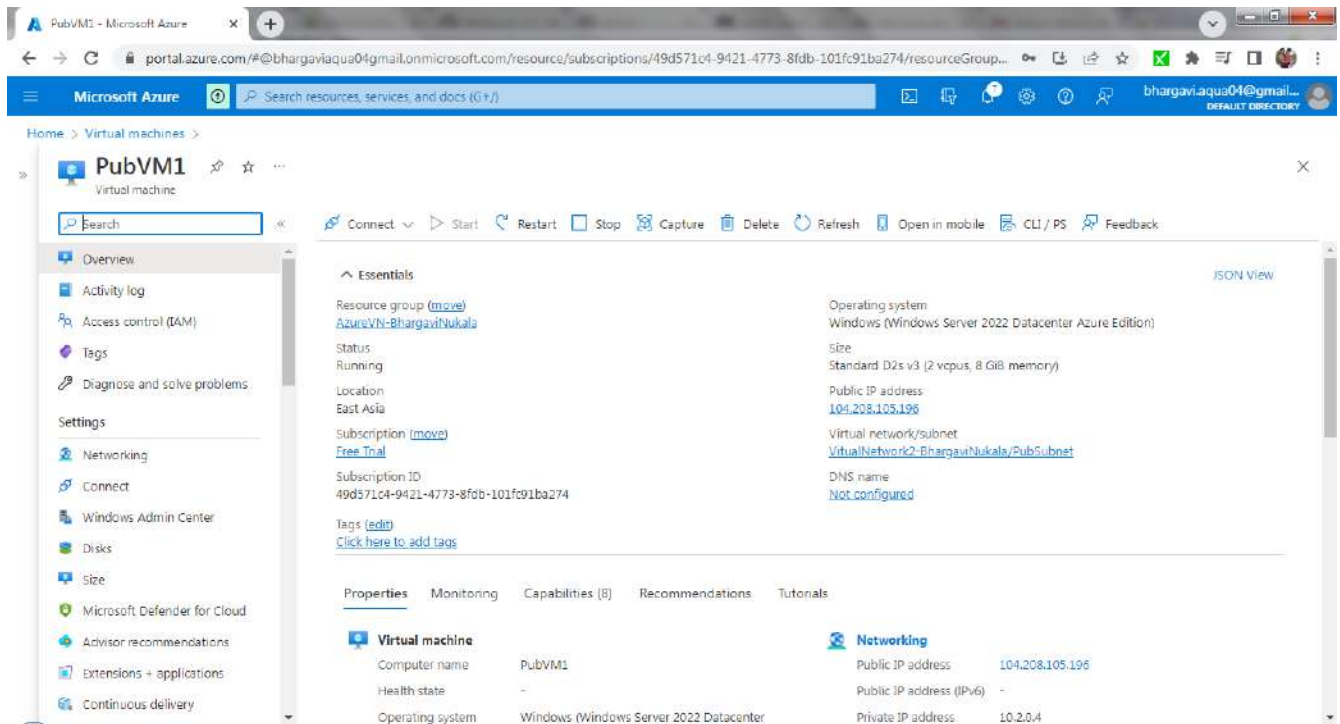
Now connect to private machine from public machine using remote desktop connection of public VM. Access google.com on private VM and the page is displayed.



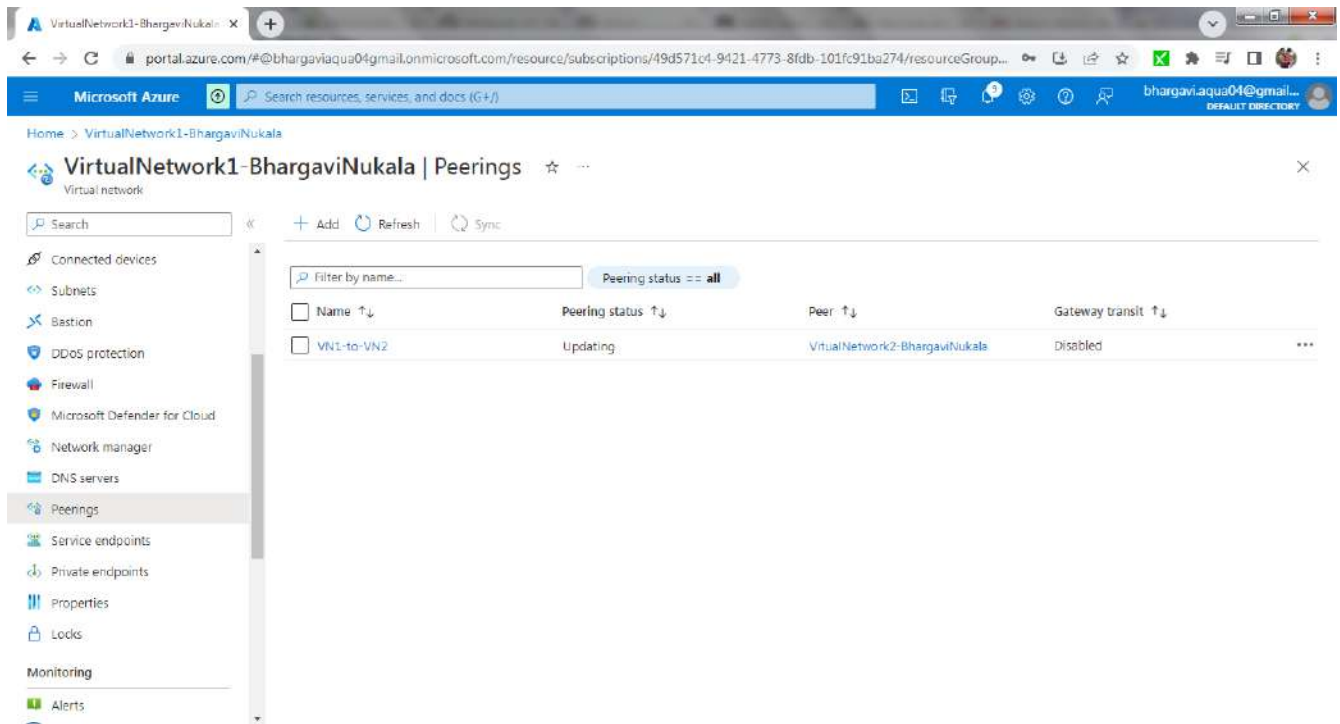
VPC Peering:
Create another Virtual Network in another region.



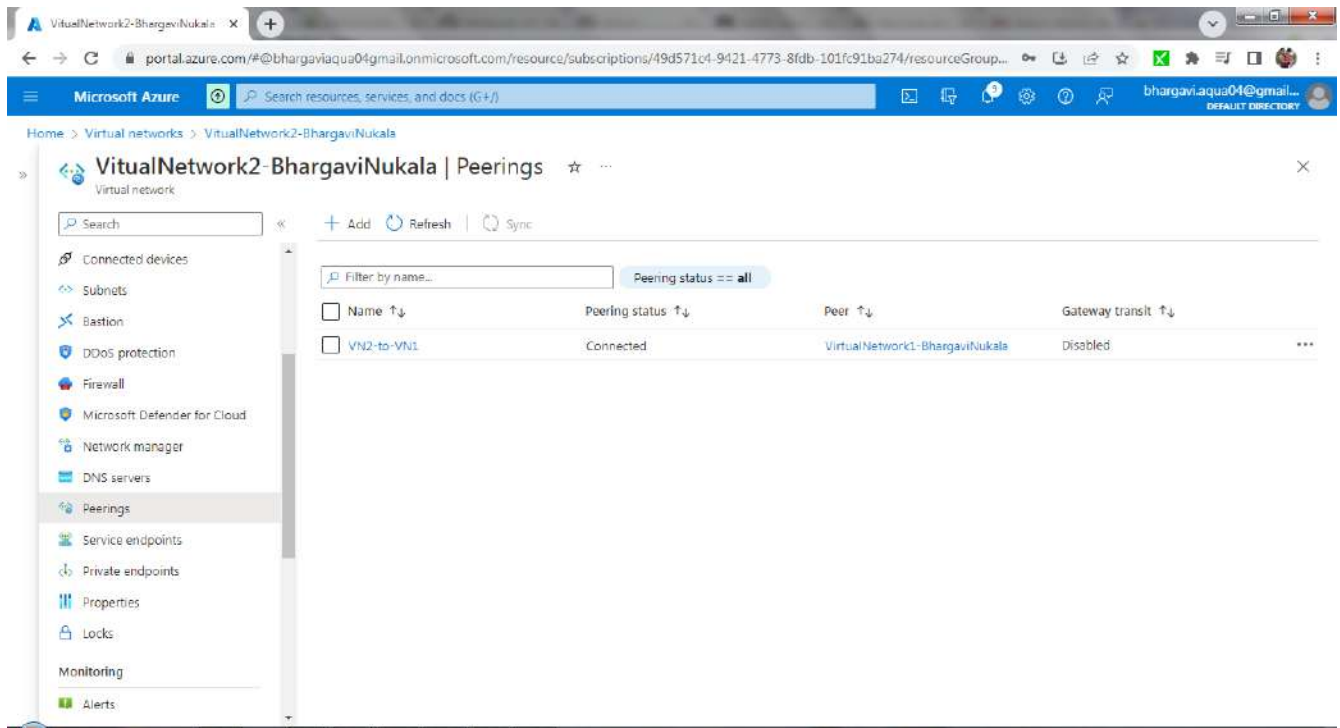
Now create a VM in Virtual Network2.



Add Peering in Virtual Network1. Goto VirtualNetwork1 and click on Peerings. Add peering between both Vnets. As seen, Two peerings are created between 2 Vnets.
VN1->VN2



VN2->VN1



Now a peering is established and when you use private IP of VM of VN2 to connect from VM of VN1, you should be able to connect.

