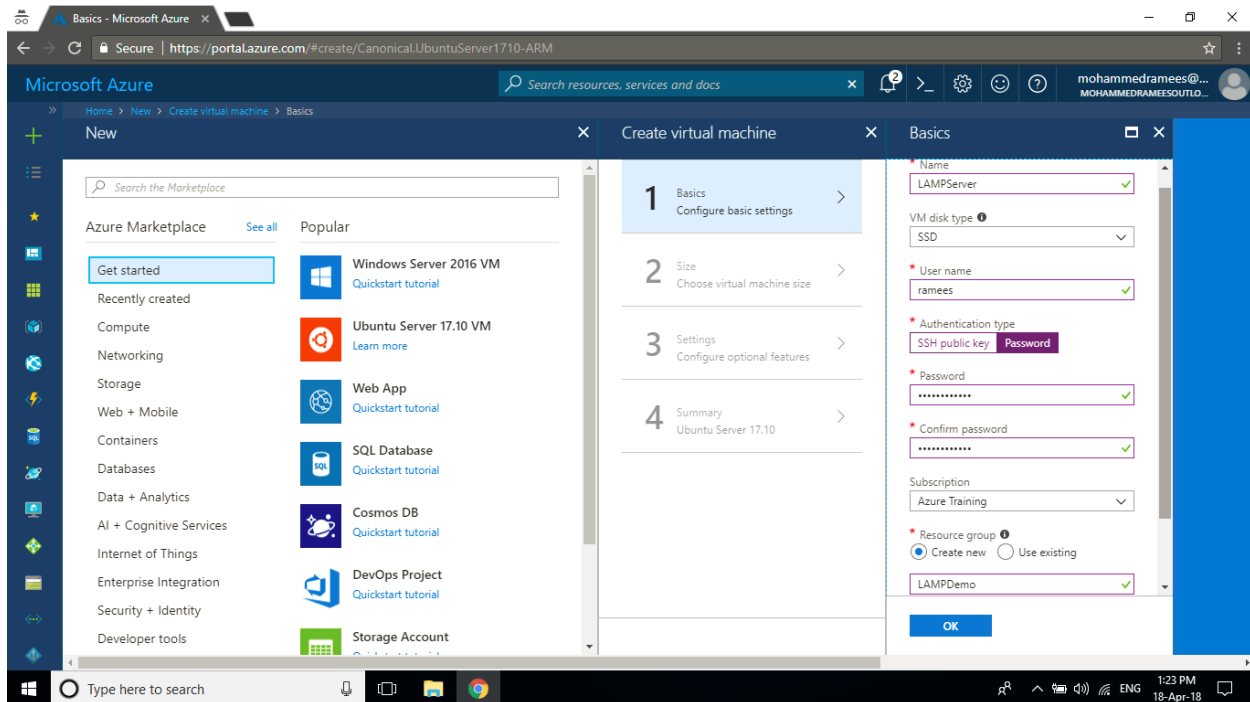
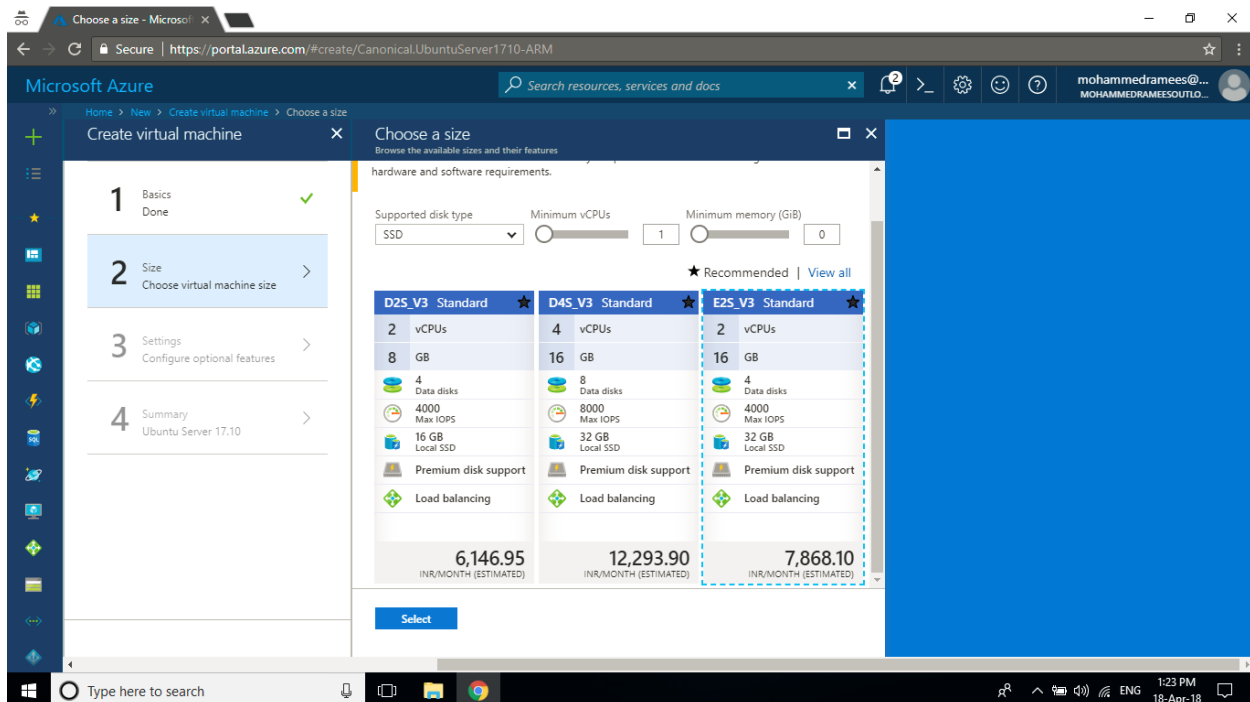


# Configuring a LAMP Server

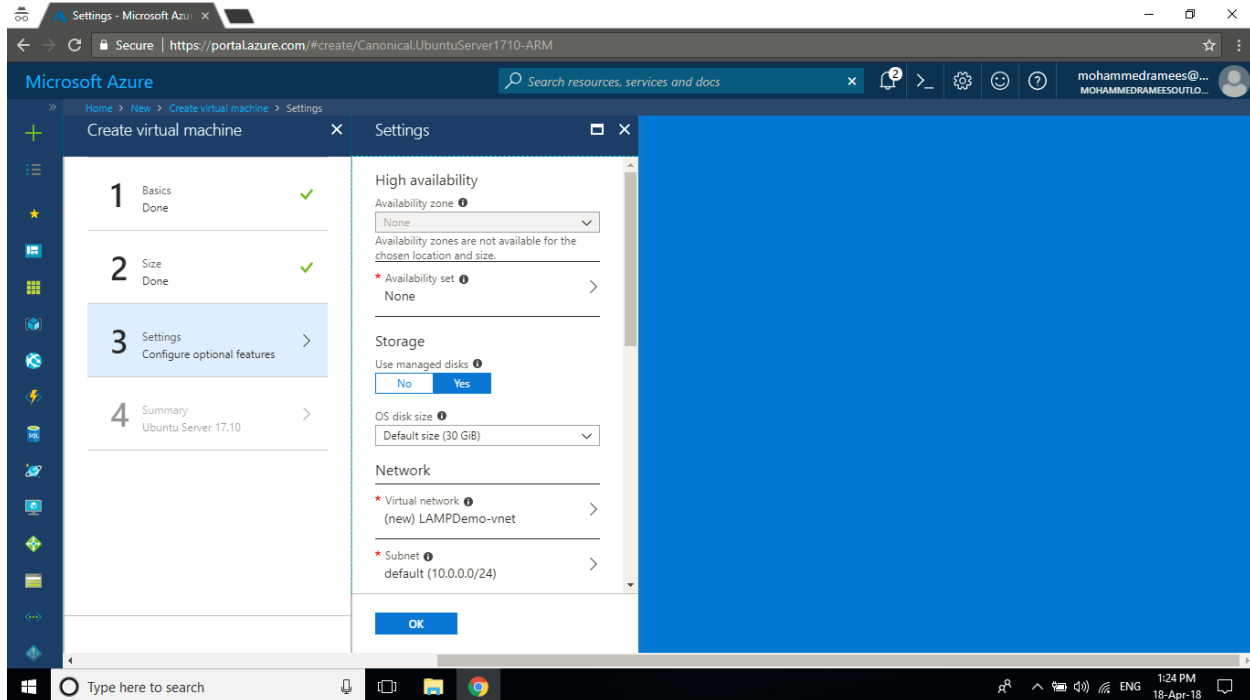
**Step 1:** Create a new **Ubuntu Server** from compute category. Fill the necessary details like **Name**, **Disk Type(SSD/HDD)**, **Username** and **Password** for login credentials, **Subscription** (if you have multiple), **Location** and put the associated resources in a new **Resource Group**



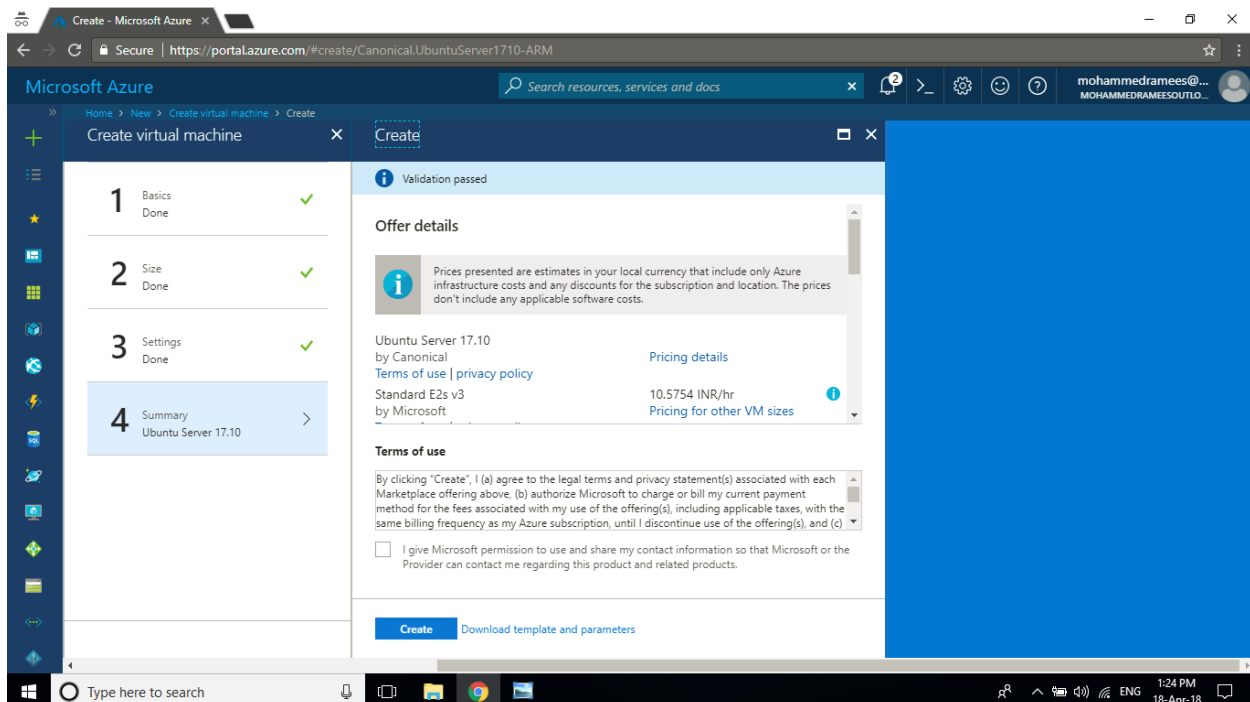
**Step 2:** In the second step select the **Size** of the required VM as per your business requirements. You can see what each size offers and how much it costs per month for the same.



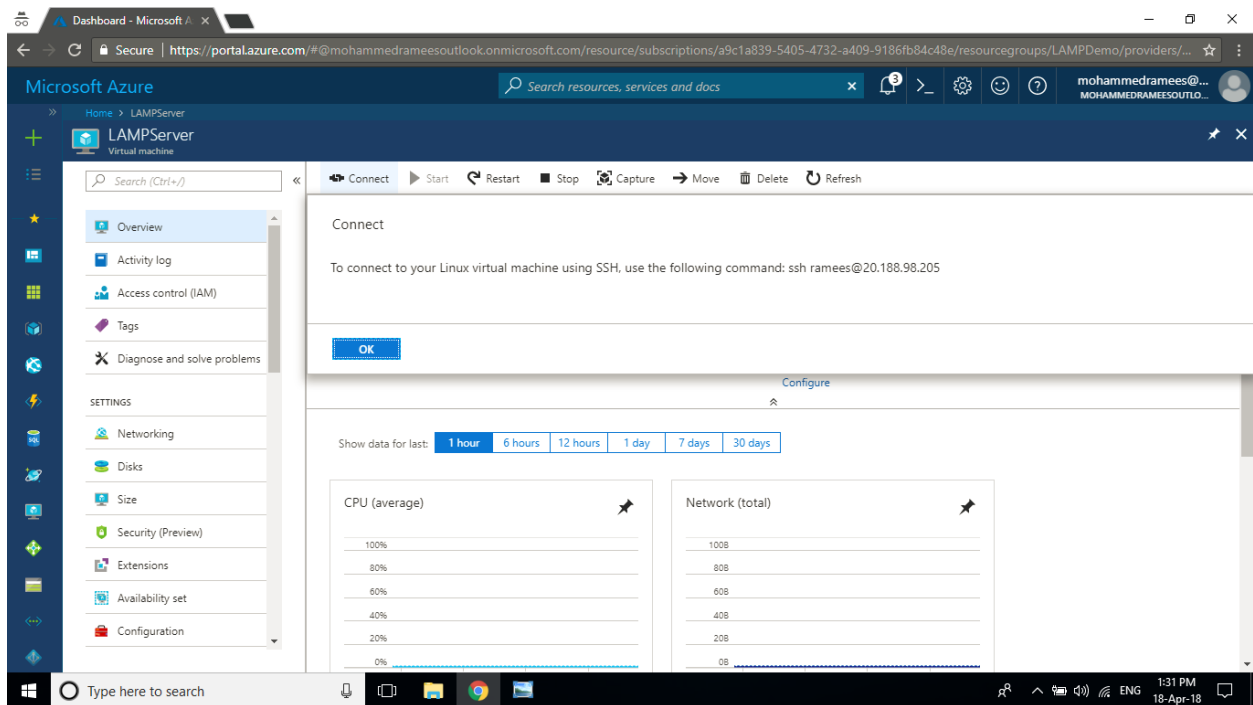
**Step 3:** In the third step you can specify the **Availability Zone** ( only available in some regions), **Availability Set** (if required), opt for **Managed** or **Unmanaged Disks** ( if unmanaged, have to create a new storage account also), **OS Disk size**, **Virtual Network** and **Subnet**, **Network Security Group**, **Public IP**, add any **Extensions**, enable or disable **Auto Shutdown**, **Boot Diagnostics**, **Guest OS Diagnostics**, **Backup** ( In this demo we are going with the defaults)



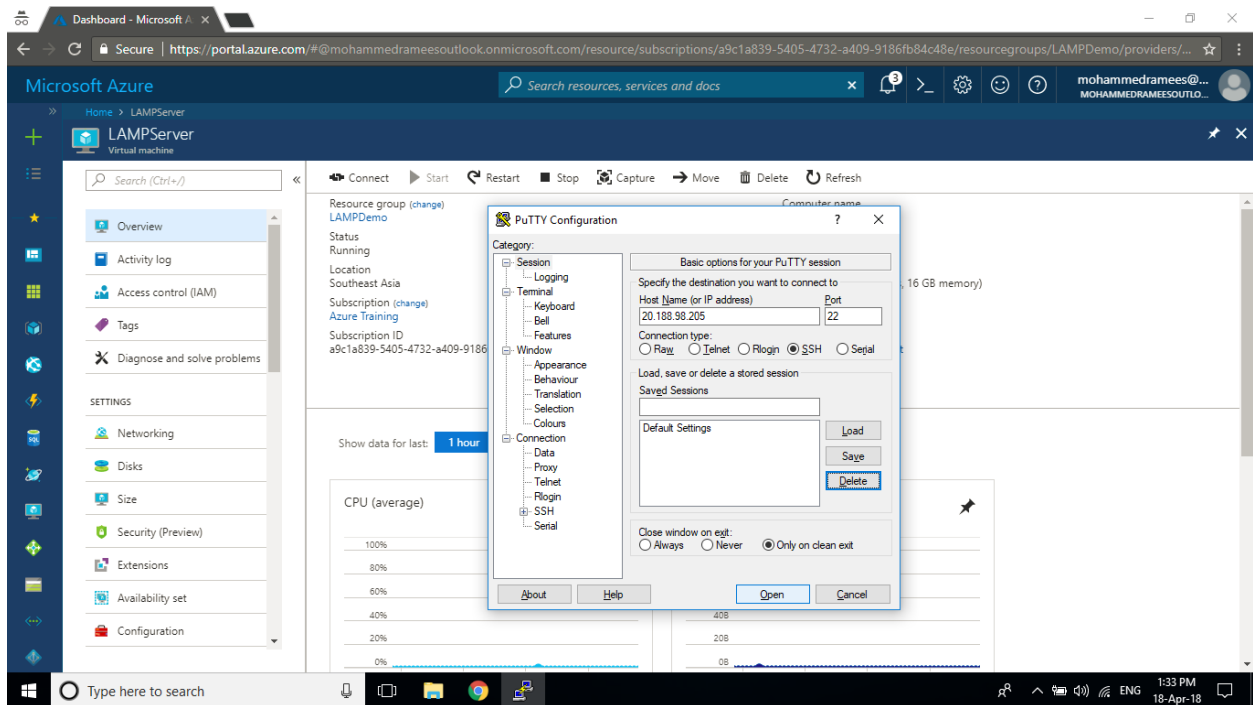
**Step 4:** On the final step wait for the final **Validation** to be passed and clicking on create will initiate the deployment of the VM. This may take few minutes and you can check the status from the notification bar.

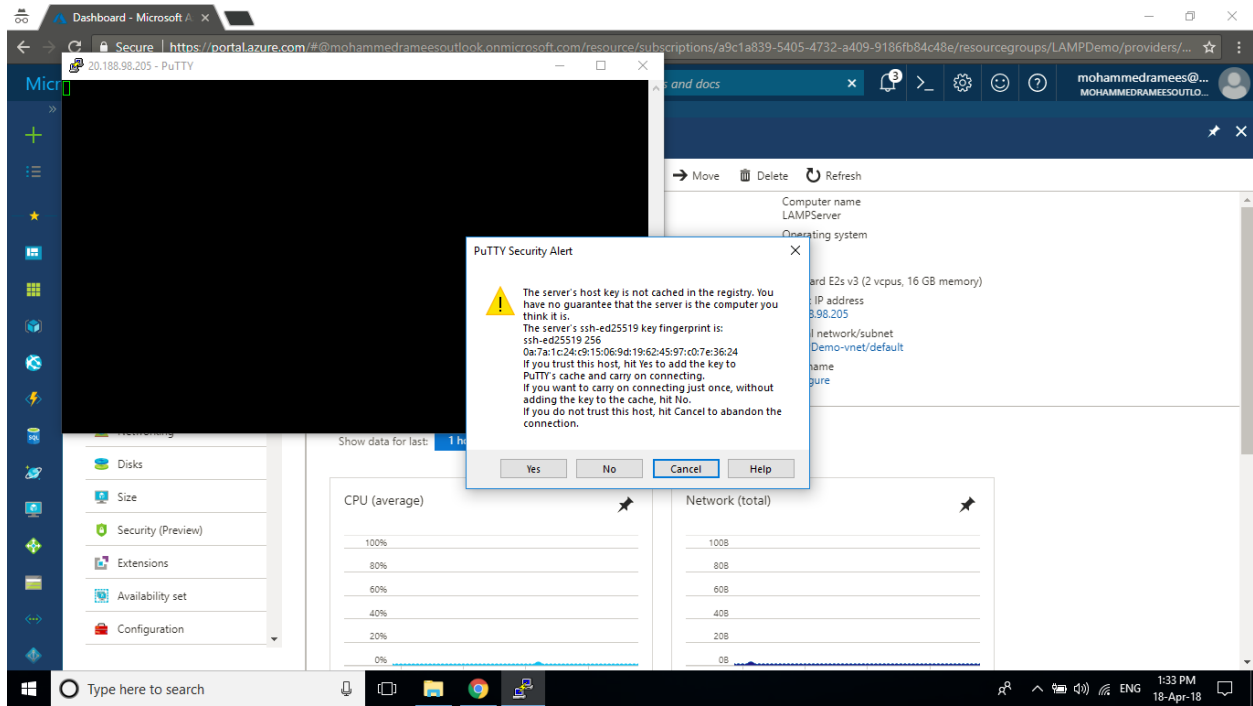


**Step 5:** Once the deployment is completed the dashboard for the new VM will be opened. The same you can access from the Virtual Machine listing or through Resource Group from the portal. Click on **Connect** option will show the command to connect via the terminal. Here we are using Putty to connect.

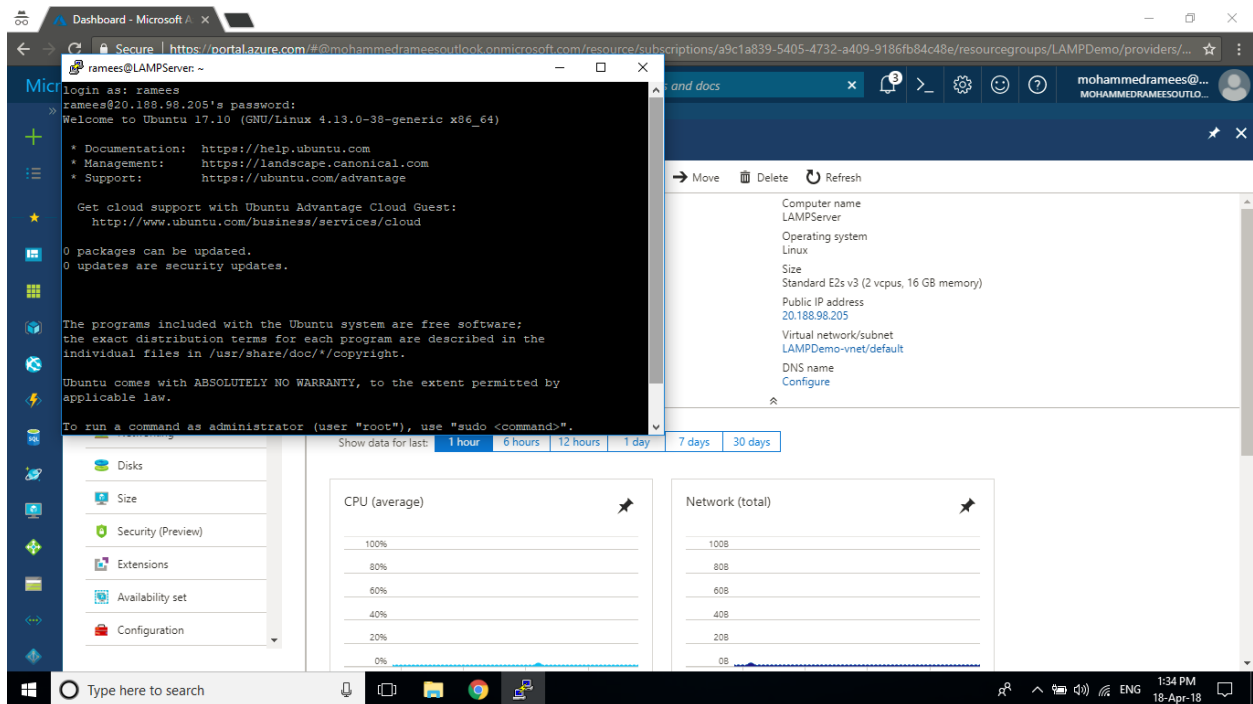


**Step 6:** Copy the Public IP address to the **Putty** and click on open. Click on yes to add the key to the Putty's cache as you are login in for the first time



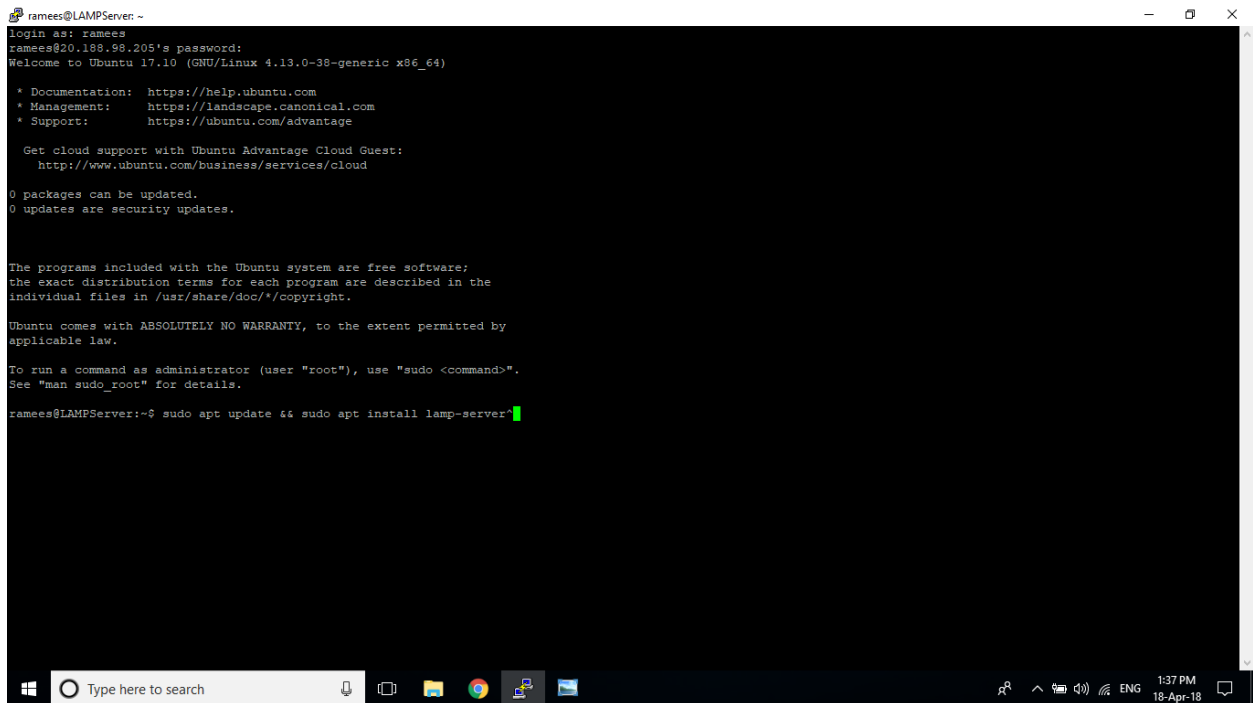


**Step 7:** Give the same **Username** and **Password** that you assigned while you created the VM. Now you have logged in to your new Ubuntu VM.



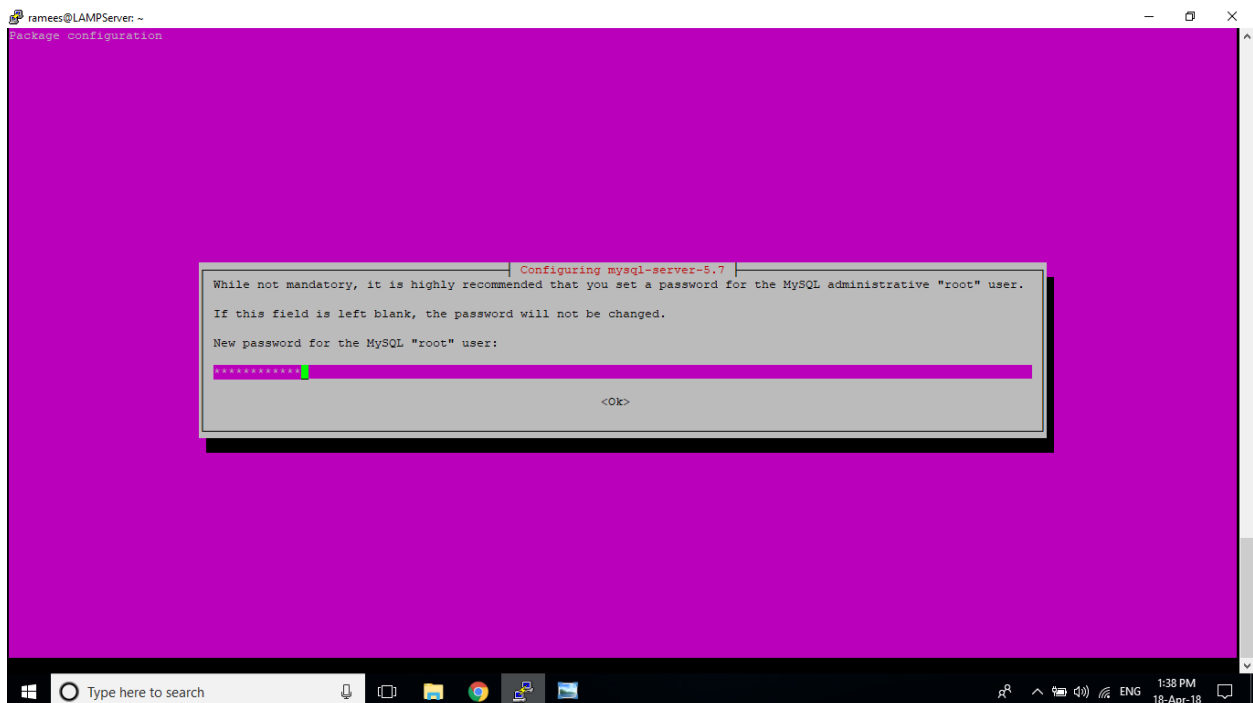
**Step 8:** Now we have **Ubuntu** and have to install **Apache**, **MySQL**, and **PHP**. Run the following command to update Ubuntu package sources and install the same. It may take few minutes and may ask for permission in some steps.

```
sudo apt update && sudo apt install lamp-server^
```



```
ramees@LAMPServer: ~  
login as: ramees  
ramees@20.188.98.205's password:  
Welcome to Ubuntu 17.10 (GNU/Linux 4.13.0-38-generic x86_64)  
  
 * Documentation:  https://help.ubuntu.com  
 * Management:    https://landscape.canonical.com  
 * Support:       https://ubuntu.com/advantage  
  
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.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ramees@LAMPServer:~$ sudo apt update && sudo apt install lamp-server^
```

**Step 9:** During the installation it will prompt for **Password** for **MySQL** database. Give the password and you may have to confirm your password.



```
ramees@LAMPServer: ~  
package configuration  
  
Configuring mysql-server-5.7  
While not mandatory, it is highly recommended that you set a password for the MySQL administrative "root" user.  
If this field is left blank, the password will not be changed.  
New password for the MySQL "root" user:  
*****  
<Ok>
```

**Step 10: Enable Port 80** by adding an inbound rule in networking tab from portal to allow HTTP requests as our VM is a Web Server. By default, only SSH connections are allowed into Linux VMs deployed in Azure.

The screenshot shows the Microsoft Azure portal interface. On the left, the 'LAMPServer - Networking' page is open, displaying a table of inbound port rules. On the right, the 'Add inbound security rule' dialog box is open, showing the configuration for a new rule.

**Network Interface: lampserver154**  
Virtual network/subnet: LAMPDemo-vnet/default Public IP: 20.188.98.205 Private IP: 10.0.0.4

**INBOUND PORT RULES**  
Network security group LAMPServer-nsg (attached to network interface: lampserver154)  
Impacts 0 subnets, 1 network interfaces

PRIORITY	NAME	PORT	PROTOCOL	SOURCE
1000	default-allow-ssh	22	TCP	Any
65000	AllowVnetInBound	Any	Any	VirtualNetwork
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer
65500	DenyAllInBound	Any	Any	Any

**OUTBOUND PORT RULES**  
Network security group LAMPServer-nsg (attached to network interface: lampserver154)  
Impacts 0 subnets, 1 network interfaces

**Add inbound security rule**  
LAMPServer-nsg

Service: HTTP  
Port ranges: 80  
Priority: 1010  
Name: HTTP  
Description:

OK

The screenshot shows the Microsoft Azure portal interface. On the left, the 'LAMPServer - Networking' page is open, displaying a table of inbound port rules. On the right, the 'Add inbound security rule' dialog box is open, showing the configuration for a new rule.

**Network Interface: lampserver154**  
Virtual network/subnet: LAMPDemo-vnet/default Public IP: 20.188.98.205 Private IP: 10.0.0.4

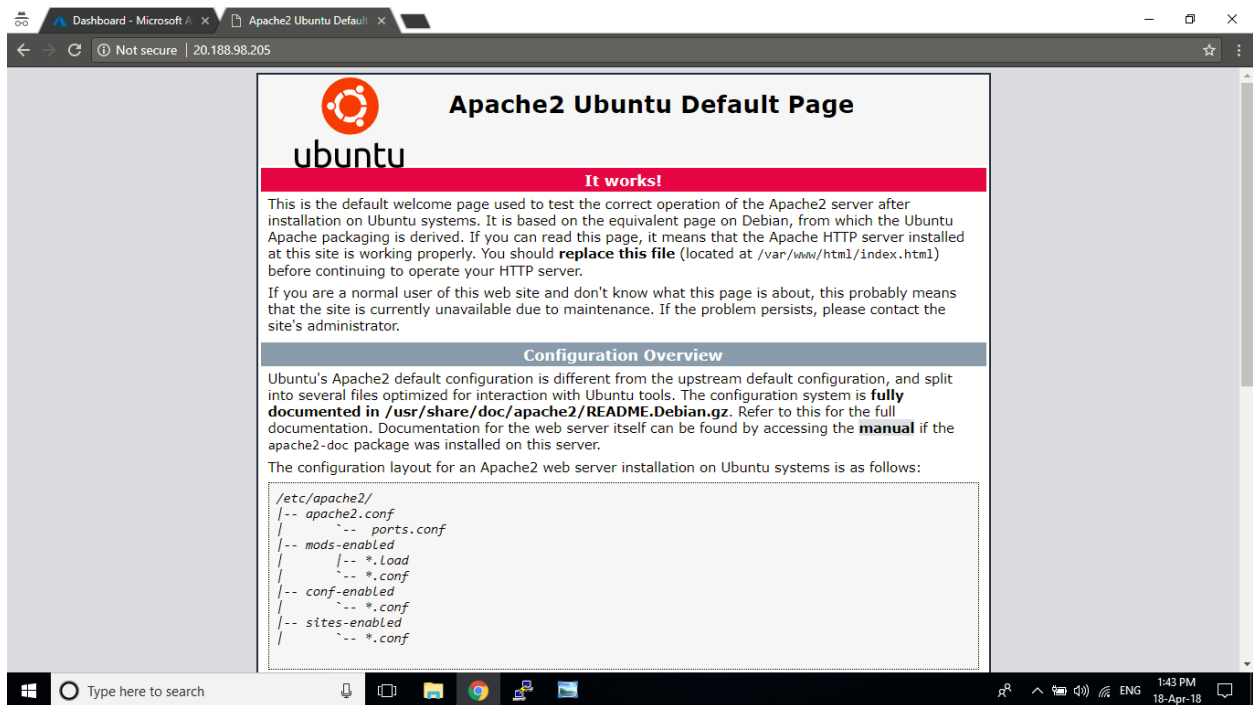
**INBOUND PORT RULES**  
Network security group LAMPServer-nsg (attached to network interface: lampserver154)  
Impacts 0 subnets, 1 network interfaces

PRIORITY	NAME	PORT	PROTOCOL	SOURCE	DESTINATION	ACTION
1000	default-allow-ssh	22	TCP	Any	Any	Allow
1010	HTTP	80	TCP	Any	Any	Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny

**OUTBOUND PORT RULES**  
Network security group LAMPServer-nsg (attached to network interface: lampserver154)  
Impacts 0 subnets, 1 network interfaces

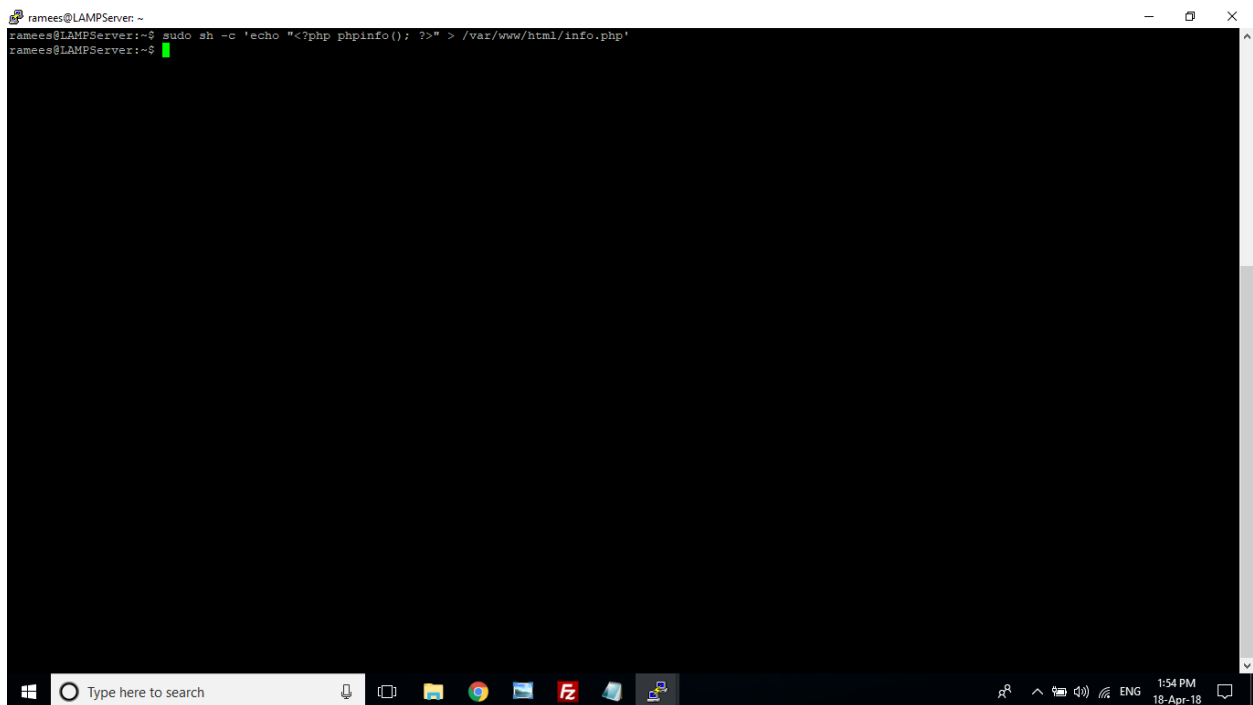
**Add inbound port rule**

**Step 11:** With Apache installed, and port 80 open to your VM, the web server can now be accessed from the internet. To view the Apache2 Ubuntu Default Page, open a web browser, and enter the public IP address of the VM.

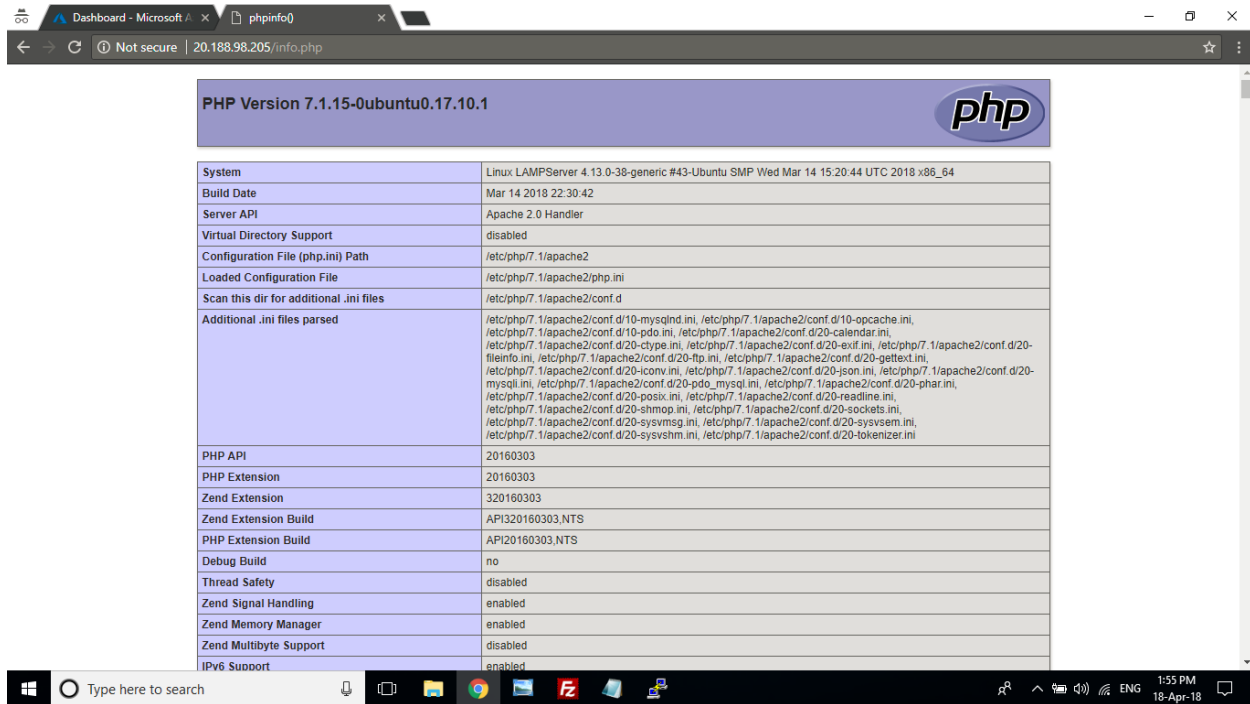


**Step 12:** To test PHP, create a quick PHP info page to view in a browser by using the following command in the terminal

```
sudo sh -c 'echo "<?php phpinfo(); ?>" > /var/www/html/info.php'
```



**Step 13:** You can verify the existence of the PHP page that you just created by visiting [http://\[yourPublicIPAddress\]/info.php](http://[yourPublicIPAddress]/info.php) in a browser and it should show a similar page as below.



PHP Version 7.1.15-0ubuntu0.17.10.1

System	Linux LAMPServer 4.13.0-38-generic #43-Ubuntu SMP Wed Mar 14 15:20:44 UTC 2018 x86_64
Build Date	Mar 14 2018 22:30:42
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.1/apache2
Loaded Configuration File	/etc/php/7.1/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.1/apache2/conf.d
Additional .ini files parsed	/etc/php/7.1/apache2/conf.d/10-mysqld.ini, /etc/php/7.1/apache2/conf.d/10-opcache.ini, /etc/php/7.1/apache2/conf.d/10-pdo.ini, /etc/php/7.1/apache2/conf.d/20-calendar.ini, /etc/php/7.1/apache2/conf.d/20-ctype.ini, /etc/php/7.1/apache2/conf.d/20-exif.ini, /etc/php/7.1/apache2/conf.d/20-fileinfo.ini, /etc/php/7.1/apache2/conf.d/20-ftp.ini, /etc/php/7.1/apache2/conf.d/20-gettext.ini, /etc/php/7.1/apache2/conf.d/20-iconv.ini, /etc/php/7.1/apache2/conf.d/20-json.ini, /etc/php/7.1/apache2/conf.d/20-mysqli.ini, /etc/php/7.1/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.1/apache2/conf.d/20-phar.ini, /etc/php/7.1/apache2/conf.d/20-posix.ini, /etc/php/7.1/apache2/conf.d/20-readline.ini, /etc/php/7.1/apache2/conf.d/20-shmop.ini, /etc/php/7.1/apache2/conf.d/20-sockets.ini, /etc/php/7.1/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.1/apache2/conf.d/20-sysvsem.ini, /etc/php/7.1/apache2/conf.d/20-sysvshm.ini, /etc/php/7.1/apache2/conf.d/20-tokenizer.ini
PHP API	20160303
PHP Extension	20160303
Zend Extension	320160303
Zend Extension Build	API320160303.NTS
PHP Extension Build	API20160303.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled