

Module 3: Implementing Azure Resource Manager Virtual Machines

Demo – Solution

edureka!

edureka!

➤ Creating a Windows Virtual Machine

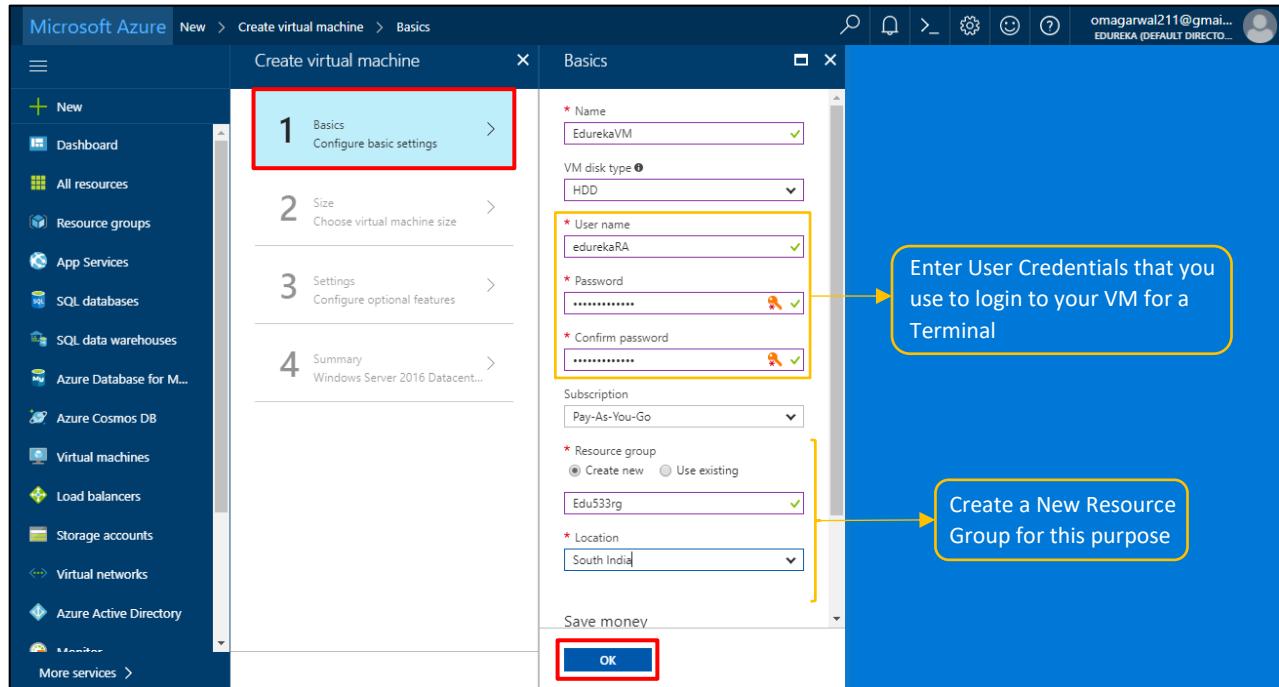
Step 1: In the Portal, Goto: +New > Compute > Select a Windows Server of your choice:

The screenshot shows the Microsoft Azure portal's 'New' blade. On the left, a sidebar lists various service categories like Dashboard, All resources, Resource groups, etc. A red box highlights the '+ New' button and the 'Compute' category under it. The main area shows a 'Featured' section with several options: 'Windows Server 2016 Datacenter' (highlighted with a green box), Red Hat Enterprise Linux 7.2, Ubuntu Server 16.04 LTS, SQL Server 2016 SP1 Enterprise on Windows Server 2016, Virtual machine scale set, Service Fabric Cluster, Web App for Containers, and Function App.

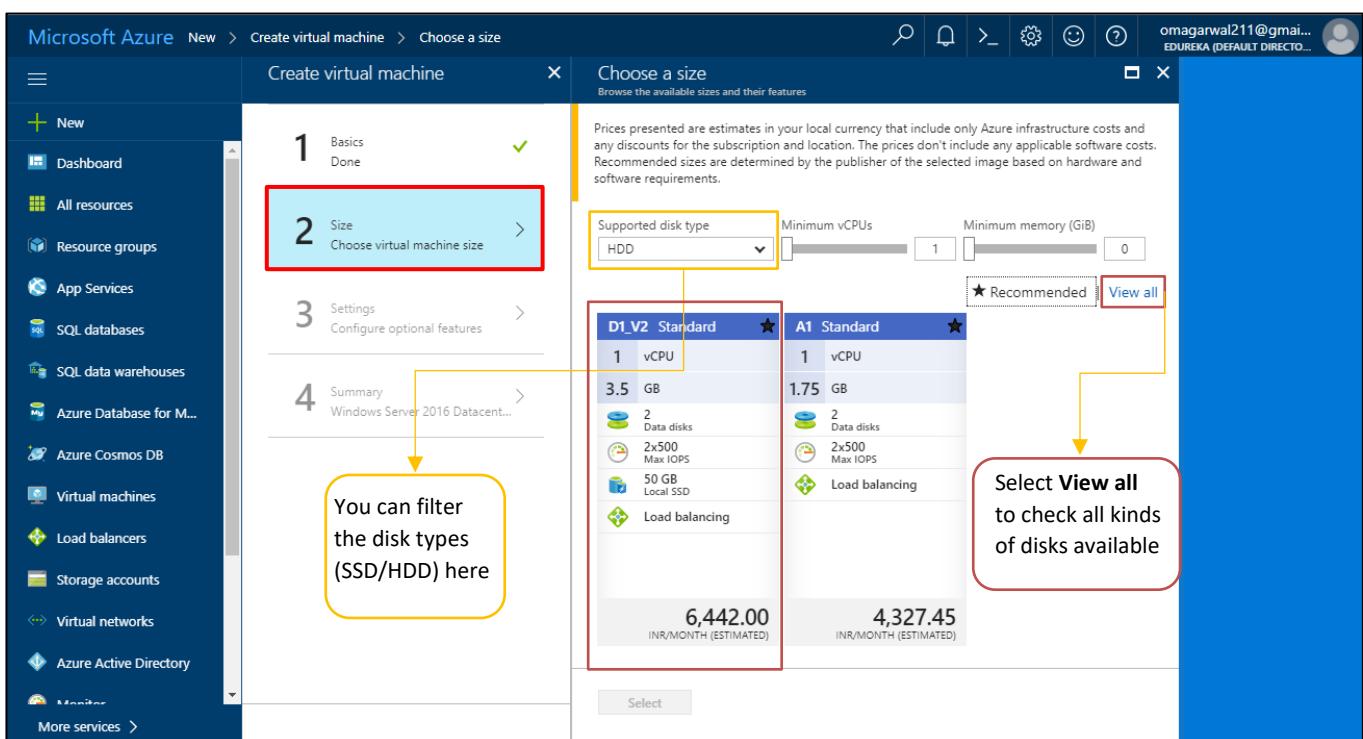
Step 2: In the VM Creation **Basic** blade, Enter the VM name and select the desired disk type as shown below:

The screenshot shows the 'Create virtual machine' Basics blade. On the left, a vertical navigation bar lists steps: 1. Basics (Configure basic settings), 2. Size (Choose virtual machine size), 3. Settings (Configure optional features), and 4. Summary (Windows Server 2016 Datacenter). The main area shows fields for Name (EdurekaVM), VM disk type (HDD selected), Password, Confirm password, Subscription (Free Trial), Resource group (Create new selected), and Location (South India). A large yellow callout box points to the 'Name' field with the text: 'The name of a VM can be up to 15 characters'. The 'OK' button at the bottom right is highlighted with a red box.

Step 3: After selecting the Disk type, Enter the Credentials for VM server login and other select the desired options:



Step 4: In the VM Size blade, select the desired VM size from the list of different tier listed below



Step 5: In the VM settings blade, you can select an existing Availability set or create one (discussed in Mod-4 in detail):

High availability

- * Availability set ⓘ
- None

Storage

Network

Extensions

OK

- Availability sets provide redundancy to your application
- It is recommended that you group two or more virtual machines in an availability set
- This configuration ensures that during either a **planned** or **unplanned maintenance event**, at least one virtual machine will be available

Planned maintenance events are periodic updates made by Microsoft to the underlying Azure platform to improve overall reliability, performance, and security of the platform infrastructure that your VMs run on

Unplanned maintenance events occur when the hardware or physical infrastructure underlying your VM has faulted because of local network failures, local disk failures, or other rack level failures

Step 6: To create an Availability Set, click on Availability set > Create new > Fill in the details and click **OK**:

Settings

High availability

* Availability set ⓘ

None

Storage

Network

Extensions

OK

Change availability set

Create new

+ Create new

None

Name

Edu533aset1

Fault domains ⓘ

Update domains ⓘ

Use managed disks ⓘ

No (Classic) Yes (Aligned)

Update domains

This ensures that, during any planned/unplanned maintenance event, at least One VM will be available and meet the 99.95% Azure SLA

Fault domains

Step 7: In the Settings blade, select the options for Managed disks as per your requirements:

The screenshot shows the 'Settings' blade for creating a virtual machine. In the 'Storage' section, the 'Use managed disks' dropdown is set to 'Yes'. A yellow box highlights this setting. To the right, a yellow callout bubble contains the following text:

- **Managed Disks** handles the storage account creation/management in the background for you
- It ensures that you do not have to worry about the scalability limits of the storage account
- You simply specify the disk size and the performance tier (Standard/Premium), and Azure creates and manages the disk for you
- Even as you add disks or scale the VM up and down, you don't have to worry about the storage being used

Step 8: In the Settings blade, you can either select any existing VNets/Subnets/PIP/NSG or Create new ones here:

The screenshot shows the 'Settings' blade for creating a virtual machine. In the 'Network' section, several network components are listed: Virtual network (new Edu533rg-vnet), Subnet (default 10.0.0.0/24), Public IP address (new EdurekaVM-ip), and Network security group (firewall) (new EdurekaVM-nsg). A yellow box highlights the 'Virtual network' entry. A yellow callout bubble on the right side contains the following text:

Azure assigns default VNet, Subnets, PIP and NSG for the VM, or you can even select existing/create new ones by clicking on those options

Step 9: In the VM setting blade, you can add any type of available **Extensions** by clicking on the option:

Azure VM extensions are **small applications** that provide post-deployment configuration and automation tasks on Azure VMs

For example:

- Apply PowerShell Desired State configurations to a virtual machine by using the DSC extension for Windows, which we will discuss briefly in Module 4
- You can even add tools such as antivirus, security, etc

Step 10: At the end of the VM creation, let us configure the monitoring of the resources and click **OK**:

While booting the VM, there can be many reasons why that VM gets into a non-bootable state

The Boot diagnostics feature enable you to easily diagnose and recover your VMs from boot failures

Create a new Diagnostics storage account in order to store all the diagnostic log data by mentioning the resource group

Step 11: You will be directed to the **Summary** page after successful validation and below is the summary of our VM, click **Create**:

The screenshot shows the Microsoft Azure 'Create virtual machine' wizard. The steps are numbered 1 through 4. Step 4 is highlighted in blue and labeled 'Summary Windows Server 2016 Datacenter...'. A callout box points to this step with the text: 'All the resources along with their parameters configured and deployed can be downloaded in the form of a JSON template file'. To the right of the summary, there is a detailed view of the configuration settings. At the bottom right of the summary section, there is a red box around the 'Create' button. Below the summary, there is a 'Terms of use' section with a checked checkbox and a 'Download template and parameters' button.

Setting	Value
Subscription	Pay-As-You-Go (new) Edu533rg
Resource group	Edu533rg
Location	South India
Computer name	EdurekaVM
Disk type	HDD
User name	edurekaRA
Size	Standard D1 v2
Managed	Yes
Virtual network	(new) Edu533rg-vnet
Subnet	(new) default (10.0.0.0/24)
Public IP address	(new) EdurekaVM-ip
Network security group (firewall)	(new) EdurekaVM-nsg
Availability set	None

Step 12: Once the new VM is deployed and running, you will be redirected to the Overview window as shown below:

The screenshot shows the Microsoft Azure 'EdurekaVM' Overview page. On the left, there is a navigation menu with various service icons. The main area displays the VM details under the 'Overview' tab. It shows the resource group (Edu533rg), status (Running), location (South India), subscription (Pay-As-You-Go), and subscription ID (a3eee433-bf80-45b8-acae-b581d33f6a40). Below this, there are two performance charts: 'CPU (average)' and 'Network (total)'. The CPU chart shows usage peaking around 3:30 PM. The Network chart shows traffic peaking around the same time, with labels for 'NETWORK IN' and 'NETWORK OUT'.

➤ Connecting to a Windows VM

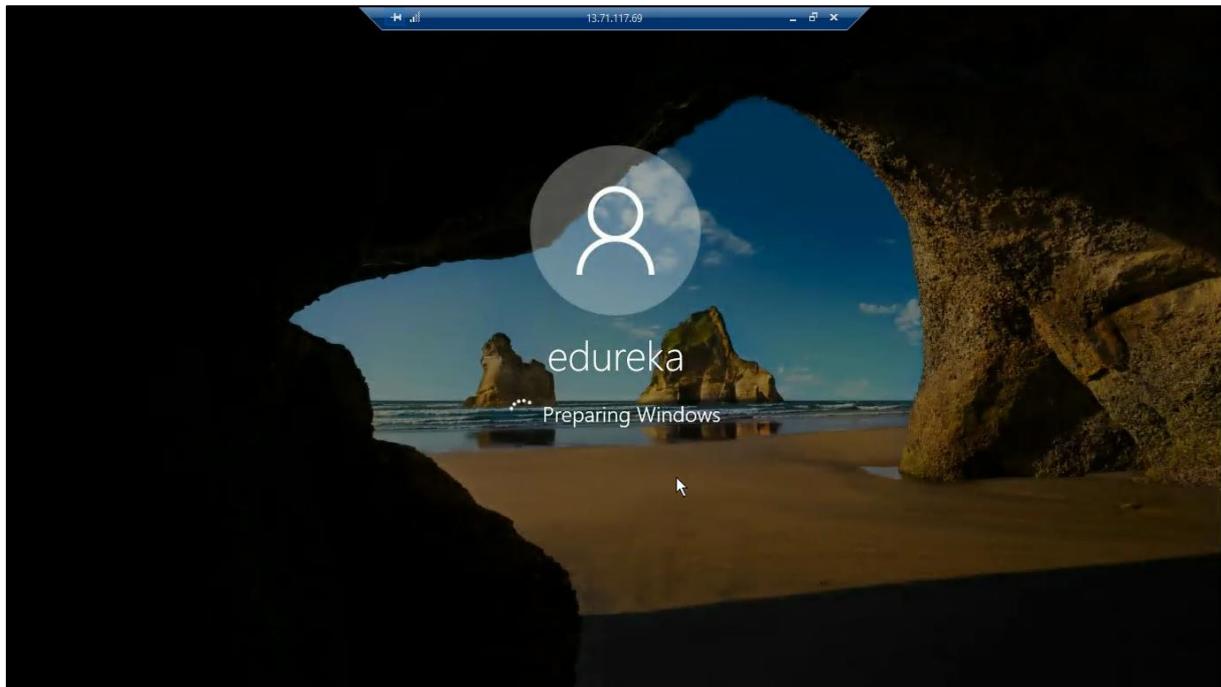
Step 1: In the VM Overview window, Click on the **Connect** option on the top to download the RDP file:

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various service icons like New, Dashboard, All resources, etc. The main area is titled 'EdurekaVM' and shows the 'Overview' tab selected. At the top, there are several action buttons: Connect, Start, Restart, Stop, Capture, Move, Delete, and Refresh. Below these are detailed resource settings: Resource group (Edureka), Status (Running), Location (South India), Subscription (Pay-As-You-Go), and various performance metrics like CPU and Network usage over time. At the bottom left, there's a download link for 'EdurekaVM.rdp'.

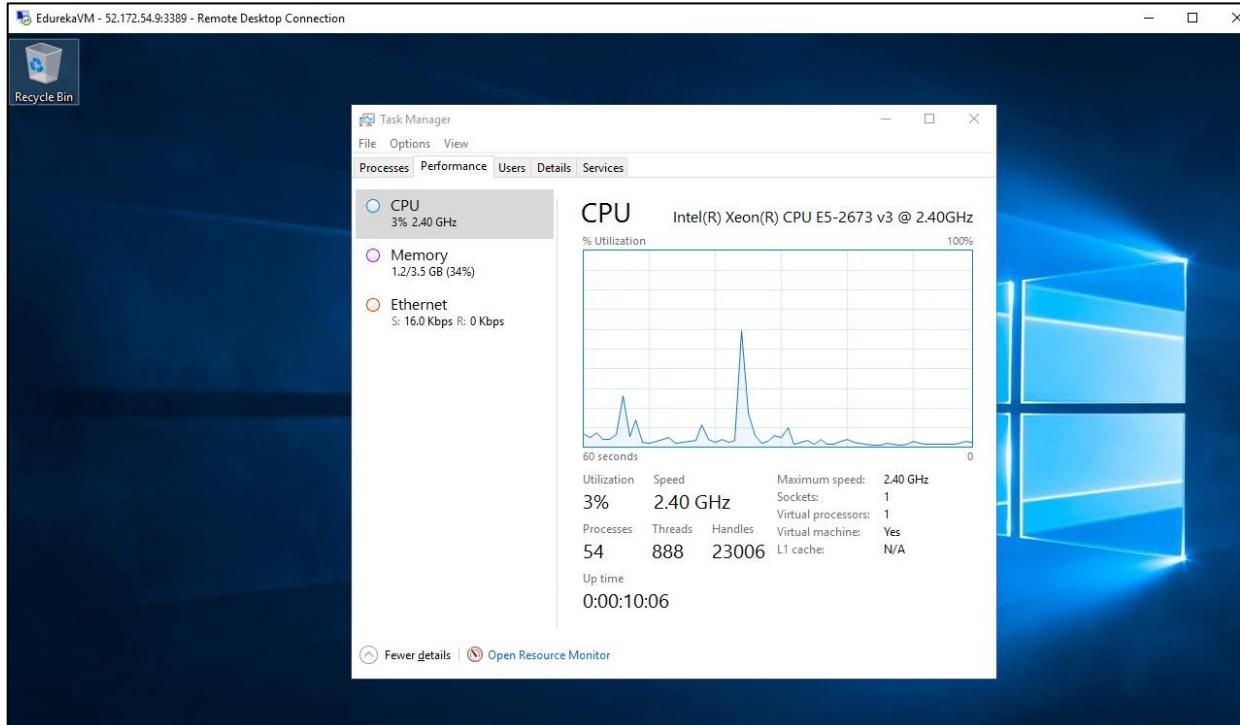
Step 2: Once downloaded, Open the RDP file > Allow Connection > Enter the login Credentials:

This screenshot shows the same Azure portal interface as the previous one, but with a 'Windows Security' dialog box overlaid. The dialog box is titled 'Enter your credentials' and contains fields for 'Username', 'Password', and 'Domain', along with a 'Remember me' checkbox. The 'OK' and 'Cancel' buttons are at the bottom. The background shows the same VM overview details as the first screenshot.

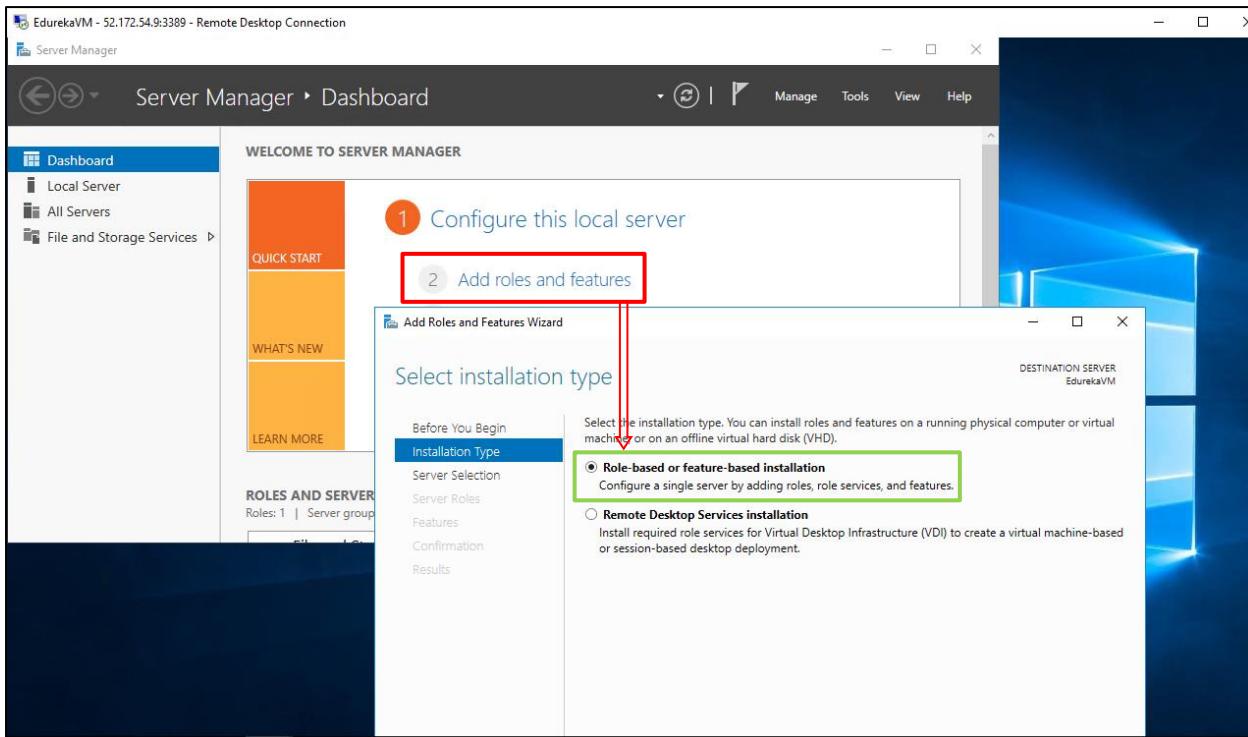
Step 3: Once you login with the Credentials, the Remote Desktop Connection to the Virtual Machine is successfully established:



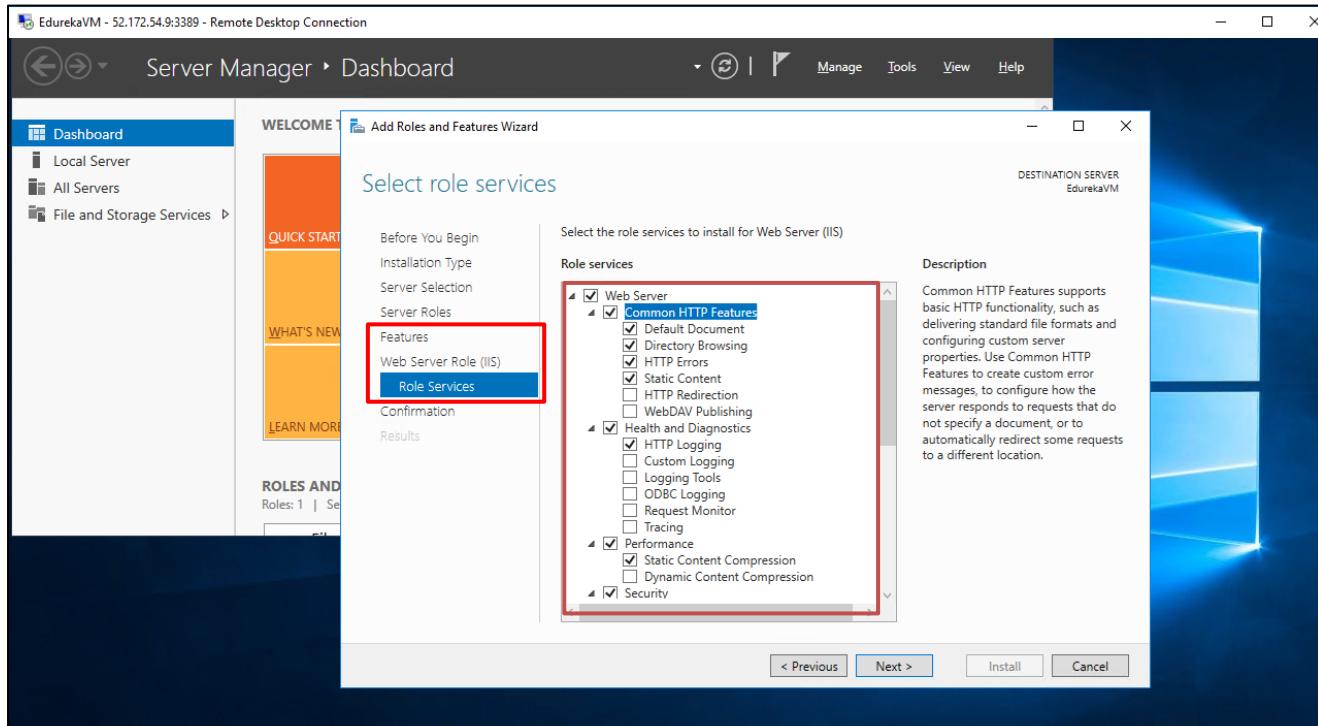
Step 4: In the New VM, Goto: Start Menu > Find **Task Manager** > Performance > Check **Utilization** and **Throughput**:



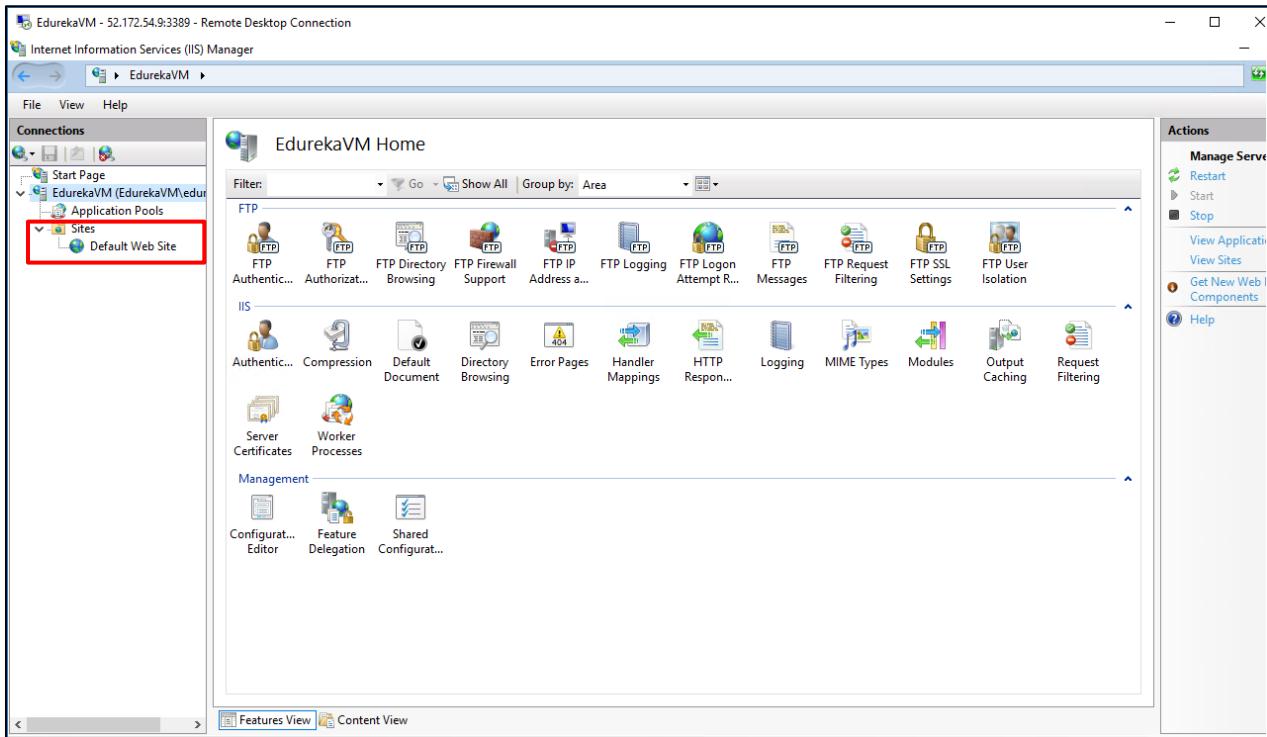
Step 5: Open the Task Manager from the Start Menu > Select Add Roles > Click Next > Select the desired Type:



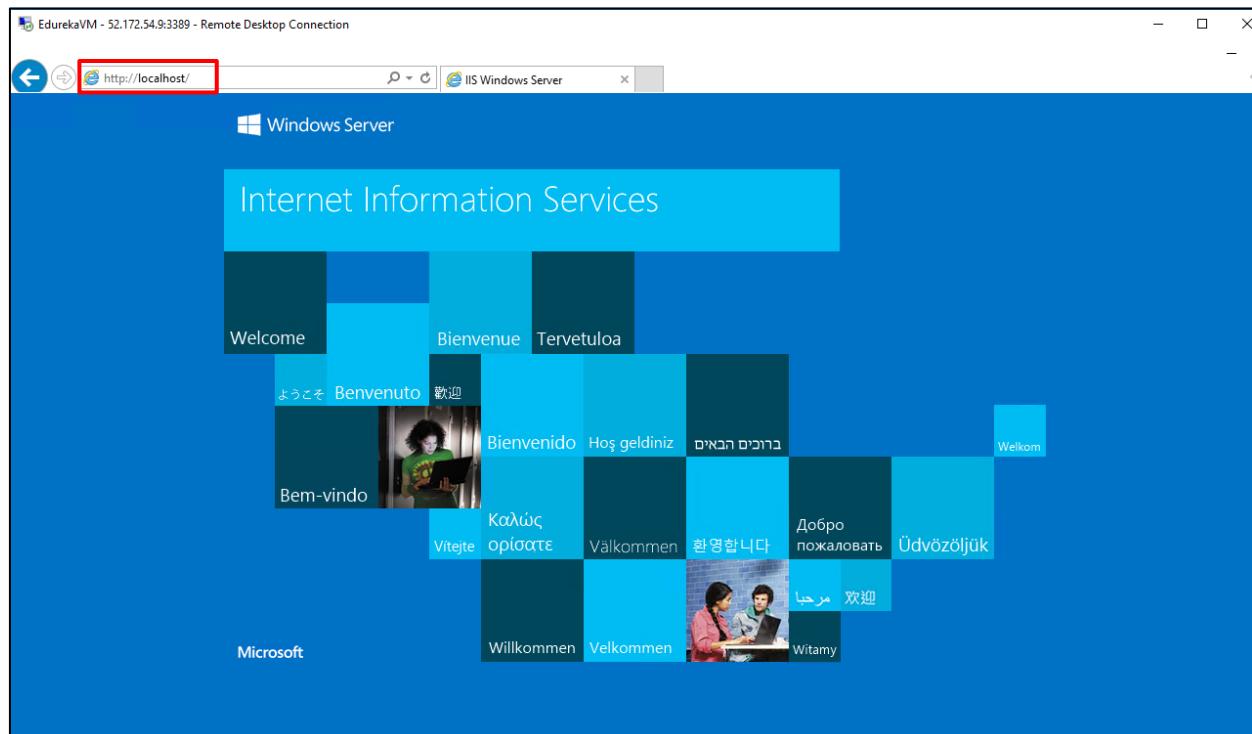
Step 6: After selecting installation type, Select the desired Server (**IIS Web**) > Select the **Role Services** > Next > Install:



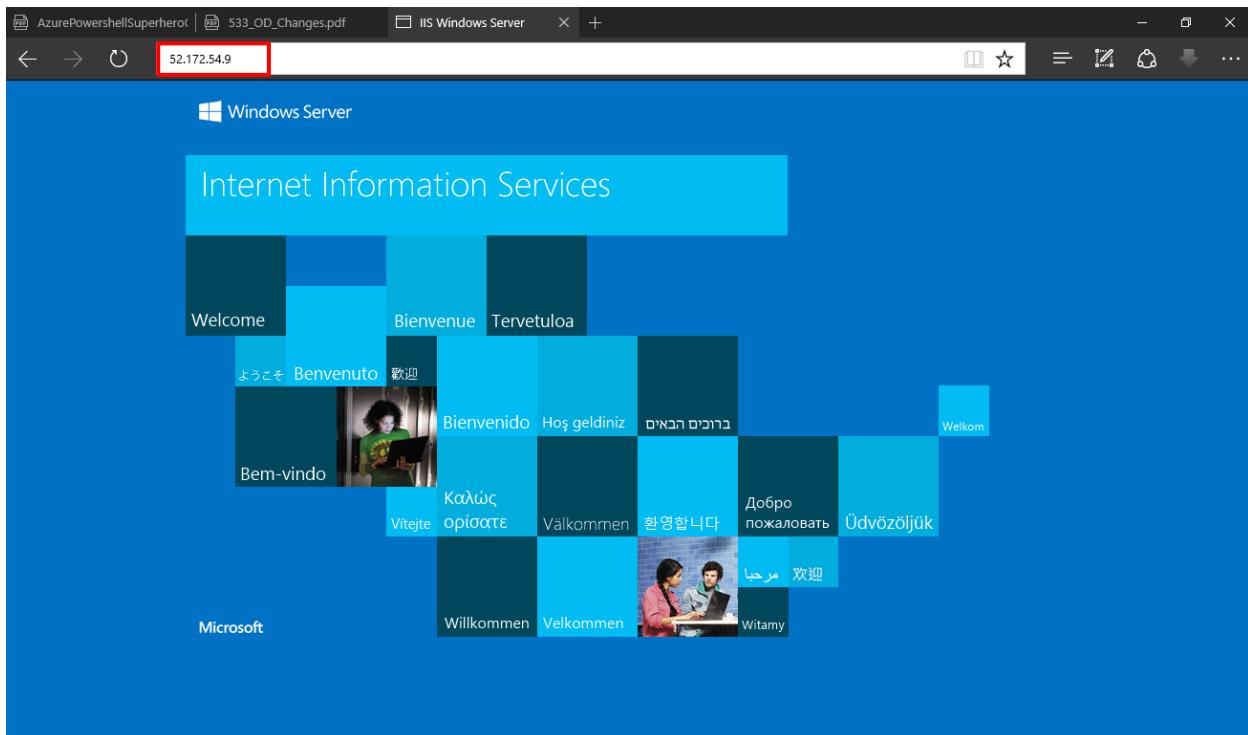
Step 7: Once the Installation is complete, Goto Start Menu > Open IIS Manager > Under your VM, Check the website:



Step 8: Open a browser > Enter “localhost” in the URL address to connect to IIS website internally:



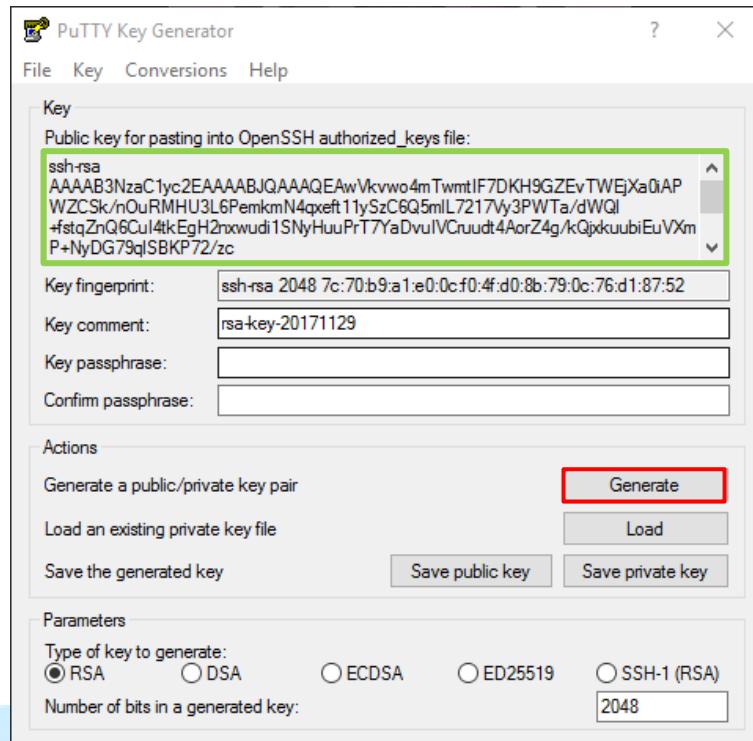
Step 9: To connect to the IIS from any other devices, Enter the **Public IP Address** of the VM in the Browser:



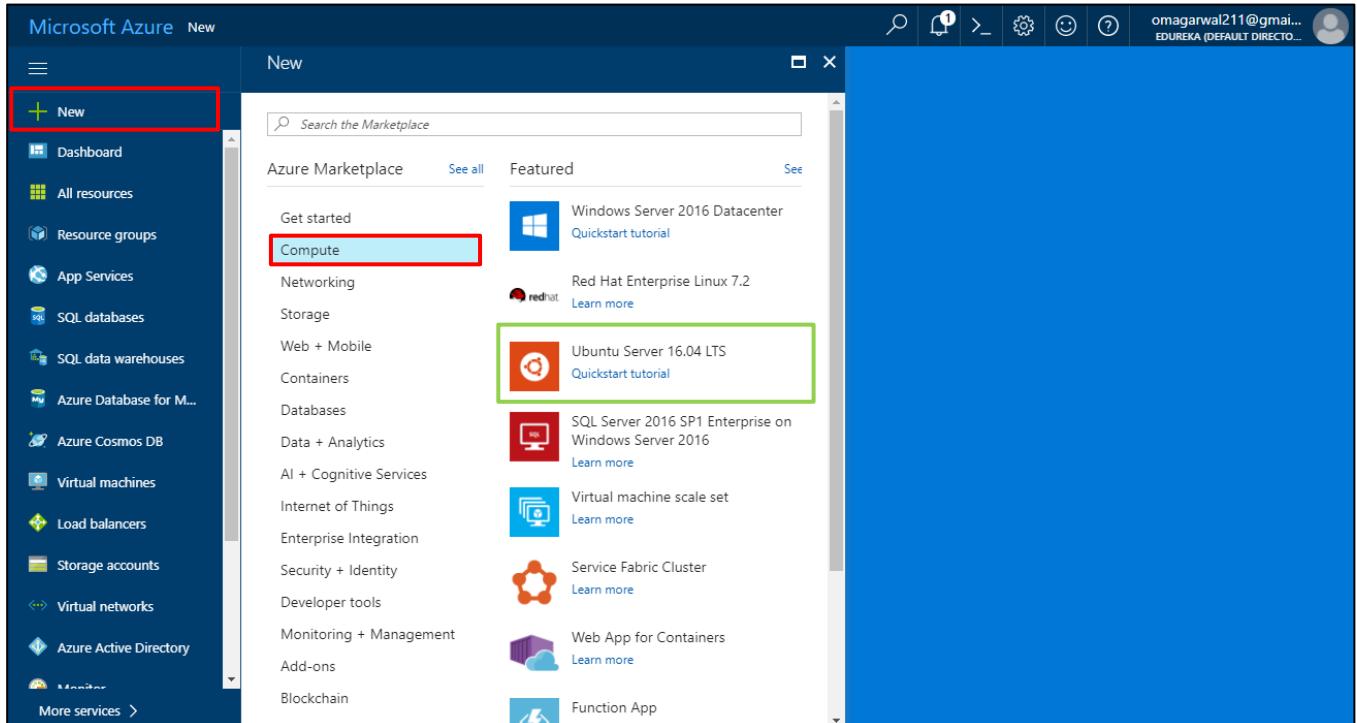
edureka!

➤ Creating a LINUX VM using Azure Portal

Step 1: Once you have downloaded the Putty tool, Open the PuttyGen on your system
 > Click on **Generate** option:



Step 2: In the Azure Portal and Goto: +New > Compute > **Ubuntu Server** and fill the required details just like Windows:



Step 3: Fill all the details as same as that of a Windows creation except for a typical SSH public key login> Click **Create**:

Step 4: Select the required VM disk size in the **Size** blade:

Size	vCPUs	Data disks	Local SSD	Premium disk support	Load balancing	Price
DS1_V2 Standard	1	2	7 GB	3200 Max IOPS	Load balancing	4,179.93 INR/MONTH (ESTIMATED)
DS2_V2 Standard	2	4	14 GB	6400 Max IOPS	Load balancing	8,310.68 INR/MONTH (ESTIMATED)
DS11_V2 Standard	2	4	28 GB	6400 Max IOPS	Load balancing	9,343.37 INR/MONTH (ESTIMATED)

Step 5: Configure the VM settings in the same way you did for Windows VM and click **OK**:

The screenshot shows the Microsoft Azure portal interface for creating a new virtual machine. The left sidebar lists various services: New, Dashboard, All resources, Resource groups, App Services, SQL databases, SQL data warehouses, Azure Database for MySQL, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, Azure Active Directory, and More services. The main area shows the 'Create virtual machine' wizard with four steps: Basics (Done), Size (Done), Settings (Configure optional features, highlighted with a red box), and Summary (Ubuntu Server 16.04 LTS). To the right, the 'Settings' blade is open, displaying configuration sections for High availability, Storage (with 'Yes' selected for managed disks), Network (Virtual network: Edu533rg-vnet, Subnet: default (10.0.0.0/24), Public IP address: (new) Edu533linVM-ip, Network security group (firewall): (new) Edu533linVM-nsg), and Extensions. The 'OK' button at the bottom of the settings blade is highlighted with a red box.

Step 6: Once the settings is configured, it is **Validated** and upon clicking **OK**, VM is deployed:

The screenshot shows the Microsoft Azure portal interface for creating a new virtual machine. The left sidebar lists various services. The main area shows the 'Create virtual machine' wizard with four steps: Basics (Done), Size (Done), Settings (Done), and Summary (Ubuntu Server 16.04 LTS, highlighted with a red box). To the right, the 'Create' blade is open, showing validation status 'Validation passed' and a summary of settings. The summary includes: Basics (Subscription: Pay-As-You-Go, Resource group: Edu533rg, Location: South India); Settings (Computer name: Edu533linVM, Disk type: SSD, User name: edu533admin, Size: Standard DS1 v2, Managed: Yes, Virtual network: Edu533rg-vnet, Subnet: default (10.0.0.0/24), Public IP address: (new) Edu533linVM-ip); Terms of use (checkbox checked: I give Microsoft permission to use and share my contact information so that Microsoft or the Provider can contact me regarding this product and related products); and buttons for 'Create' (highlighted with a red box) and 'Download template and parameters'.

Step 7: Once the new VM is deployed and running, you will be redirected to the Overview window as shown below:

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various service icons like New, Dashboard, All resources, Resource groups, App Services, SQL databases, SQL data warehouses, Azure Database for MySQL, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, and Azure Active Directory. Below these are 'More services' and a 'More services >' button.

The main content area is titled 'Edu533linVM' under 'Virtual machine'. At the top, there are several action buttons: Connect, Start, Restart, Stop, Capture, Move, Delete, and Refresh. To the right of these buttons is the user's email address: omagarwal211@gmail.com and the text 'EDUREKA (DEFAULT DIRECTO...)'.

The 'Overview' section displays the following details:

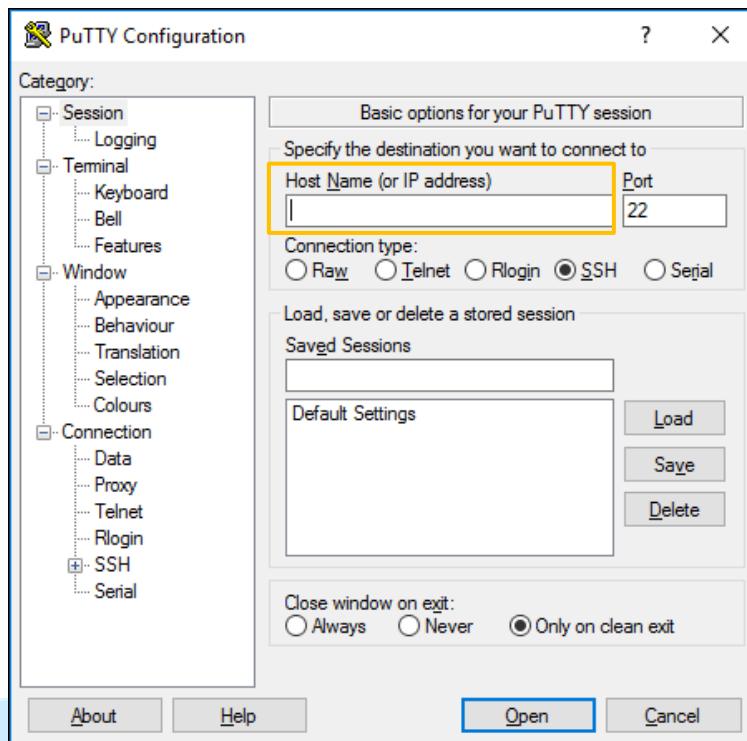
- Resource group: Edu533rg
- Status: Running
- Location: South India
- Subscription: Pay-As-You-Go
- Subscription ID: a3eee433-bf80-45b8-acae-b581d33f6a40
- Computer name: Edu533linVM
- Operating system: Linux
- Size: Standard DS1 v2 (1 vcpu, 3.5 GB memory)
- Public IP address: 13.71.114.48
- Virtual network/subnet: Edu533rg-vnet/default
- DNS name: Configure

Below this, there's a time range selector for monitoring data: 'Show data for last: 1 hour, 6 hours, 12 hours, 1 day, 7 days, 30 days'. Two charts are displayed: 'CPU (average)' and 'Network (total)'. The CPU chart shows usage from 5:45 PM to 6:30 PM. The Network chart shows traffic from 5:45 PM to 6:30 PM, with 'NETWORK IN' in blue and 'NETWORK OUT' in red.

edureka!

➤ **Connect to a LINUX VM**

Step 1: Open your local remote desktop client (PuTTY) and connect to the IP address or DNS name of your Linux VM:



Step 2: Once you have established a connection to your Linux VM, the below terminal will be accessible:

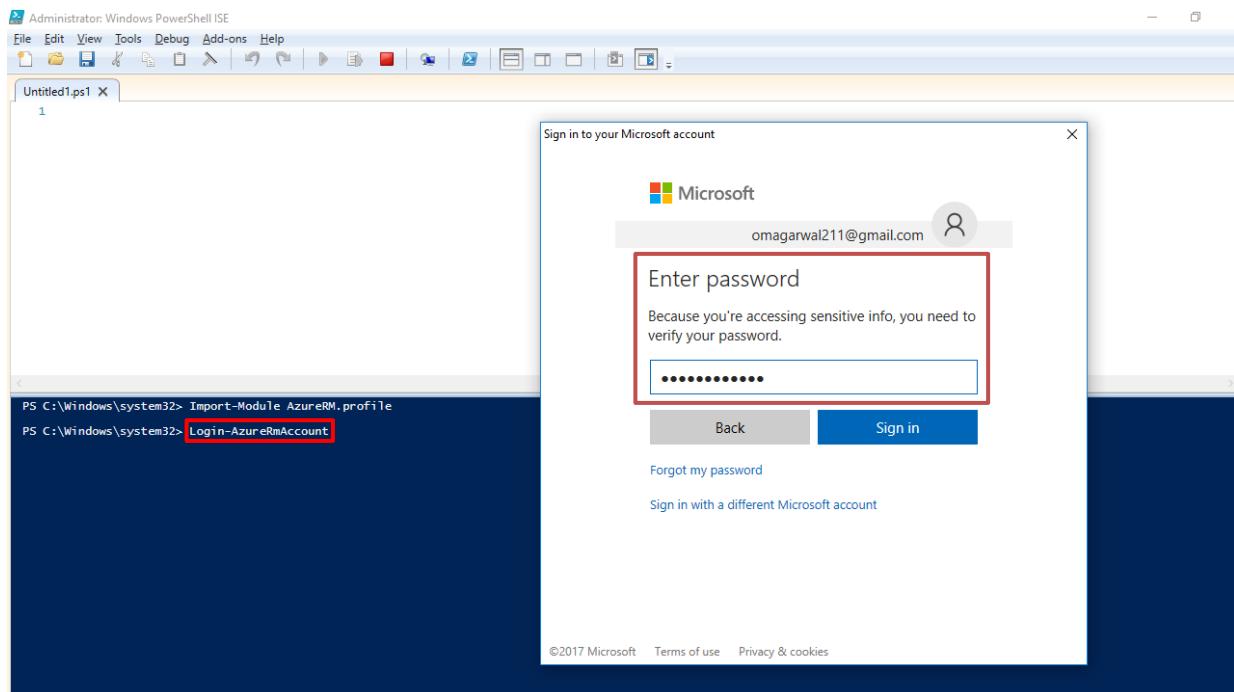
```
edurekara01@edurekalinvm: ~
Using username "edurekara01".
edurekara01@edurekalinvm's password:

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
edurekara01@edurekalinvm:~$
```

➤ **Connect to a LINUX VM**

Step 1: Once you have opened the PowerShell, Run **Login-AzureRmAccount** and enter the Account credentials:



Step 2: In the Command line, Enter the below shown command to create a Resource Group:

The screenshot shows a Windows PowerShell ISE window with a script named "creating a VM.ps1". The script contains the following code:

```
1 # Variables
2 ## Global
3 $ResourceGroupName = "Edu533newrg"
4 $Location = "SouthIndia"
```

An orange callout box points from the script area to a yellow box containing the text: "Keep Adding every parameter here as shown, to save it as a script and Run it as a whole to deploy a VM at once".

In the command line, the command `PS C:\Windows\system32> New-AzureRmResourceGroup -Name Edu533newrg -Location SouthIndia` is highlighted with a red box. The output of the command is displayed below, showing the resource group details:

```
ResourceGroupName : Edu533newrg
Location        : southindia
ProvisioningState : Succeeded
Tags            :
ResourceId      : /subscriptions/a3eee433-bf80-45b8-acae-b581d33f6a40/resourceGroups/Edu533newrg
```

Step 3: In the Command line, Enter the below shown command to create a Storage Account:

```

Administrator: Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
creating a VM.ps1* X
1 # Variables
2 ## Global
3 $ResourceGroupName = "Edu533newrg"
4 $Location = "SouthIndia"
5
6 # Resource Group
7 New-AzureRmResourceGroup -Name $ResourceGroupName -Location $Location
8
9 ## Storage
10 $StorageName = "edu533storage"
11 $StorageType = "Standard_GRS"
12
13 # Storage
14 $StorageAccount = New-AzureRmStorageAccount -ResourceGroupName $ResourceGroupName -Name $StorageName -Type $StorageType -Location $Location

```

PS C:\Windows\system32> New-AzureRmStorageAccount -ResourceGroupName Edu533newrg -Name edu533storage -Type Standard_GRS -Location SouthIndia

```

ResourceGroupName : edu533newrg
StorageAccountName : edu533storage
Id : /subscriptions/a3eee433-bf80-45b8-acae-b581d33f6a40/resourceGroups/edu533newrg/providers/Microsoft.Storage/storageAccounts/edu533storage
Location : southindia
Sku : Microsoft.Azure.Management.Storage.Models.Sku
Kind : Storage
Encryption : Microsoft.Azure.Management.Storage.Models.Encryption
AccessTier :
CreationTime : 09-11-2017 14:33:51
CustomDomain :
Identity :
LastGeoFailoverTime :
PrimaryEndpoints :
PrimaryLocation : southindia
ProvisioningState : Succeeded
SecondaryEndpoints :
SecondaryLocation :
StatusOfPrimary : Available
StatusOfSecondary : Available
Tags :
EnableHttpsTrafficOnly : False
Context : Microsoft.WindowsAzure.Commands.Common.Storage.LazyAzureStorageContext
ExtendedProperties : {}

```

Step 4: Enter the below commands to create PIP, Subnet, VNet, and NIC one after the other respectively as shown:

```

Administrator: Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
creating a VM.ps1* X
13 # Storage
14 $StorageAccount = New-AzureRmStorageAccount -ResourceGroupName $ResourceGroupName -Name $StorageName -Type $StorageType -Location $Location
15
16 ## Network
17 $InterfaceName = "ServerInterface533"
18 $Subnet1Name = "Subnet1"
19 $VNetName = "Edu533VNet"
20 $VNetAddressPrefix = "10.0.0.0/16"
21 $VNetSubnetAddressPrefix = "10.0.0.0/24"
22
23 # Network
24 $PIP = New-AzureRmPublicIpAddress -Name $InterfaceName -ResourceGroupName $ResourceGroupName -Location $Location -AllocationMethod Dynamic
25 $SubnetConfig = New-AzureRmVirtualNetworkSubnetConfig -Name $Subnet1Name -AddressPrefix $VNetSubnetAddressPrefix
26 $VNet = New-AzureRmVirtualNetwork -Name $VNetName -ResourceGroupName $ResourceGroupName -Location $Location -AddressPrefix $VNetAddressPrefix -Subnet $SubnetConfig
27 $Interface = New-AzureRmNetworkInterface -Name $InterfaceName -ResourceGroupName $ResourceGroupName -Location $Location -SubnetId $VNet.Subnets[0].Id -PublicIpAddressId

```

PS C:\Windows\system32> New-AzureRmNetworkInterface -Name ServerInterface533 -ResourceGroupName Edu533newrg -Location SouthIndia -SubnetId \$VNet.Subnets[0].Id -PublicIpAddressId

WARNING: The output object type of this cmdlet will be modified in a future release.

```

Name : ServerInterface533
ResourceGroupName : Edu533newrg
Location : southindia
Id : /subscriptions/a3eee433-bf80-45b8-acae-b581d33f6a40/resourceGroups/Edu533newrg/providers/Microsoft.Network/networkInterfaces/ServerInterface533
Etag : "W/"3456084-849f-43b5-8cf7-b95e2a762b44"
ResourceGuid : daa37558-f483-49f1-8cb3-c0cdf55c23c0
ProvisioningState : Succeeded
Tags :
VirtualMachine :
IpConfigurations :
  {
    "Name": "ipconfig1",
    "Etag": "W/"3456084-849f-43b5-8cf7-b95e2a762b44"",
    "Id": "/subscriptions/a3eee433-bf80-45b8-acae-b581d33f6a40/resourceGroups/Edu533newrg/providers/Microsoft.Network/networkInterfaces/ServerInterface533/ipConfigurations/ipconfig1",
    "PrivateIpAddress": "10.0.0.4",
    "PrivateIpAddressAllocationMethod": "Dynamic",
    "Subnet": {
      "Id": "/subscriptions/a3eee433-bf80-45b8-acae-b581d33f6a40/resourceGroups/Edu533newrg/providers/Microsoft.Network/virtualNetworks/Edu533VNet/subnets/Subnet1",
      "ResourceNavigationLinks": []
    }
  }

```

Step 5: First step to create a VM is to provide the desired credentials to that VM as shown below:

The screenshot shows a Windows PowerShell ISE window with a script named 'creating a VM.ps1'. The script is as follows:

```

25 $SubnetConfig = New-AzureRmVirtualNetworkSubnetConfig -Name $SubnetName -AddressPrefix $VNetSubnetAddressPrefix
26 $VNet = New-AzureRmVirtualNetwork -Name $VNetName -ResourceGroupName $ResourceGroupName -Location $Location -AddressPrefix $VNetAddressPrefix -Subnet $SubnetConfig
27 $Interface = New-AzureRmNetworkInterface -Name $InterfaceName -ResourceGroupName $ResourceGroupName -Location $Location -SubnetId $VNet.Subnets[0].Id -PublicIpAddressId
28
29 # Compute
30
31 ## Setup local VM object
32 $Credential = Get-Credential
33 $VirtualMachine = New-AzureRmVMConfig -VMName $VMName -VMSize $VMSize
34 $VirtualMachine = Set-AzureRmVMOperatingSystem -VM $VirtualMachine -Windows -ComputerName $ComputerName -Credential $Credential -ProvisionVMAgent -EnableAutoUpdate
35 $VirtualMachine = Set-AzureRmVMSourceImage -VM $VirtualMachine -PublisherName MicrosoftWindowsServer -Offer WindowsServer -Skus 2012-R2-Datacenter -Version "latest"
36 $VirtualMachine = Add-AzureRmNetworkInterface -VM $VirtualMachine -Id $Interface.Id
37 $OSDiskUri = $StorageAccount.PrimaryEndpoints.Blob.ToString()

```

A red box highlights the line 'cmdlet Get-Credential at command pipeline position 1'. A red arrow points from this box to a credential dialog box titled 'Supply values for the following parameters:'.

The credential dialog box contains:

- Username: edurekatestVM
- Password: [REDACTED]
- OK button
- Cancel button

Step 6: After creating Network resources, **Setup Local VM Object > Deploy the VM:**

The screenshot shows the same Windows PowerShell ISE window with the completed script 'creating a VM.ps1'. The script now includes additional code for setting up a local VM object and deploying it to Azure:

```

17 $InterfaceName = "ServerInterface533"
18 $SubnetName = "Subnet1"
19 $VNetName = "Edu533VNet"
20 $VNetAddressPrefix = "10.0.0.0/16"
21 $VNetSubnetAddressPrefix = "10.0.0.0/24"
22
23 # Network
24 $IP = New-AzureRmPublicIpAddress -Name $InterfaceName -ResourceGroupName $ResourceGroupName -Location $Location -AllocationMethod Dynamic
25 $SubnetConfig = New-AzureRmVirtualNetworkSubnetConfig -Name $SubnetName -AddressPrefix $VNetSubnetAddressPrefix
26 $VNet = New-AzureRmVirtualNetwork -Name $VNetName -ResourceGroupName $ResourceGroupName -Location $Location -AddressPrefix $VNetAddressPrefix -Subnet $SubnetConfig
27 $Interface = New-AzureRmNetworkInterface -Name $InterfaceName -ResourceGroupName $ResourceGroupName -Location $Location -SubnetId $VNet.Subnets[0].Id -PublicIpAddressId
28
29 ## Compute
30 $VMName = "edurekaVM533"
31 $ComputerName = "Edu533TestVM"
32 $VMSize = "Standard_LA2"
33 $OSDiskName = $VMName + "OSDisk"
34
35
36
37 ## Setup local VM object
38 $Credential = Get-Credential
39 $VirtualMachine = New-AzureRmVMConfig -VMName $VMName -VMSize $VMSize
40 $VirtualMachine = Set-AzureRmVMOperatingSystem -VM $VirtualMachine -Windows -ComputerName $ComputerName -Credential $Credential -ProvisionVMAgent -EnableAutoUpdate
41 $VirtualMachine = Set-AzureRmVMSourceImage -VM $VirtualMachine -PublisherName MicrosoftWindowsServer -Offer WindowsServer -Skus 2012-R2-Datacenter -Version "latest"
42 $VirtualMachine = Add-AzureRmNetworkInterface -VM $VirtualMachine -Id $Interface.Id
43 $VirtualMachine = $StorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds/" + $OSDiskName + ".vhd"
44 $OSDiskUri = $StorageAccount.PrimaryEndpoints.Blob.ToString() + "vhds/" + $OSDiskName + ".vhd"
45
46 ## Create the VM in Azure
47 New-AzureRmVM -ResourceGroupName $ResourceGroupName -Location $Location -VM $VirtualMachine

```

A yellow box highlights the line 'New-AzureRmVM -ResourceGroupName \$ResourceGroupName -Location \$Location -VM \$VirtualMachine'. A yellow arrow points from this box to the PowerShell command line at the bottom of the window.

The PowerShell command line shows the output of the deployment:

```

PS C:\Windows\system32> New-AzureRmVM -ResourceGroupName $ResourceGroupName -Location $Location -VM $VirtualMachine
WARNING: Since the VM is created using premium storage, existing standard storage account, edu533storage, is used for boot diagnostics.
RequestId IsSuccess StatusCode ReasonPhrase
----- ----- -----
True OK OK

```

After the deployment, you can save script for one time deployment. Find the Additional resource provided with this module

Step 7: Once the VM Creation command has been executed successfully, You can check the same on **Portal**:

NAME	TYPE	RESOURCE GROUP	LOCATION	SUBSCRIPTION
edu533storage	Storage account	Edu533newrg	South India	Pay-As-You-Go
Edu533VNet	Virtual network	Edu533newrg	South India	Pay-As-You-Go
edurekaVM533	Virtual machine	Edu533newrg	South India	Pay-As-You-Go
edurekaVM533_OsDisk_1_6f171c2dabc1443a864ea...	Disk	EDU533NEWRG	South India	Pay-As-You-Go
ServerInterface533	Network interface	Edu533newrg	South India	Pay-As-You-Go
ServerInterface533	Public IP address	Edu533newrg	South India	Pay-As-You-Go

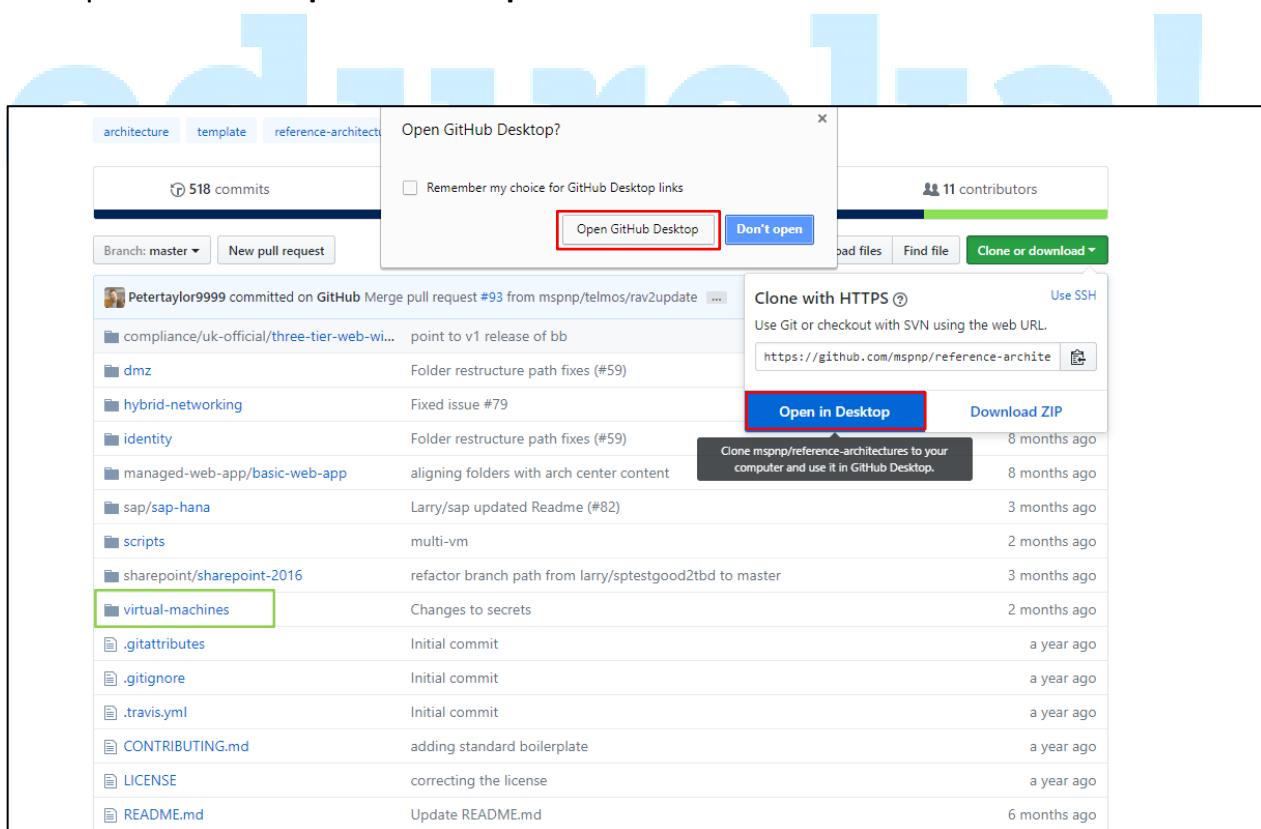
edureka!

➤ Running a LINUX based workload on Azure

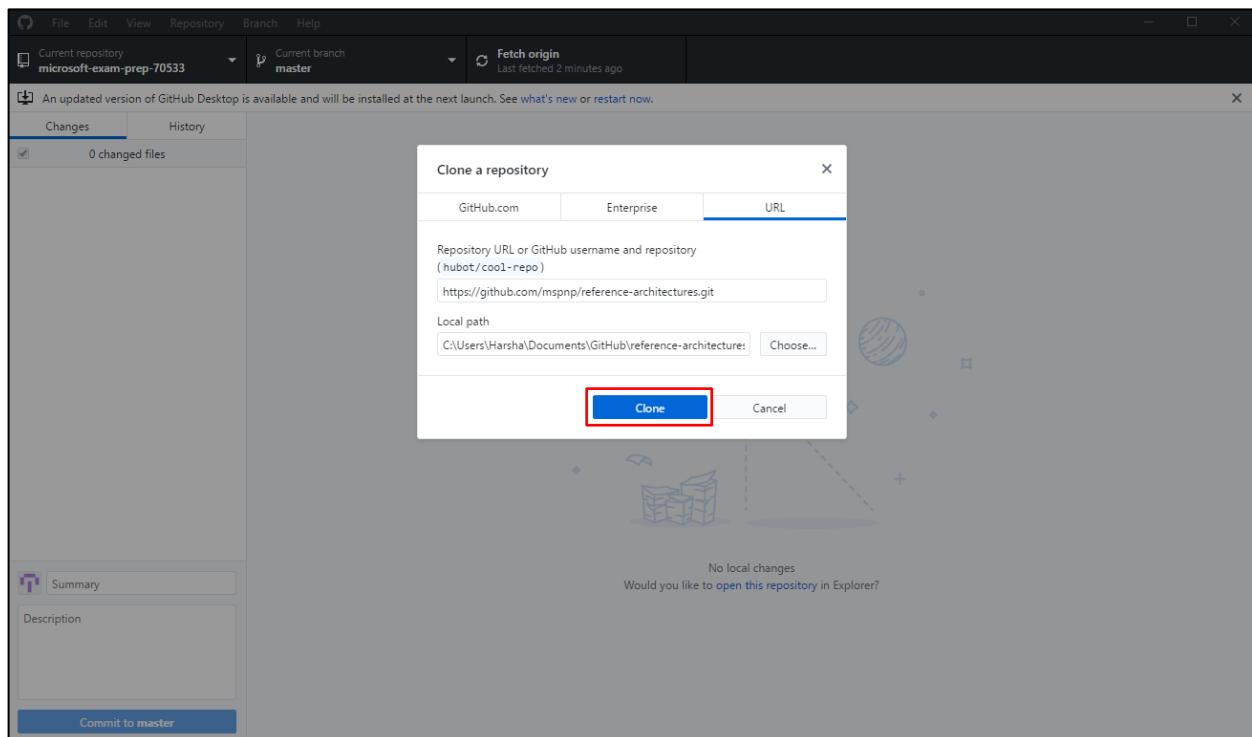
NOTE

- You need to download the GitHub Desktop tool if you choose to clone the repo on to your computer and use it, in case you already don't have it
- You can download the tool from the below link:
<https://desktop.github.com/>
- After the downloading the tool, Click on the below link to access the Solution Repo:
<https://github.com/mspnp/reference-architectures>

Step 1: Once you are redirected to the GitHub repo, Click on **Clone or Download** option > Select **Open in Desktop**:



Step 2: Once you select **Open in Desktop**, in the GitHub desktop tool popup window, Click on **Clone**:



NOTE

- You can perform the deployment through Microsoft CLI for Azure
- Click on the below link to install through Azure CLI Installer (MSI)

<https://aka.ms/InstallAzureCliWindows>

- For other ways to install CLI, click on the below link:

<https://docs.microsoft.com/en-us/cli/azure/install-azure-cli?view=azure-cli-latest>

Step 3: Follow the steps in the installation-window to install the CLI:

Azure CLI Latest ▾

Install on Windows

Install with MSI for the Windows command-line

To install the Azure CLI 2.0 for Windows command-line, run the Azure CLI 2.0 MSI setup.

- If you have not already done so, download the Azure CLI 2.0 MSI setup from the [Azure CLI 2.0 download page](#).
- Open the downloaded file and follow the on-screen instructions to complete the installation.
- Modify your system PATH variable to include the Azure CLI 2.0 directory.
- Run the following sudo commands:

```
bash
echo "deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/ buster main" | sudo tee /etc/apt/sources.list.d/azure-cli.list
curl https://packages.microsoft.com/keys/microsoft.asc | sudo apt-key add -
sudo apt-get install apt-transport-https
sudo apt-get update && sudo apt-get install azure-cli
```

NOTE

- As a part of the Azure Building Blocks, installing Node.js (Ver 6.9.4 or greater) is trivial for its functioning. (Node.js package serves as a building block for different service modules in the overall Microsoft Azure SDK for Node.js)
- Click on the below link to download Node.js:

<https://nodejs.org/en/download/> (Select your current OS and the type i.e., 32-bit or 64-bit on which you want Node.js to run on, and run it)

Step 4: Follow the steps in the installation window to install important features (**npm package manager**) on your OS:

Downloads

Latest LTS Version: 8.9.0 (includes npm 5.5.1)

Download the Node.js source code or a pre-built installer for your platform, and start developing today.

Windows Installer (.msi)
Windows Binary (.zip)
macOS Installer (.pkg)
macOS Binaries (.tar.gz)
Linux Binaries (x86/x64)
Linux Binaries (ARM)
Source Code

Additional Platforms

Step 5: Create a new directory for the Azure Building Blocks and Install the Azure Building Blocks **npm package** in the new directory:

```
18 Dir(s) 7,433,105,408 bytes free
C:\Users\Harsha> mkdir NPM
C:\Users\Harsha> cd NPM
C:\Users\Harsha\NPM>npm install -g @mspnp/azure-building-blocks
C:\Users\Harsha\AppData\Roaming\npm\azbb -> C:\Users\Harsha\AppData\Roaming\npm\node_modules\@mspnp\azure-building-blocks\src\index.js
+ @mspnp/azure-building-blocks@2.0.2
updated 1 package in 3.736s
C:\Users\Harsha\NPM>
```

Step 6: Run **azbb** command to test Azure Building blocks, if successful, below options are displayed:

```
C:\Users\Harsha\NPM>azbb
Usage: azbb [options]

[REDACTED]
A tool for deploying Azure infrastructure based on proven practices.

Options:
-V, --version                                output the version number
-s, --subscription-id <subscription-id>      Azure subscription id
-l, --location <location>                      Azure region in which to create the resource group
-g, --resource-group <resource-group>          name of the resource group
-p, --parameters-file <parameters-file>        path to a parameters file
--deploy                                         deploy resources, if --deploy is not used, azbb just creates the output files
-c, --cloud, <cloud>                           registered Azure cloud to use (use az cloud list to see all cloud names)
-o, --output-file <output-file>                 output file name prefix
-f, --output-format <output-format>            output format: json, files
-d, --defaults-directory <defaults-directory> directory containing customized default values
-t, --template-base-uri <template-base-uri>    base uri for building block templates
-k, --sas-token <sas-token>                     sas token to pass to access template-base-uri
-b, --building-blocks <building-blocks>         additional building blocks to add to the pipeline
-h, --help                                         output usage information

Visit https://aka.ms/azbbv2 for more information.
```

Step 7: Login to Azure Account using the below command and follow the prompts:

```
Node.js command prompt - az login
C:\Users\Harsha\NPM>az login
Usage: azbb [options]

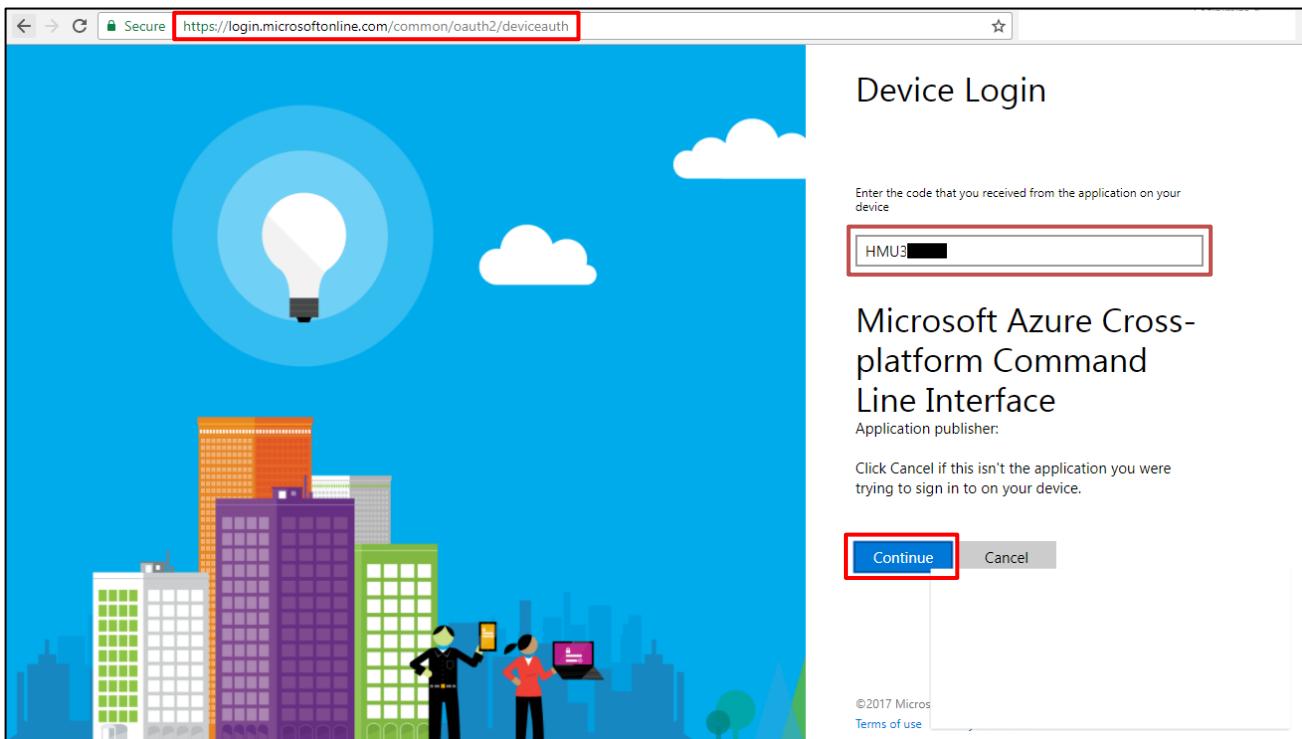
[REDACTED]
A tool for deploying Azure infrastructure based on proven practices.

Options:
-V, --version                                output the version number
-s, --subscription-id <subscription-id>      Azure subscription id
-l, --location <location>                      Azure region in which to create the resource group
-g, --resource-group <resource-group>          name of the resource group
-p, --parameters-file <parameters-file>        path to a parameters file
--deploy                                         deploy resources, if --deploy is not used, azbb just creates the output files
-c, --cloud, <cloud>                           registered Azure cloud to use (use az cloud list to see all cloud names)
-o, --output-file <output-file>                 output file name prefix
-f, --output-format <output-format>            output format: json, files
-d, --defaults-directory <defaults-directory> directory containing customized default values
-t, --template-base-uri <template-base-uri>    base uri for building block templates
-k, --sas-token <sas-token>                     sas token to pass to access template-base-uri
-b, --building-blocks <building-blocks>         additional building blocks to add to the pipeline
-h, --help                                         output usage information

Visit https://aka.ms/azbbv2 for more information.

C:\Users\Harsha\NPM>az login
To sign in, use a web browser to open the page https://aka.ms/devicelogin and enter the code HMUJ [REDACTED] to authenticate.
```

Step 8: In the opened URL page (Azure Cross-Platform CLI), enter the **Code** obtained from the CLI application > Continue:



Step 9: Once your device is **logged in** to Azure (CLI Integration), the same is acknowledged in the **Command Prompt**:

```
Node.js command prompt
-f, --output-format <output-format>          output format: json, files
-d, --defaults-directory <defaults-directory>    directory containing customized default values
-t, --template-base-uri <template-base-uri>      base uri for building block templates
-k, --sas-token <sas-token>                      sas token to pass to access template-base-uri
-b, --building-blocks <building-blocks>          additional building blocks to add to the pipeline
-h, --help                                         output usage information

Visit https://aka.ms/azbbv2 for more information.

C:\Users\Harsha\NPM>az login
To sign in, use a web browser to open the page https://aka.ms/devicelogin and enter the code HMU3 [REDACTED] to authenticate.

[{"cloudName": "AzureCloud", "id": "a3eee433-bf80-[REDACTED]", "isDefault": true, "name": "Pay-As-You-Go", "state": "Enabled", "tenantId": "d2220475-f677-[REDACTED]", "user": {"name": "[REDACTED]@gmail.com", "type": "user"}}]
```

NOTE

To deploy the sample single VM workload, follow these steps:

1. Navigate to the virtual-machines\single-vm\parameters\linux folder for the repository you downloaded in the pre-requisites step at the beginning
2. Open the single-vm-v2.json file and enter a username and SSH key between the quotes, as shown below, then save the file

```
{  
  "type": "VirtualMachine",  
  "settings": {  
    "vmCount": 1,  
    "namePrefix": "ra-single-linux",  
    "computerNamePrefix": "web",  
    "size": "Standard DS1 v2",  
    "adminUsername": "EdutestVM533",  
    "sshPublicKey": "REDACTED",  
    "virtualNetwork": {  
      "name": "ra-single-linux-vm-vnet"  
    },  
  },  
}
```



Step 10: Run azbb to deploy the sample VM as shown below: **azbb -s <subscription_id> -g <resource_group_name> -l <location> -p single-vm-v2.json -deploy**

```
Nodejs command prompt - azbb -s a3eee433-bf80-45b8-acae-b581d33f6a40 -g Edusampledeploy -l southindia -p single-vm-v2.json --deploy

C:\Users\Harsha\Documents\GitHub\reference-architectures\virtual-machines\single-vm\parameters\linux>azbb -s a3eee433-bf80-45b8-acae-b581d33f6a40 -g Edusampledeploy -l southindia -p single-vm-v2.json --deploy

parameters written to C:\Users\Harsha\Documents\GitHub\reference-architectures\virtual-machines\single-vm\parameters\linux\single-vm-v2-output-01.json

parameters written to C:\Users\Harsha\Documents\GitHub\reference-architectures\virtual-machines\single-vm\parameters\linux\single-vm-v2-output-02.json

parameters written to C:\Users\Harsha\Documents\GitHub\reference-architectures\virtual-machines\single-vm\parameters\linux\single-vm-v2-output-03.json

parameters written to C:\Users\Harsha\Documents\GitHub\reference-architectures\virtual-machines\single-vm\parameters\linux\single-vm-v2-output-04.json

{
  "id": "/subscriptions/a3eee433-bf80-45b8-acae-b581d33f6a40/resourceGroups/Edusampledeploy",
  "location": "southindia",
  "managedBy": null,
  "name": "Edusampledeploy",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null
}
/ Running ...

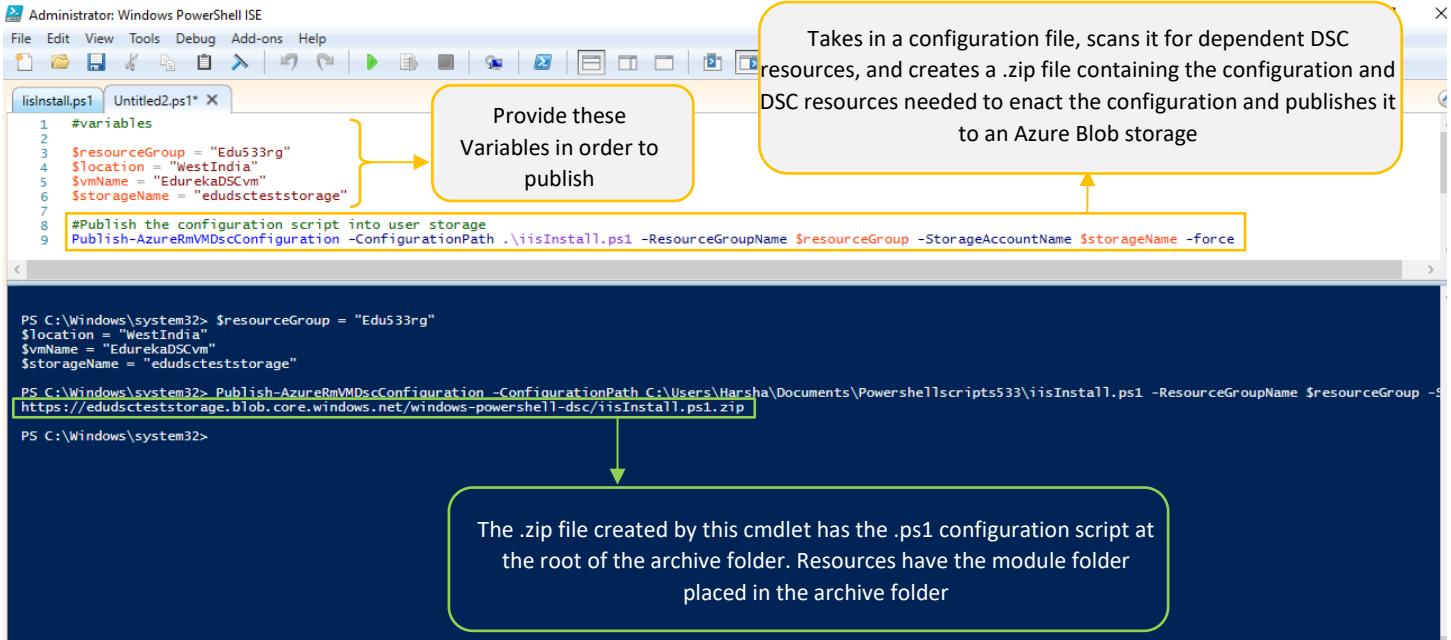
All the required resources are created one after the other
```

Step 11: All the required resources are deployed as **parameterized** in the **template** and you can check that in **Portal**:

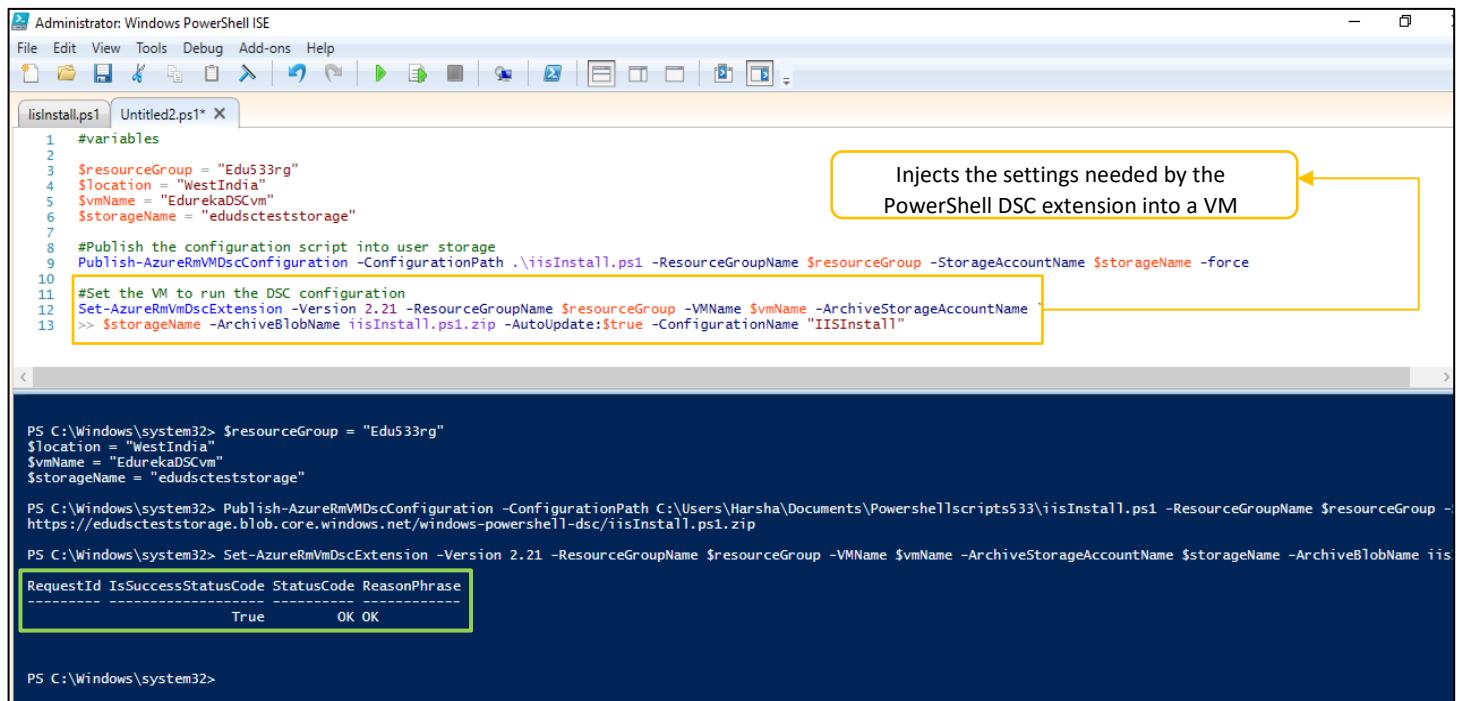
NAME	TYPE	RESOURCE GROUP	LOCATION	SUBSCRIPTION
ra-single-linux-vm1-nic1	Network interface	Edusampledeploy	South India	Pay-As-You-Go
ra-single-linux-vm1-nic1-pip	Public IP address	Edusampledeploy	South India	Pay-As-You-Go
ra-single-linux-vm-nsg	Network security group	Edusampledeploy	South India	Pay-As-You-Go
ra-single-linux-vm-vnet	Virtual network	Edusampledeploy	South India	Pay-As-You-Go
vmv4zfhaycb72gdiag1	Storage account	Edusampledeploy	South India	Pay-As-You-Go

➤ Configuration Management using PowerShell DSC

Step 1: Once you save the Configuration file, publish it to an User Storage as shown below:



Step 2: Insert the command shown, to Set the VM to run the DSC configuration:



Step 3: Once the DSC Configuration has been set to a VM, you can check the Status as shown below:

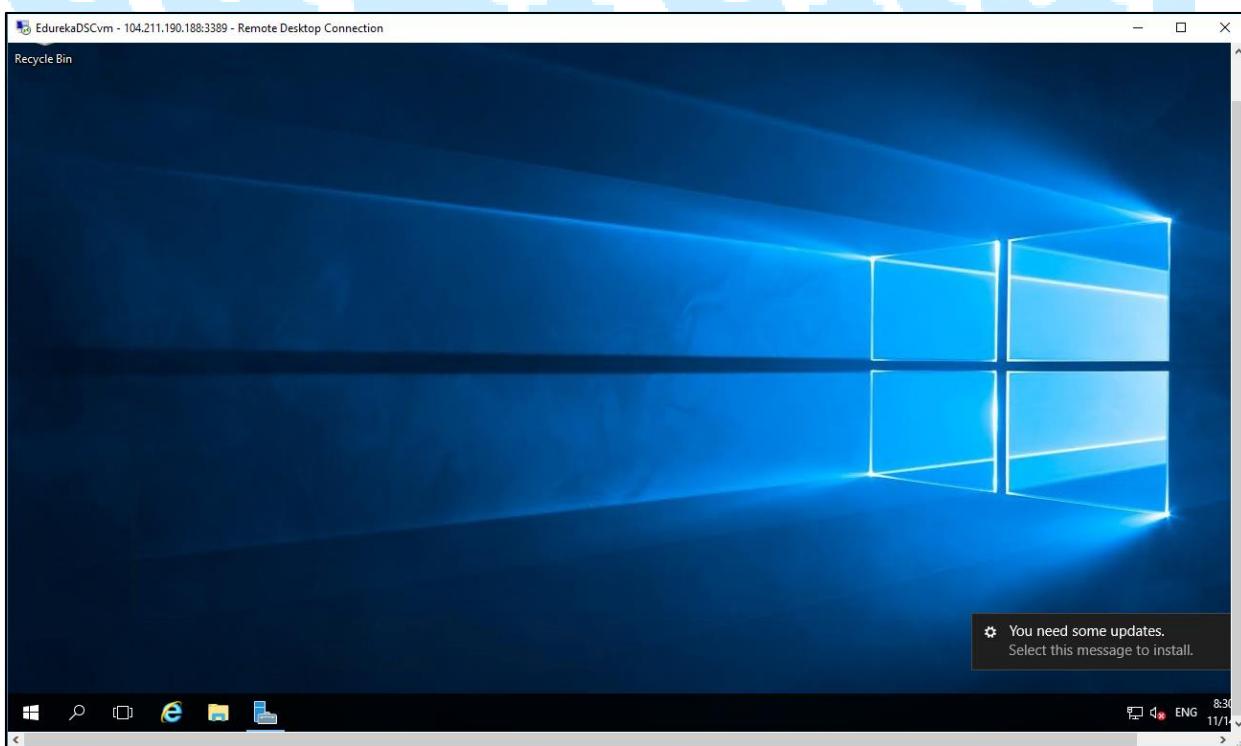
The screenshot shows the Windows PowerShell ISE interface. The top menu bar includes File, Edit, View, Tools, Debug, Add-ons, and Help. Below the menu is a toolbar with various icons. Two tabs are open: 'iisInstall.ps1' and 'Untitled2.ps1'. The 'iisInstall.ps1' tab contains a PowerShell script for publishing a configuration and retrieving its status. The 'Untitled2.ps1' tab is currently empty. In the bottom pane, a command is run in the PowerShell window:

```
PS C:\Windows\system32> Get-AzureRmMDscExtensionStatus
cmdlet Get-AzureRmMDscExtensionStatus at command pipeline position 1
Supply values for the following parameters:
(Type !? for Help.)
ResourceGroupName: Edu533rg
VMName: EduurekaDSCvm

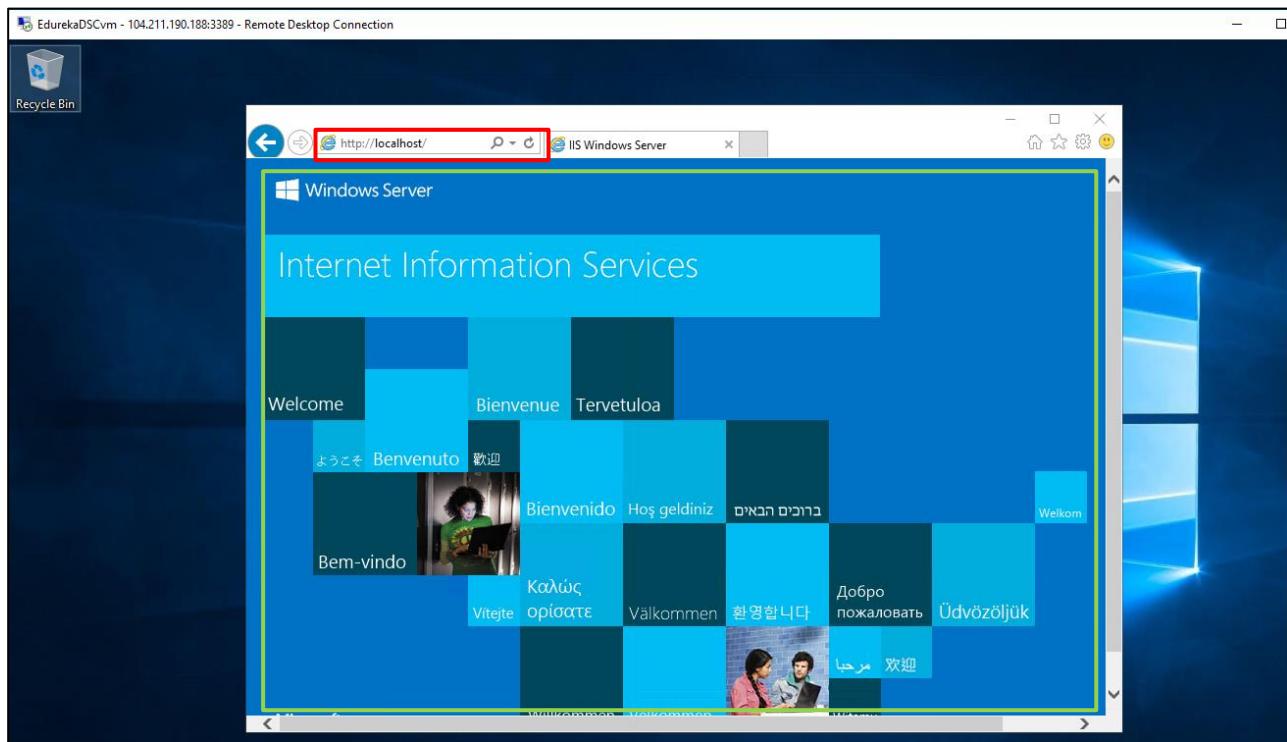
ResourceGroupName : Edu533rg
VmName          : EduurekaDSCvm
Version         : 2.21
Status          : Provisioning succeeded
StatusCode       : ProvisioningState/succeeded
Timestamp        : 14-11-2017 07:09:16
StatusMessage    : DSC configuration was applied successfully.
DscConfigurationLog : {{[2017-11-14 07:08:10Z] [VERBOSE] [EduurekaDSCvm]: LCM: [ Start Set ], [2017-11-14 07:08:10Z] [VERBOSE] [EduurekaDSCvm]: LCM: [ Start Resource ] [[WindowsFeature]IIS], [2017-11-14 07:08:10Z] [VERBOSE] [EduurekaDSCvm]: LCM: [ Start Test ] [[WindowsFeature]IIS], [2017-11-14 07:08:13Z] [VERBOSE] [EduurekaDSCvm]: [[WindowsFeature]IIS] The operation 'Get-WindowsFeature' started: Web-Server...}

PS C:\Windows\system32>
```

Step 4: Once the implementation is succeeded, you can check the same logging on to the DSC configured VM:



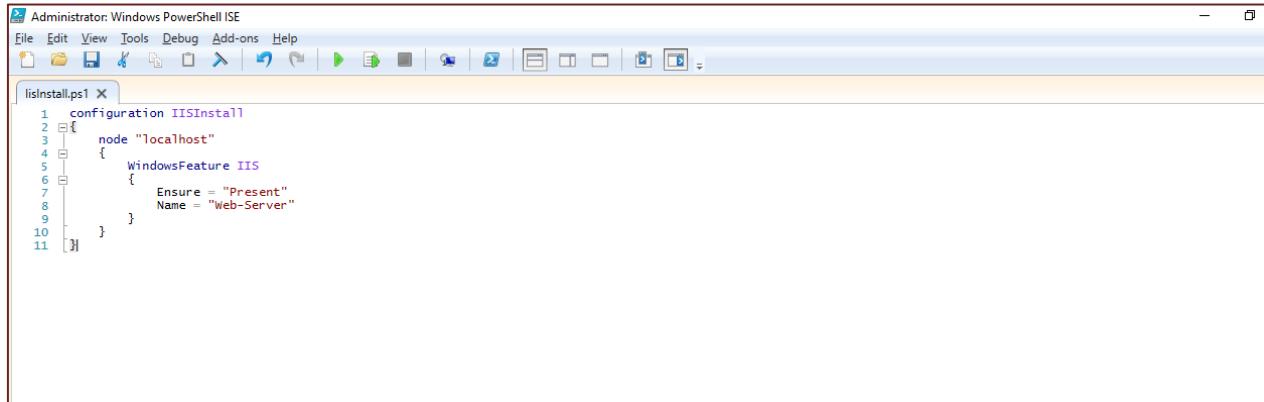
Step 5: In the DSC configured VM, open the browser and check the **localhost** if IIS has been installed:



edureka!

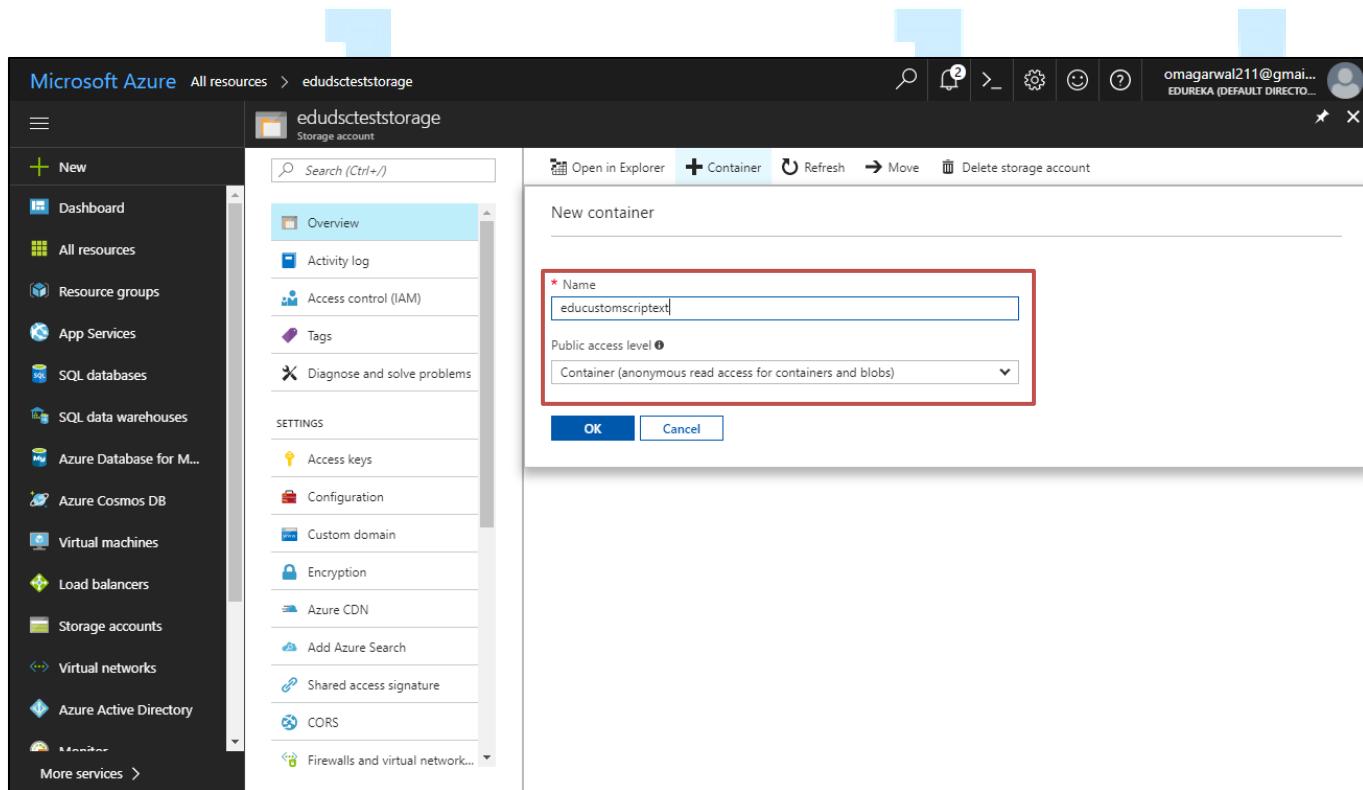
➤ Configuration Management using Custom Script Management

Step 1: Write a Script File as you did for the DSC Extension, for installing a IIS website:



```
Administrator: Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
File Explorer Task List Run Taskbar
iisInstall.ps1 X
1 configuration IISInstall
2 {
3     node "localhost"
4     {
5         WindowsFeature IIS
6         {
7             Ensure = "Present"
8             Name = "Web-Server"
9         }
10    }
11 }
```

Step 2: In the **portal**, under your **Storage Account**, create a **New Container** to hold the **Script** file, as shown below:



The screenshot shows the Microsoft Azure portal interface. On the left, the navigation menu includes options like Dashboard, All resources, Resource groups, App Services, SQL databases, SQL data warehouses, Azure Database for MySQL, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, Azure Active Directory, and More services. The main area displays the 'edudscteststorage' storage account. A modal window titled 'New container' is open, prompting for a container name ('educustomscripttext') and public access level ('Container (anonymous read access for containers and blobs)').

Step 3: Once the Container is created, **Upload** the desired **Script File** as shown below:

The screenshot shows the Microsoft Azure Storage Explorer interface. On the left, there's a sidebar with various service icons like Dashboard, All resources, Resource groups, App Services, etc. The main area shows a container named 'educustomscripttext'. Inside this container, there's a list of blobs with the header 'NAME' and 'MODIFIED'. A red box highlights the 'Upload' button and the file input field where 'IisInstall.ps1' is selected. There's also a checkbox for 'Overwrite if files already exist'.

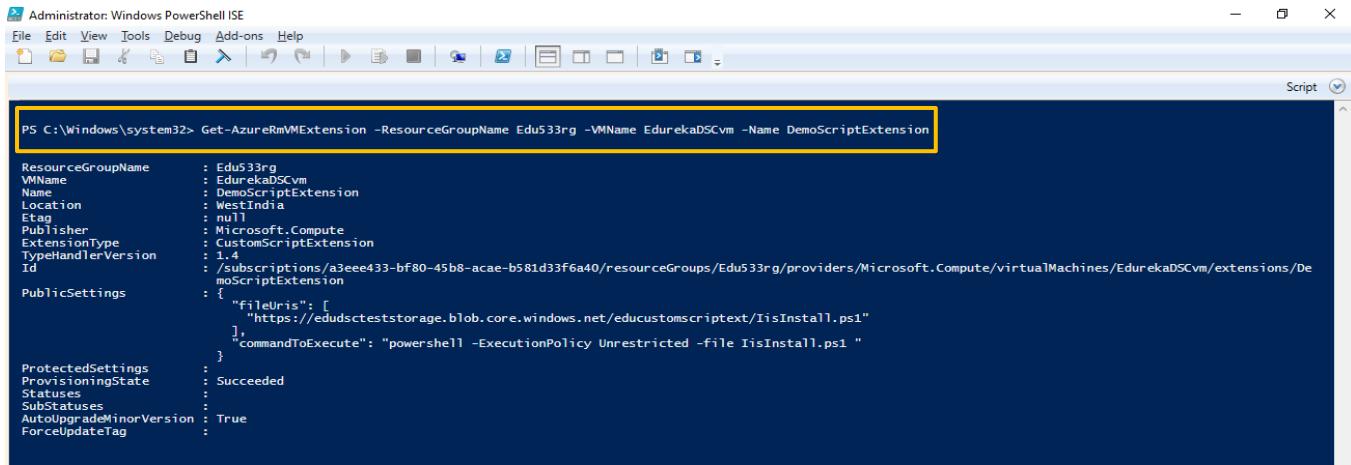
Step 4: Run the below cmdlet along with the parameters to add the Custom Script Extension to the VM:

```
PS C:\Windows\system32> Set-AzureRmVMCustomScriptExtension -ResourceGroupName Edu533rg ` 
    -VMName EdurekaDSCvm ` 
    -Location westindia ` 
    -FileUri https://edudscteststorage.blob.core.windows.net/educustomscripttext/IisInstall.ps1 ` 
    -Run 'IisInstall.ps1' ` 
    -Name DemoScriptExtension

RequestId IsSuccessStatusCode StatusCode ReasonPhrase
----- ----- ----- -----
True      OK      OK      OK
```

The screenshot shows a Windows PowerShell ISE window. The command `Set-AzureRmVMCustomScriptExtension` is run with several parameters: Resource Group Name (Edu533rg), VM Name (EdurekaDSCvm), Location (westindia), File Uri (https://edudscteststorage.blob.core.windows.net/educustomscripttext/IisInstall.ps1), Run Script (IisInstall.ps1), and Extension Name (DemoScriptExtension). The output shows a successful execution with status codes OK for all metrics.

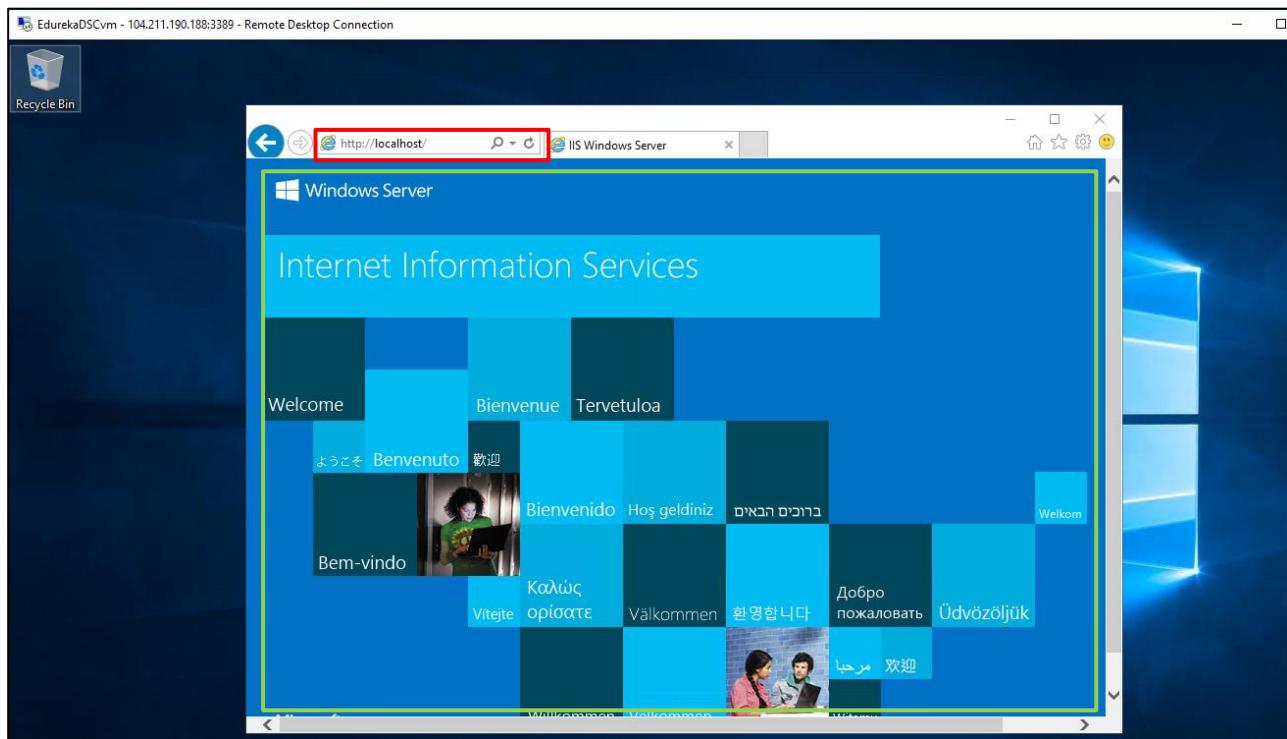
Step 5: Once you have added the extension, you can check the Deployment State using the below command:



```
PS C:\Windows\system32> Get-AzureRmVMExtension -ResourceGroupName Edu533rg -VMName EdurekaDSCvm -Name DemoScriptExtension

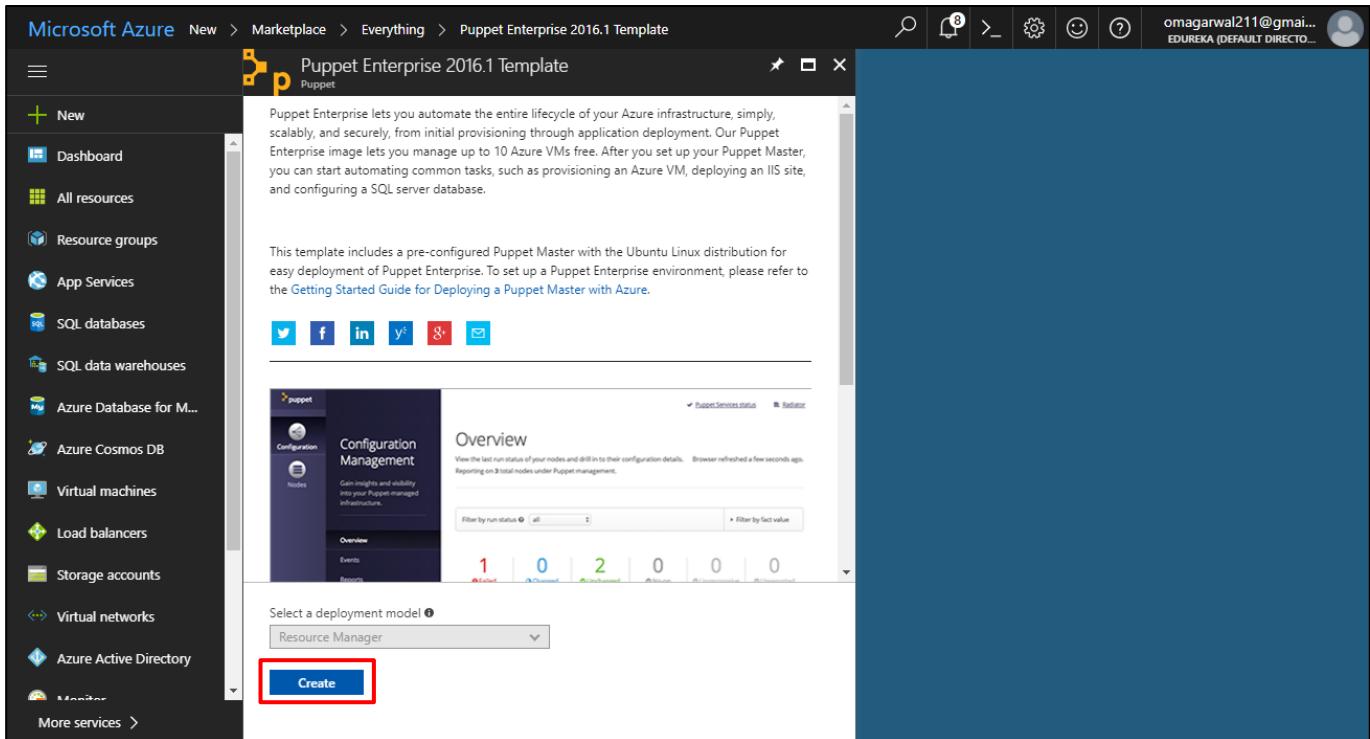
ResourceGroupName : Edu533rg
VMName          : EdurekaDSCvm
Name            : DemoScriptExtension
Location        : WestIndia
Etag            : null
Publisher       : Microsoft.Compute
ExtensionType   : CustomScriptExtension
TypeHandlerVersion : 1.4
Id              : /subscriptions/a3eee433-bf80-45b8-acae-b581d33f6a40/resourceGroups/Edu533rg/providers/Microsoft.Compute/virtualMachines/EdurekaDSCvm/extensions/DemoScriptExtension
PublicSettings  : {
    "fileUris": [
        "https://edudscteststorage.blob.core.windows.net/educustomscripttext/IisInstall.ps1"
    ],
    "commandToExecute": "powershell -ExecutionPolicy Unrestricted -file IisInstall.ps1"
}
ProtectedSettings: 
ProvisioningState: Succeeded
Statuses        : 
SubStatuses     : 
AutolicenseMinorVersion: True
ForceUpdateTag  :
```

Step 6: Once the deployment is successful, login to the VM and check if IIS is installed:

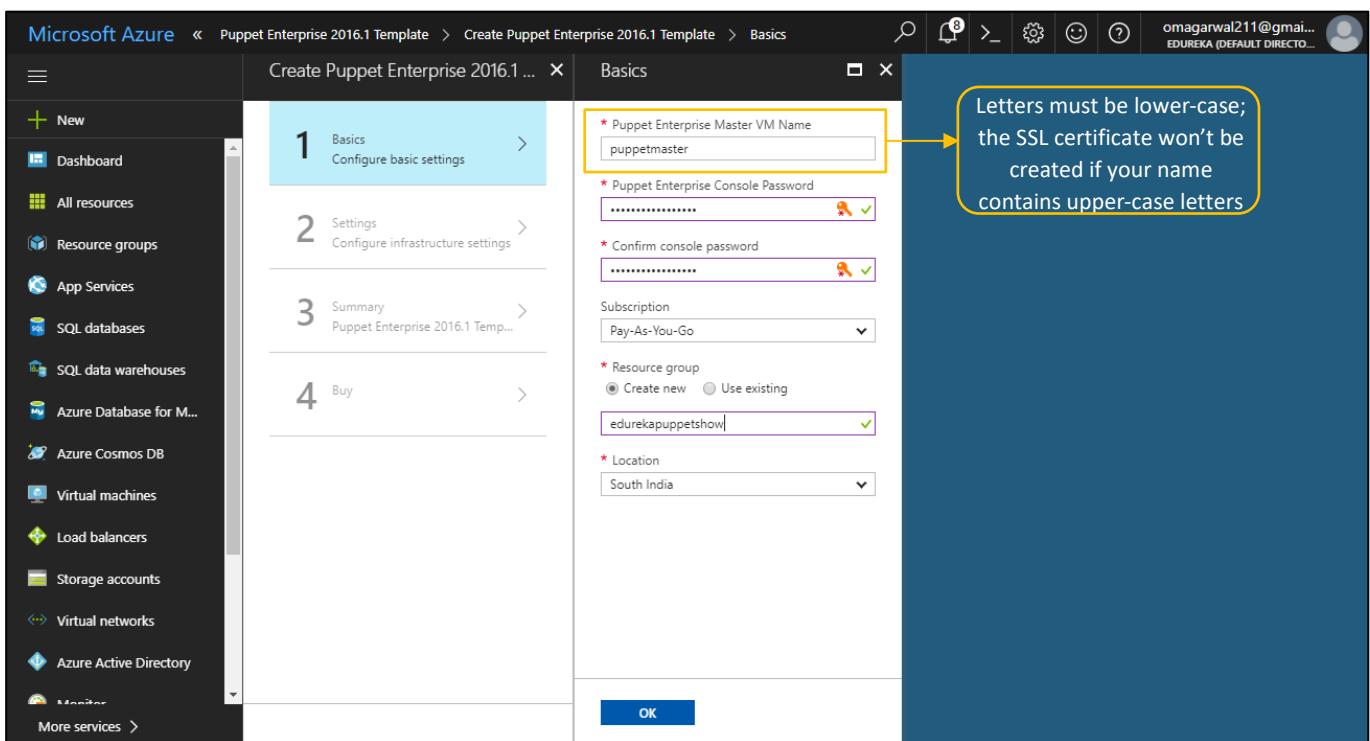


➤ Deploying Puppet Enterprise in Microsoft Azure

Step 1: In the Portal, Goto: +New > Search Puppet > Select **Puppet Enterprise 2016.1 Template** > Click **Create**:



Step 2: In the basics blade, configure as shown below:



Step 3: In Settings blade, Add the required **Admin Username** and authentication type (SSH/Password) to the **PE Master**:

Microsoft Azure < Puppet Enterprise 2016.1 Template > Create Puppet Enterprise 2016.1 Template > Settings

1 Basics Done ✓

2 Settings Configure infrastructure settings >

3 Summary Puppet Enterprise 2016.1 Temp... >

4 Buy >

Compute

- * Username
- * Authentication type Password SSH public key
- * Password
- * Confirm password

Network

- * Public IP address
- * DNS label

Storage

OK

Leave these options with default values

Step 4: Check the summary of the whole PE configuration > Click **OK** > Select **I Agree** in next blade > Click **Create**:

Microsoft Azure < Puppet Enterprise 2016.1 Template > Create Puppet Enterprise 2016.1 Template > Summary

1 Basics Done ✓

2 Settings Done ✓

3 Summary Puppet Enterprise 2016.1 Temp... >

4 Buy >

Validation passed

Basics	Subscription: Pay-As-You-Go
	Resource group: edurekappuppetshow
	Location: South India
Settings	Puppet Enterprise Master VM ...: puppetmaster
	Puppet Enterprise Console Pass...: *****
	Username: edupemaster
	Password: *****
	Size: Standard D2 v2
	Public IP address: ip01
	DNS label: puppet87d9
	Storage account: strd3c9

OK Download template and parameters

Step 5: Check the successful deployments:

The screenshot shows the Microsoft Azure portal interface. On the left, the navigation menu includes options like New, Dashboard, All resources, Resource groups, App Services, SQL databases, SQL data warehouses, Azure Database for MySQL, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, Azure Active Directory, and Monitor. The main content area is titled 'edurekapuppetshow' and shows the 'Resource group' overview. It displays a summary: Subscription name (Pay-As-You-Go), Deployment ID (a3eee433-bf80-45b8-acae-b581d33f6a40), and 6 Succeeded deployments. Below this, a table lists seven items under 'SETTINGS': ip01 (Public IP address, South India), puppetmaster (Virtual machine, South India), puppetmasterNic (Network interface, South India), puppetNetworkSecurityGroup (Network security group, South India), strd3c9 (Storage account, South India), strd3c92 (Storage account, South India), and vnet01 (Virtual network, South India). A red box highlights the list of resources.

Step 6: Copy the DNS Name which is found here:

The screenshot shows the Microsoft Azure portal interface, specifically the 'ip01' Public IP address details page. The left sidebar has the same set of options as the previous screenshot. The main content area shows the 'Overview' tab selected. Key details include the Resource group (edurekapuppetshow), IP address (52.172.52.67), and the DNS name (puppet87d9.southindia.cloudapp.azure.com), which is highlighted with a green box. Other details shown include Location (South India), Subscription name (Pay-As-You-Go), and Subscription ID (a3eee433-bf80-45b8-acae-b581d33f6a40). The 'Associated to' section lists puppetmasterNic and puppetmaster.

Step 7: In the CLI or other SSH client (putty), run the command as shown below:

```
puppet87d9.southindia.cloudapp.azure.com - PuTTY
login as: puppetmaster
puppetmaster@puppet87d9.southindia.cloudapp.azure.com's password:
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 3.19.0-64-generic x86_64)

 * Documentation: https://help.ubuntu.com/
 
 System information as of Thu Nov 16 05:22:09 UTC 2017

 System load: 0.15           Memory usage: 1%   Processes:      90
 Usage of /:  7.4% of 29.50GB   Swap usage:  0%   Users logged in: 0

 Graph this data and manage this system at:
 https://landscape.canonical.com/
 
 Get cloud support with Ubuntu Advantage Cloud Guest:
 http://www.ubuntu.com/business/services/cloud

164 packages can be updated.
105 updates are security updates.

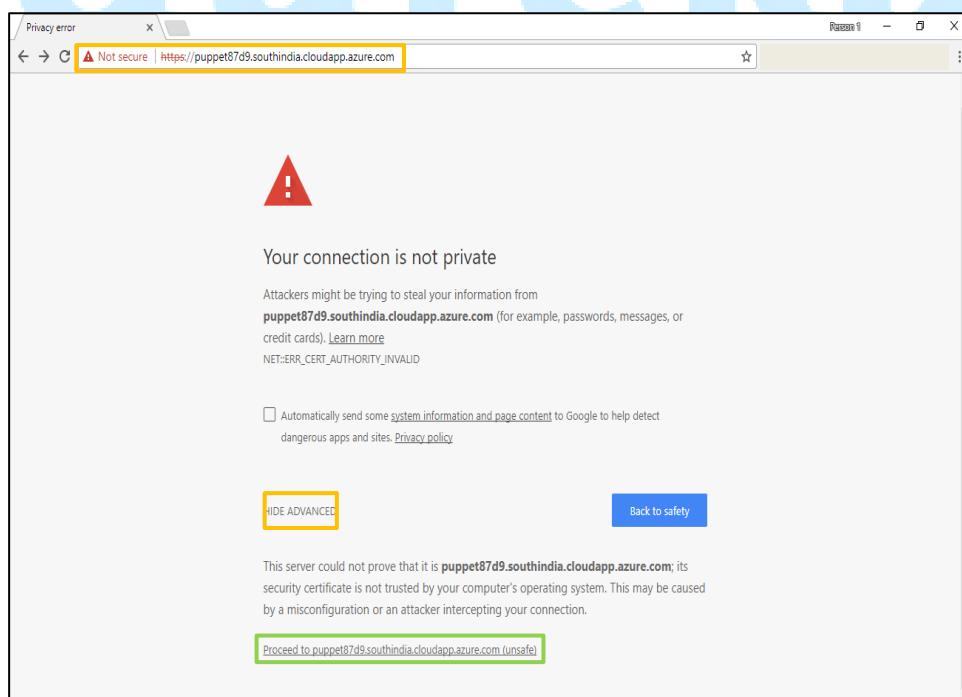
New release '16.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

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.

$
```

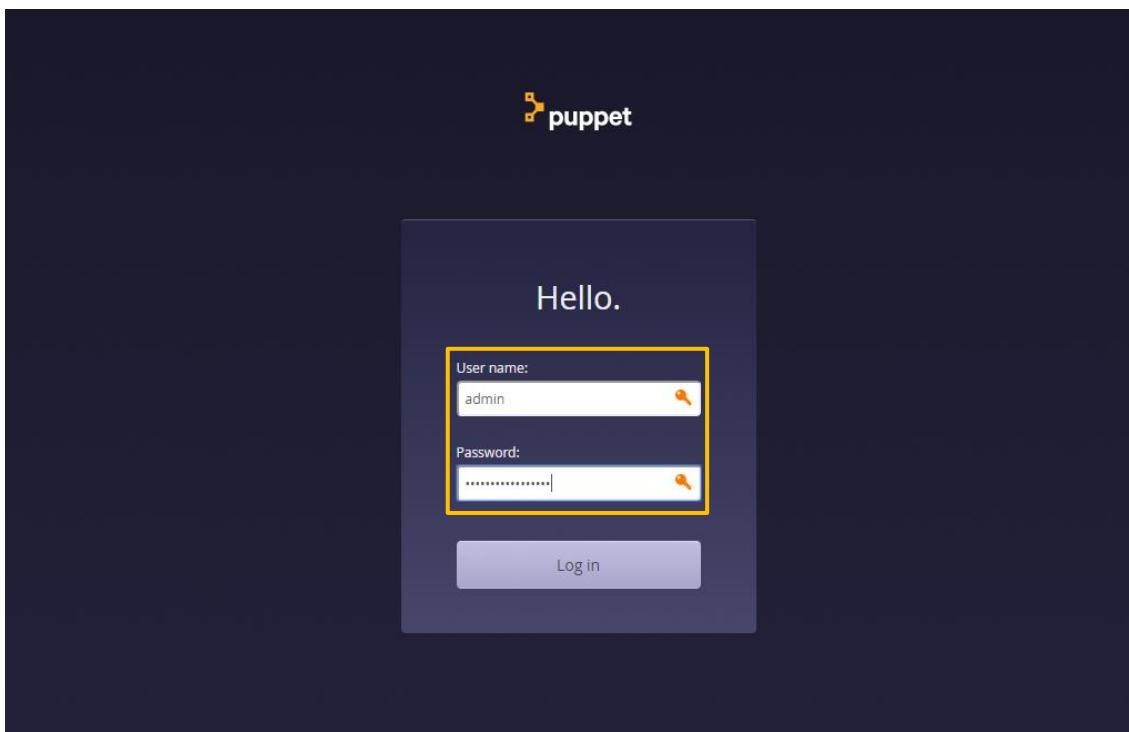
Step 8: To check the VM creation, Open browser > With **https** protocol, enter DNS name > Click **Advanced** > **Proceed**:



For More Details:

https://puppet.com/docs/pe/2017.3/accessing_console/console_accessing.html

Step 9: Once you proceed, you will be redirected to the Console window >
Provide console credentials as **admin**:



Step 10: As you haven't set up any agents yet, there's not much information here:

The screenshot shows the Puppet Dashboard's "Overview" page. On the left sidebar, under "Configuration Management", there are links for "Nodes" and "Events". The main area displays an "Overview" section with a summary of node status: 0 Failed, 0 Changed, 1 Unchanged, 0 No-op, 0 Unresponsive, and 0 Unreported. Below this is a table titled "Run status" with columns for "Run status", "Node name", "Last report", and "Node graph". One row is highlighted with a yellow box, showing "puppet87d9.southindia.cloudapp.azure.com" as the node name, "2017-11-16 05:54 Z" as the last report time, and a "View graph" link. At the bottom of the table, there are "Previous" and "Next" navigation buttons. A large orange arrow points from the highlighted row in the table to a callout box at the bottom right of the table area, which contains the text: "The puppet master node listed, along with its private DNS name here :".

Run status	Node name	Last report	Node graph
✓	puppet87d9.southindia.cloudapp.azure.com	2017-11-16 05:54 Z	View graph

Step 11: In the Azure portal, click New > click Compute > Select Windows Server 2012 R2 Datacenter > Click Create:

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various service icons like Dashboard, All resources, Resource groups, App Services, etc. The main area is titled '[smalldisk] Windows Server 2012 Datacenter'. It displays a note about the image being 30GB OS Disk, compatibility with traditional applications, and the use of Azure Hybrid Benefit. There are also links for Legal Terms, Microsoft Publisher information, and Useful Links to Documentation and What's New in 2012. At the bottom, a dropdown menu for 'Select a deployment model' is set to 'Resource Manager', and a large blue 'Create' button is highlighted with a red box.

Step 12: Enter the Basic information required and the credentials:

The screenshot shows the 'Create virtual machine' process in the Azure portal. The current step is 'Basics'. The form contains the following fields: Name (newpuppetagent), VM disk type (SSD), User name (puppetagent01), Password and Confirm password (both masked with dots), Subscription (Pay-As-You-Go), Resource group (radio buttons for 'Create new' or 'Use existing' - 'Use existing' is selected with 'edurekapuppetshow'), and Location (South India). A large blue 'OK' button at the bottom is highlighted with a red box.

Step 13: Choose whichever size will suit this VM's anticipated workload:

Step 14: In the Settings blade, Click on Extensions > Add Extension > Select Puppet Extension:

Step 15: Add the FQDN (DNS name) of the Puppet Master VM in order to register it to the Puppet Agent:

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various service icons like Dashboard, All resources, Resource groups, App Services, etc. The main area is titled 'Install extension' under 'Extensions > New resource > Puppet Agent (preview)'. A red box highlights the 'Puppet Master Server' input field, which contains the value 'puppet87d9.southindia.cloudapp.azure.com'. At the bottom right of the dialog is an 'OK' button.

Step 16: Check the Summary of the Puppet Agent VM > Agree > Click Create:

The screenshot shows the Microsoft Azure portal interface. The left sidebar includes 'New' and 'Virtual machines' under 'Create'. The main area is titled 'Create virtual machine' under 'Create virtual machine > Create'. Step 4 is labeled 'Summary [smalldisk] Windows Server 20...'. The summary details include: Offer details (Standard DS1 v2 by Microsoft, 8.6586 INR/hr), Pricing for other VM sizes, and a note about Azure resource usage. The 'Terms of use' section contains legal text and a checked checkbox for giving Microsoft permission to use contact information. At the bottom right is a large blue 'Create' button, which is also highlighted with a red box.

Step 17: Open the PE console as you did previously > Select the **Nodes** tab to see all current nodes:

Node name	Last report
puppet87d9.southindia.cloudapp.azure.com	2017-11-16 06:54:2

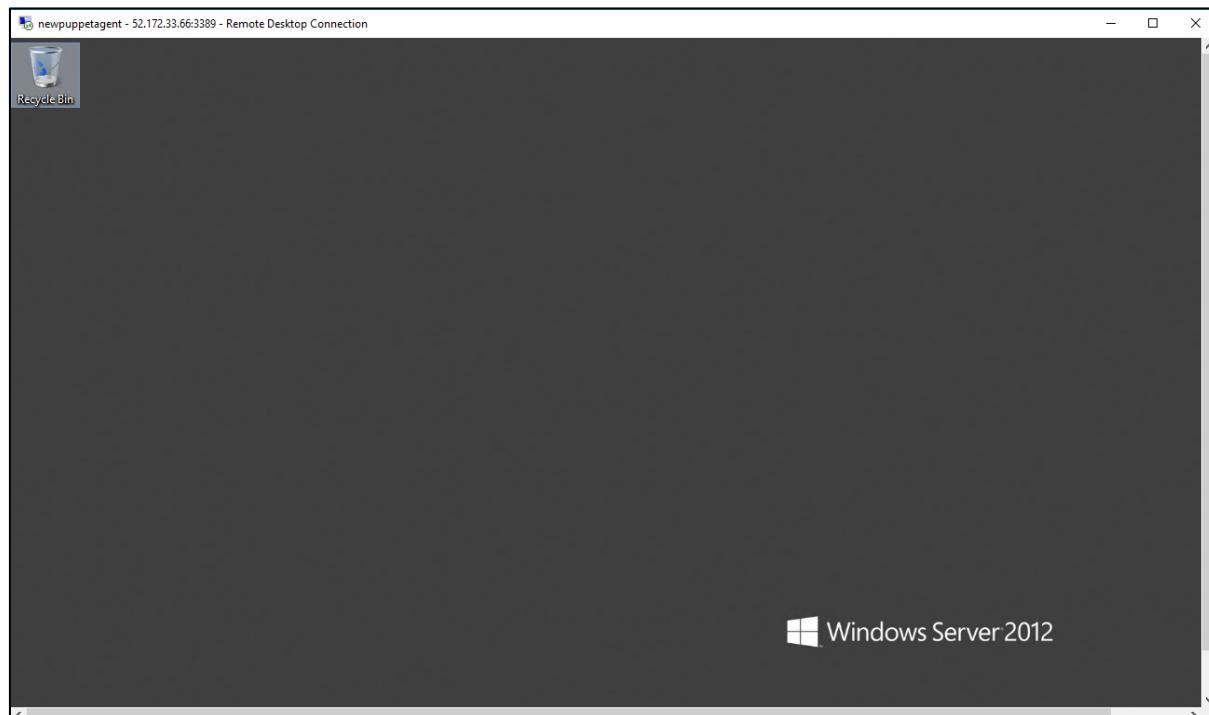
Step 18: On the left menu under **Node Management**, click **Unsigned Certificates** to open the node requests view:

Node name	Fingerprint	Actions
newpuppetagent.dr03afa4xciluvbcgx1wmnhtz...	D8:5E:47:C9:A7:D9:19:39:05:F6:14:10:81:5B:80...	Accept Reject

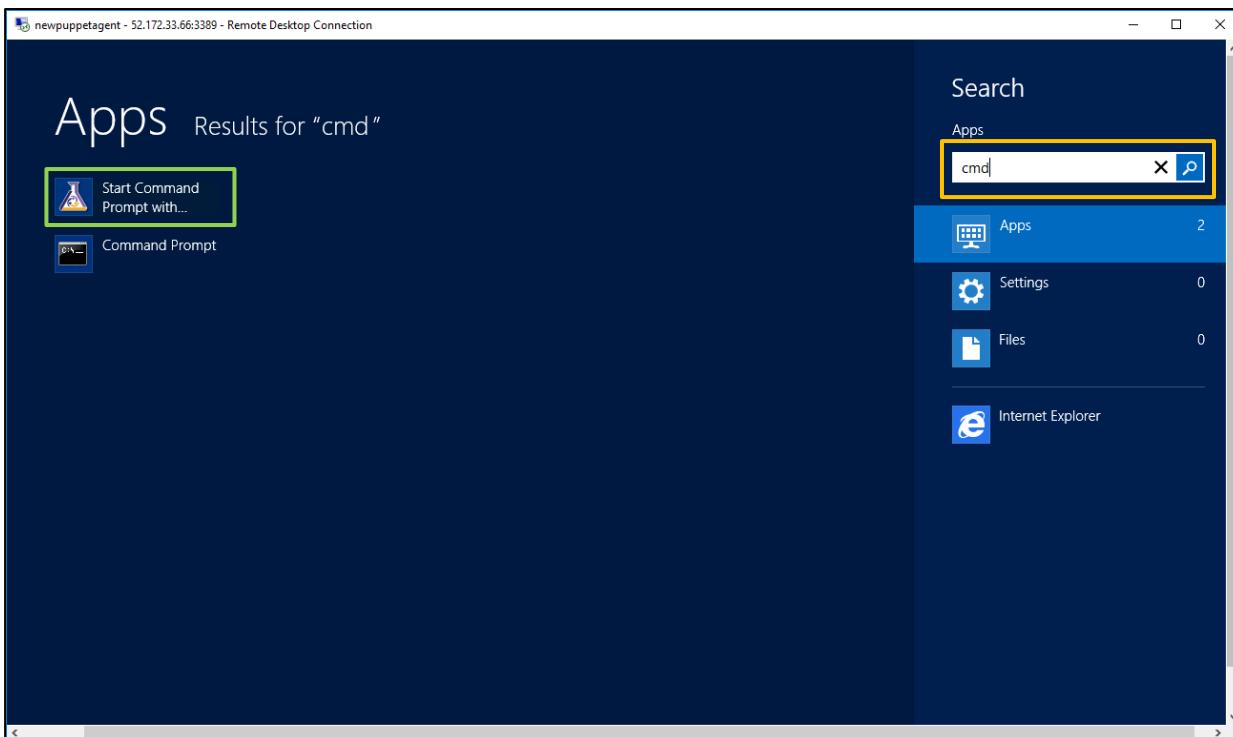
Step 19: To know what to do if your Puppet agent isn't connecting to your puppet master, Open your **Puppet Agent VM**:

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various service icons like Dashboard, All resources, Resource groups, App Services, etc. The main area is titled 'newpuppetagent' and shows the 'Virtual machine' details. At the top, there are several action buttons: 'Search (Ctrl+)/', 'Connect' (highlighted with a yellow box), 'Start', 'Restart', 'Stop', 'Capture', 'Move', 'Delete', and 'Refresh'. Below these are resource group, status, location, subscription, and public IP address details. On the right, there are two performance charts: 'CPU (average)' and 'Network (total)'. The CPU chart shows a small peak around 12:30 PM. The network chart shows a large spike for 'NETWORK OUT' at approximately 12:45 PM.

Step 20: After opening the RDP file, Login to the Puppet Agent VM with the credentials:



Step 21: Search cmd, right-click Start Command Prompt with Puppet and click Run as administrator:



Step 22: Run the below command to verify the correct connection to the Puppet Master Node: `puppet agent --configprint server`

