Microsoft Cloud Workshop

Business continuity and disaster recovery

Hands-on lab step-by-step

March 2018

Contents

Business continuity and disaster recovery hands-on lab step-by-step

Abstract and learning objectives

In this workshop, the student will gain experience designing solutions using Azure business continuity and disaster recovery (BCDR) technologies. Three different types of environments will be examined. The first will consist of on-premises VMs running applications that will be migrated to Azure IaaS. Next, Azure IaaS applications that need to be failed over from either on-premises to Azure or between two Azure Regions. Finally, the use of automated failover technologies built into Azure PaaS services App Service and SQL Database will be used for PaaS applications.

Learning Objectives:

- Understanding the different use cases for implementing Azure Site Recovery (ASR), for on-premises VM migration to Azure IaaS
- Determine how to leverage various Azure technologies together to build a complex and robust IaaS BCDR plan this consists of using Azure Site Recovery, Azure Automation, Traffic Manager and SQL Server Always On Availability Groups (VMs), to failover and failback an Azure IaaS IIS application and database from one Azure Region to another
- Design for high-availability and using BCDR techniques with Azure PaaS including Traffic Manager, SQL Database Failover Groups with Azure App Services

Overview

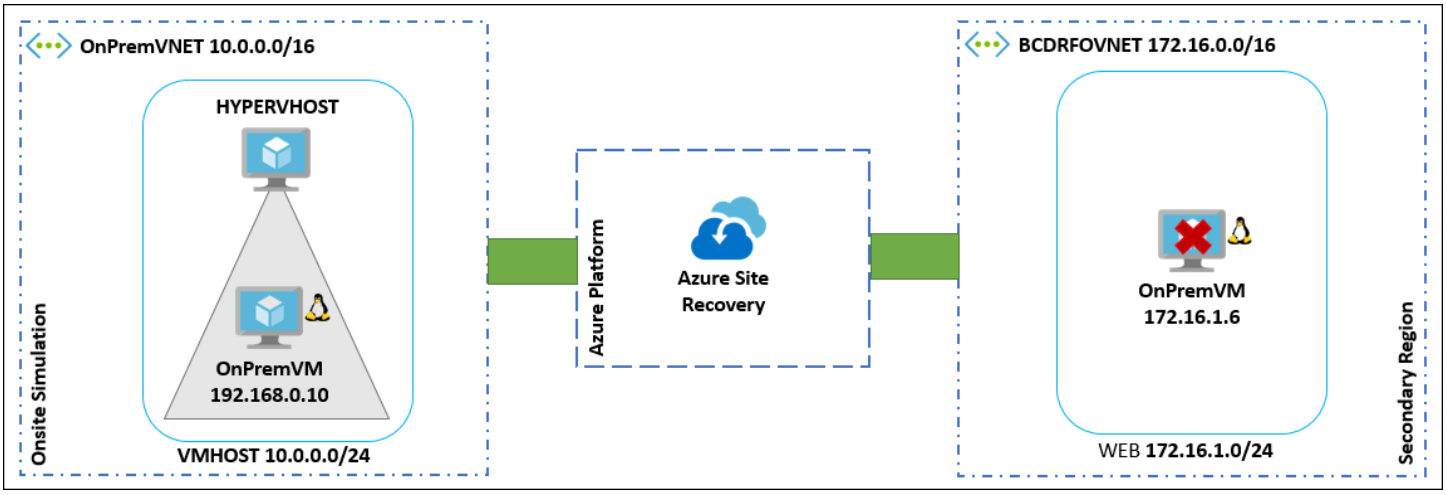
In this hands-on lab (HOL), attendees will implement three different environments and use Azure BCDR technologies to achieve three distinct goals for each environment type. These will include a migration to Azure, Azure region to region failover using Azure Site Recovery (ASR) and a PaaS implementation using BCDR technologies to ensure high availably of an application.

Solution architecture

Below are diagrams of the solution architecture you will build in this lab. Please study this carefully, so you understand the whole of the solution as you are working on the various components.

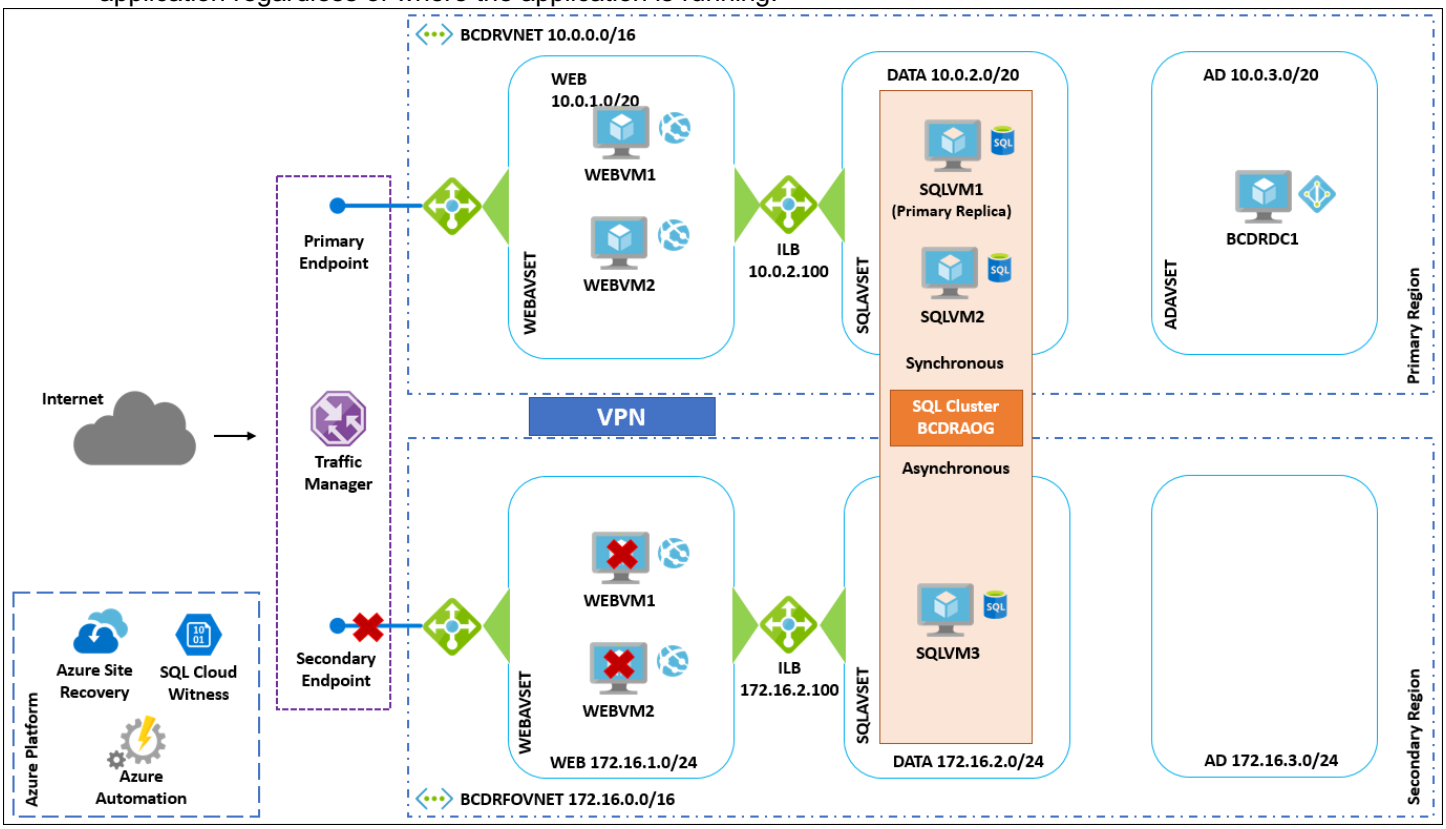
Environment: On-premises (migrate to Azure)

- **Background:** This environment will deploy a Hyper-V Host that will host a Linux VM to simulate a Linux, Apache, PHP and MySQL (LAMP), based web application deployed into an on-premises datacenter on a single VM
- **Goal using Azure BCDR:** Your goal for this environment will be to migrate this application to Azure IaaS with a one-direction failover



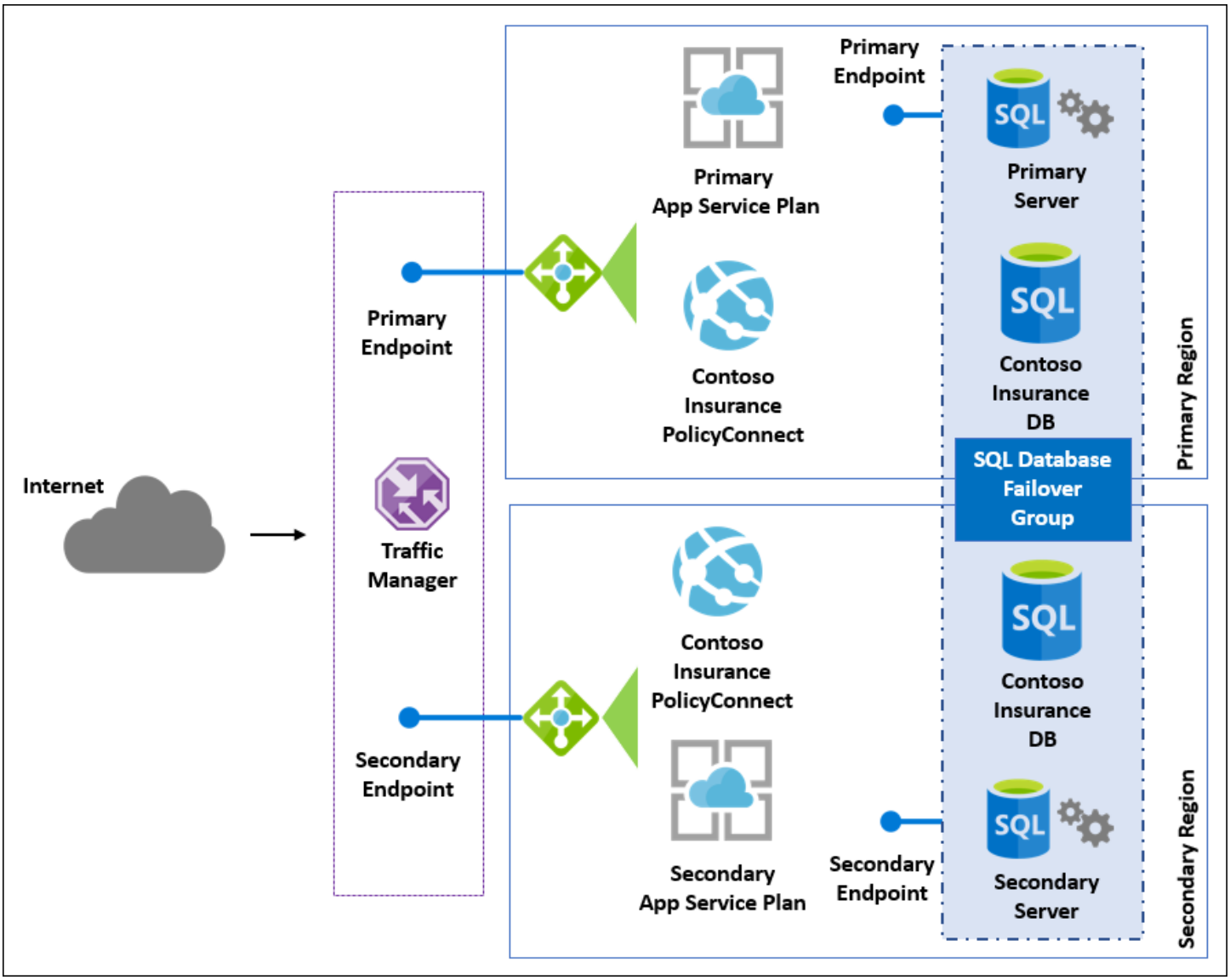
Environment: Azure IaaS (failover region to region)

- **Background:** This environment will consist of two Virtual Networks deployed to your Primary and Secondary site with an AD Domain, IIS Web Servers and Microsoft SQL Servers that you will configure into a SQL Always On Availability Group
- **Goal using Azure BCDR:** Your goal for this environment is to have the ability to have a one-click failover process using Azure Site Recovery in either direction. The users will have one URL that they will use to connect to your application regardless of where the application is running.



Environment: Azure PaaS (high-availably with seamless failover)

- **Background:** This environment will deploy an Azure Web App and Azure SQL Server in both the Primary and Secondary locations. You will configure SQL Database Failover groups to allow for seamless failover of the database.
- **Goal using Azure BCDR:** Your goal for this environment is never to have the users experience any downtime if issues arise with your Web App or the SQL Database. The users will have one URL that they will use to connect to your application regardless of where the application or database is running.



Requirements

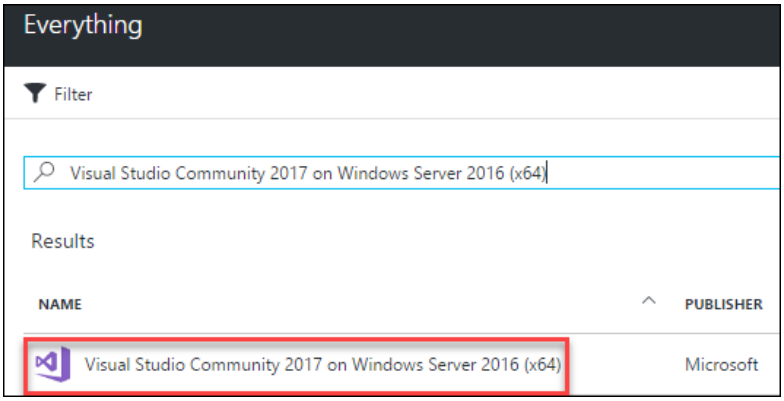
1. Azure Subscription with full access to the environment

Before the hands-on lab

Duration: 20 minutes

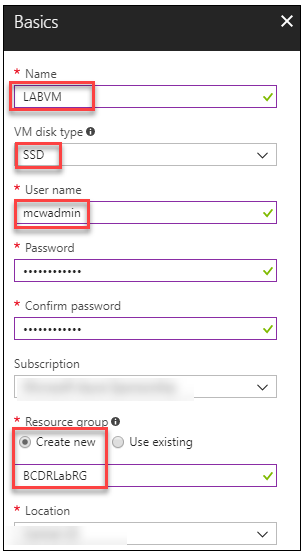
Task 1: Create a virtual machine to execute the lab

1. Launch a browser and navigate to the Azure Global portal at https://portal.azure.com. Once prompted, login with your Microsoft Azure credentials. If prompted, choose whether your account is an organization account or just a Microsoft Account.
2. Select **+NEW**, and in the search box enter in **Visual Studio Community 2017 on Windows Server 2016 (x64)** and press Enter. Select the Visual Studio Community 2017 image running on Windows Server 2016 and with the latest update.
3. In the returned search results select the image name



4. Select **Create**
5. Set the following configuration on the Basics tab and select **OK**
    a. **Name**: LABVM
    b. **VM disk type**: SSD
    c. **User name**: mcwadmin
    d. **Password**: demo@pass123
    e. **Subscription**: If you have multiple subscriptions choose the subscription to execute your labs in.
    f. **Resource Group**: BCDRLabRG

g. **Location**: Choose the closest Azure region to you



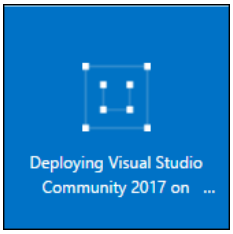
6. Choose the **D2S_V3 Standard** instance size on the Size blade.

Note: If the Azure Subscription you are using has limits on the number of cores you may wish to choose DS1_V2.

7. On the **Settings** blade accept the defaults and select **OK**

8. Select **Summary** blade select **Create**

9. Accept the remaining default values on the Settings blade and select **OK**. On the Summary page, select **Create**. The deployment should begin provisioning. It may take 10+ minutes for the virtual machine to complete provisioning.



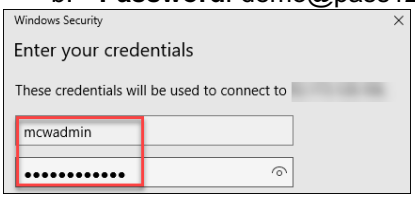
Note: Please wait for the LABVM to be provisioned prior to moving to the next step.

10. Move back to the Portal page on your local machine and wait for **LABVM** to show the Status of **Running**. Select **Connect** to establish a new Remote Desktop Session

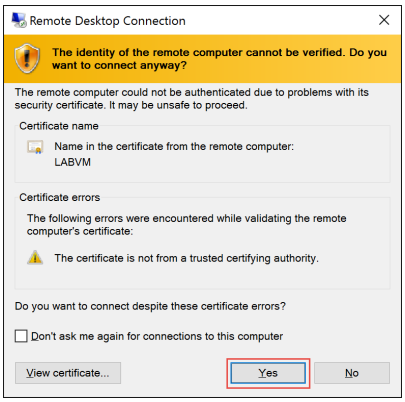
11. Depending on your remote desktop protocol client and browser configuration you will either be prompted to open an RDP file, or you will need to download it and then open it separately to connect

12. Log in with the credentials specified during creation:

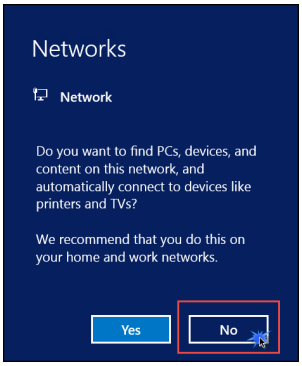
    a. **User**: mcwadmin
    b. **Password**: demo@pass123



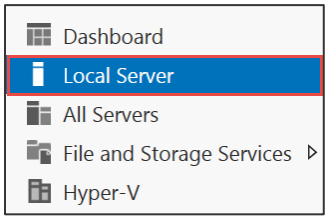
13. You will be presented with a Remote Desktop Connection warning because of a certificate trust issue. Select **Yes** to continue with the connection.



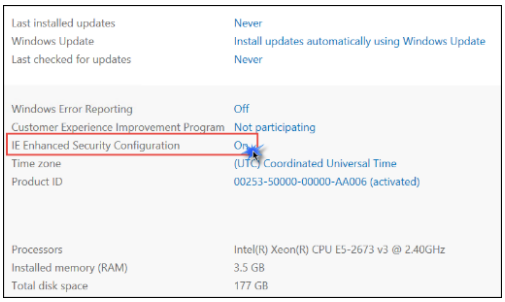
14. When logging on for the first time you will see a prompt on the right asking about network discovery. Select **No**.



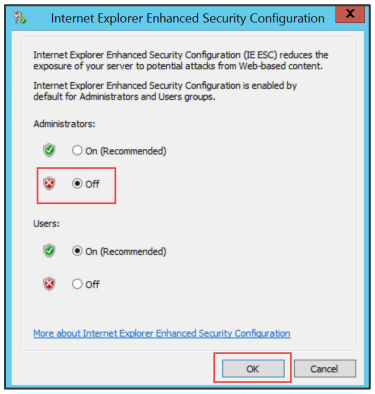
15. Notice that Server Manager opens by default. On the left, select **Local Server**.



16. On the right side of the pane, select **On** for **IE Enhanced Security Configuration**



17. Change **Administrators** to **Off** and select **OK**



Task 2: Download hands-on lab step by step support files to LABVM

1. After the reboot has completed, download the zipped Hands-on Lab Step by Step student files by clicking on this link:

```
https://www.dropbox.com/s/sx91sjcn63t980j/StudentFiles.zip?dl=1
```

2. Extract the downloaded files into the directory **C:\HOL**

Task 3: Install SQL Server Express on LABVM

1. From within LABVM open Internet Explorer and browse to the following URL:

```
https://www.microsoft.com/en-US/sql-server/sql-server-downloads
```

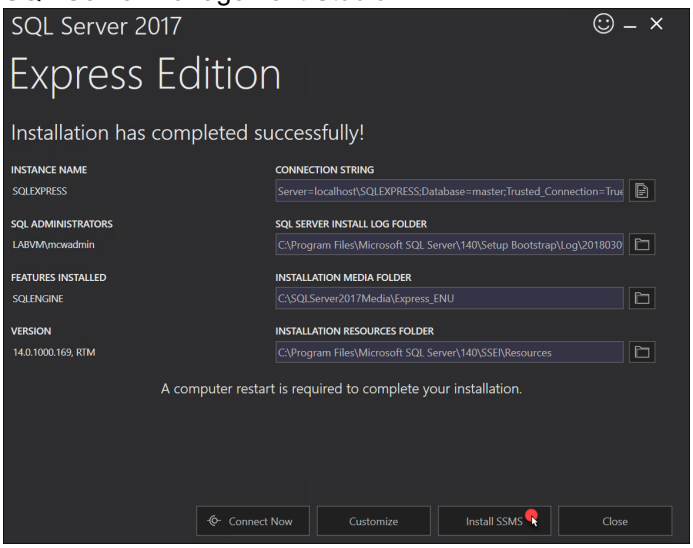
2. Click **Download now** under the Express edition

3. Click **Run**

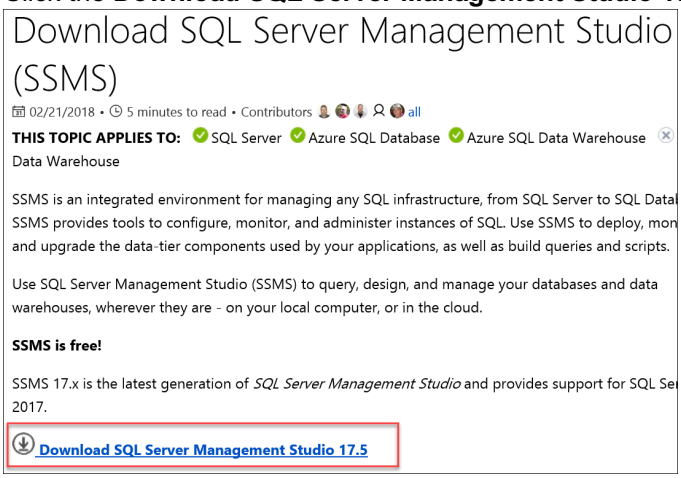
4. When the installer starts, click **Basic**

5. Accept the other defaults in the install wizard until SQL starts to install

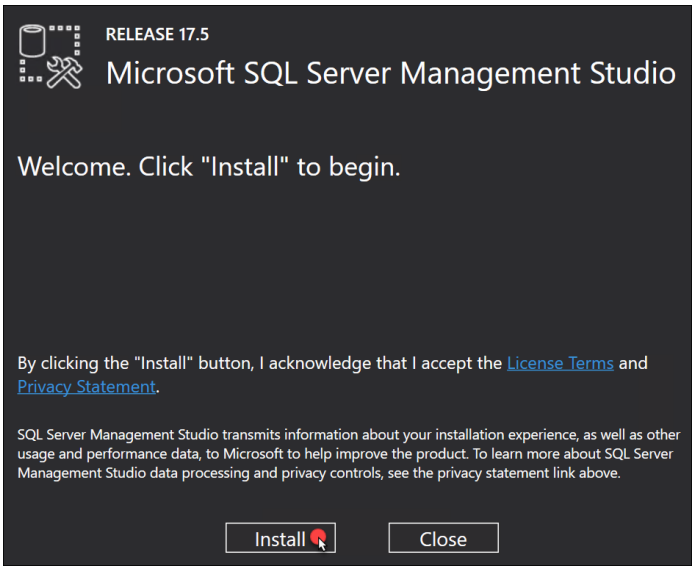
6. Once the install completes, click the **Install SSMS** button. This will webpage where you can download and install SQL Server Management Studio.



7. Click the **Download SQL Server Management Studio 17.X** link. When prompted click **Run**.



8. Click **Install**



9. Click **Close**

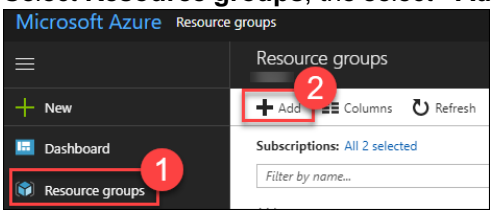
Task 4: Create the resource groups

In this task, you will select a **Primary** and **Secondary** Azure Region that will be used for the remainder of the HOL. The **Primary** region should be able to **support V3 virtual machine** sizes, and then you should select the **Secondary** region based on the region pair assigned by Microsoft. Use the Products available by region webpage to determine your **Primary** site: https://azure.microsoft.com/en-us/regions/services/. Once you have selected the Primary site, then review the BCDR page to find your Primary Site's Region Pair: https://docs.microsoft.com/en-us/azure/best-practices-availability-paired-regions.

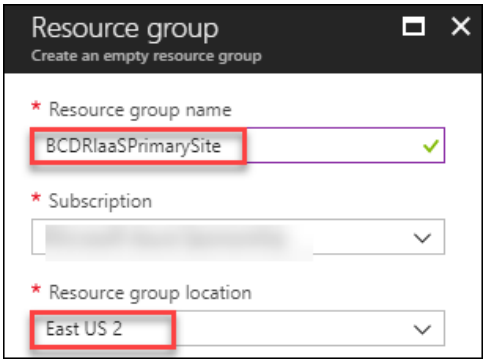
Note: The examples in this HOL Guide use these regions: **Primary:** East US 2 and **Secondary**: Central US.

1. From the **LABVM**, open Internet Explorer and connect to the Azure portal at: https://portal.azure.com

2. Select **Resource groups**, the select **+Add**

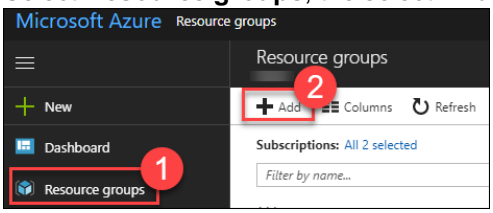


3. Complete the **Resource group** blade using the following inputs and select **Create**
    a. **Resource group name**: BCDRIaaSPrimarySite
    b. **Subscription**: Select your Subscription
    c. **Resource group location**: Select a Region for the Primary location.

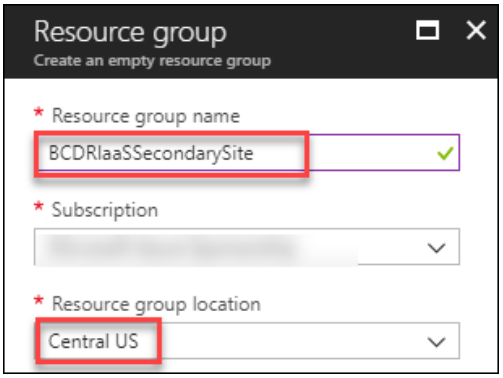


Note: It's very important for you to use these exact names. Changing the names of the Resource groups will impact the HOL setup and could cause you to not be able to complete the lab.

4. Select **Resource groups**, the select **+Add**



5. Complete the **Resource group** blade using the following inputs and select **Create**
    a. **Resource group name**: BCDRIaaSSecondarySite
    b. **Subscription**: Select your Subscription
    c. **Resource group location**: Select a Region for the Secondary location



6. Continue to add resource groups to support the HOL

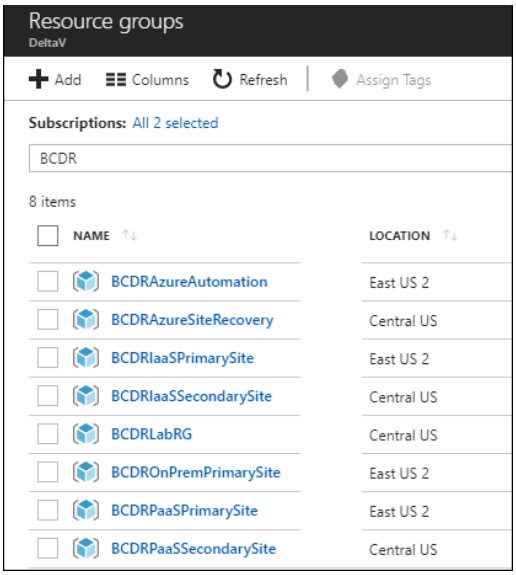
Resource Group Name	Location
BCDRAzureAutomation	Any available site other than your Primary or Secondary
BCDRAzureSiteRecovery	Secondary
BCDROnPremPrimarySite	Primary
BCDRPaaSPrimarySite	Primary
BCDRPaaSSecondarySite	Secondary

7. Once all the resource groups have been created, you should review all the resource groups for this HOL. **It is critical to ensure that the spelling is correct and that they are in the correct Azure Regions (Primary or Secondary).**

Note: If for some reason there is an error, you should delete the resource group with the error and recreate it.

Resource Group Name	Location
BCDRAzureAutomation	Any available site other than your Primary or Secondary
BCDRAzureSiteRecovery	Secondary
BCDRIaaSPrimarySite	Primary
BCDRIaaSSecondarySite	Secondary
BCDROnPremPrimarySite	Primary
BCDRPaaSPrimarySite	Primary
BCDRPaaSSecondarySite	Secondary

8. Here is the Azure Portal with each of the resource groups created in the correct Azure Region



You should follow all steps provided *before* attending the Hands-on lab.

Exercise 1: Deploy Azure environments

Duration: 15 minutes (Deployments can take as long as 75 minutes)

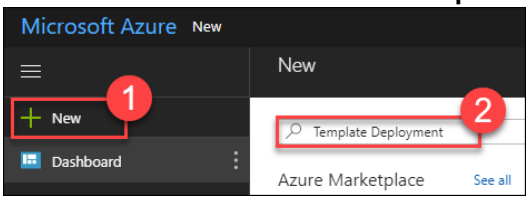
In this exercise, you will use Azure ARM Templates to deploy the following environments that will be used in this HOL:

- **Azure IaaS:** This environment will consist of two Virtual Networks deployed to your Primary and Secondary site with an AD Domain, IIS Web Servers and Microsoft SQL Servers that you will configure into a SQL Always On Availability Group
- **On-premises:** This environment will deploy a Hyper-V Host that will host a Linux VM to simulate a web application deployed into on-premises datacenter on a single VM. Your goal for this environment will be to migrate this application to Azure IaaS with a one-direction failover
- **Azure PaaS:** This environment will deploy an Azure Web App and Azure SQL Server in both the Primary and Secondary locations

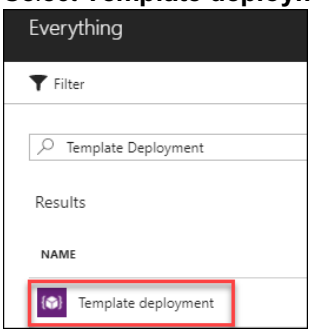
Task 1: Deploy Azure IaaS

1. From the **LABVM**, open Internet Explorer and connect to the Azure portal at: https://portal.azure.com

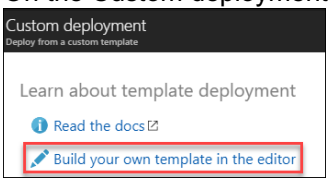
2. Select **+New** and then search for **Template Deployment**



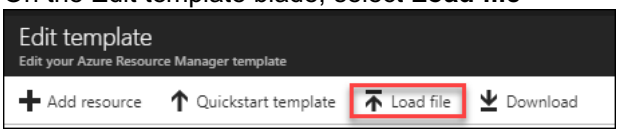
3. Select **Template deployment** and then **Create**



4. On the Custom deployment blade, select **Build your own template in the editor**

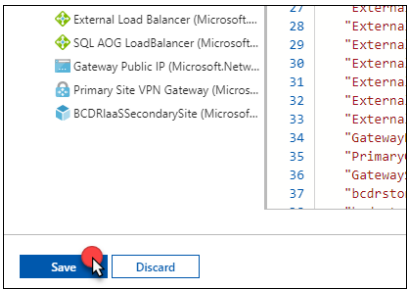


5. On the Edit template blade, select **Load file**

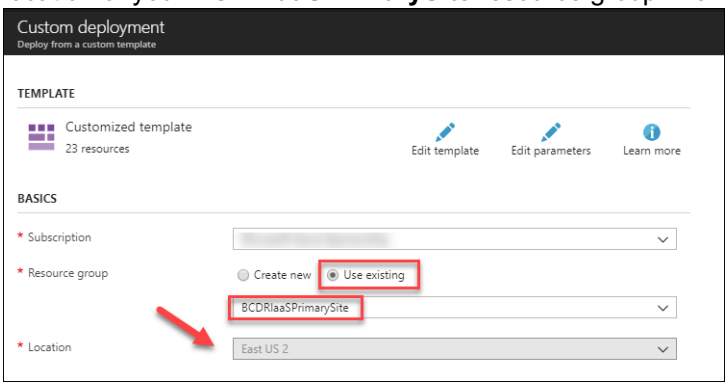


6. From the **C:\HOL\Deployments** directory locate the **BCDRIaaSPrimarySite.json** file and select **Open**

7. This will load the template into the Azure portal. Select **Save**.

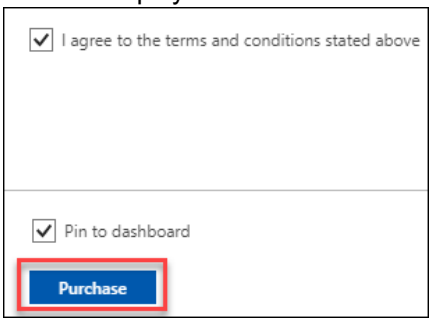


8. On the Custom deployment blade, next to **Resource group** select **Use existing** and then select your **BCDRIaaSPrimarySite** resource group. Notice how the template picked the deployment region based on the location of your **BCDRIaaSPrimarySite** resource group. Make sure that this is your **Primary** region.



9. Next, update the **Domain Controller DNS Name** in the **Settings** area. This will be the DNS name for the Active Directory Domain controller that will be your jump box into the IaaS environment. The name will need to be lowercase and 3-24 characters consisting of letters & numbers and be unique to all of Azure. In the example the name bcdrdc18675309 was used.

10. Finally, select **I agree to the terms and conditions stated above** and **Pin to dashboard.** Select **Purchase** to start the deployment.

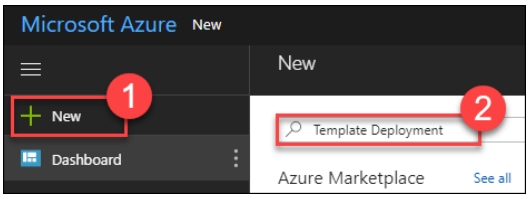


Note: This deployment will take at least 60 minutes, but you can continue to the next task.

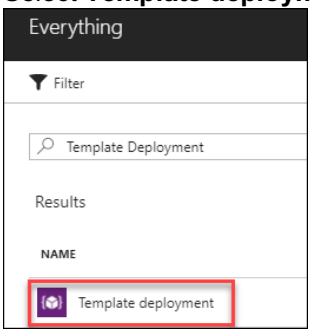
Task 2: Deploy on-premises environment

1. From the **LABVM**, open Internet Explorer and connect to the Azure portal at: https://portal.azure.com

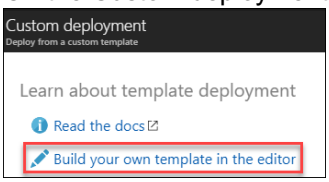
2. Select **+New** and then search for **Template Deployment**



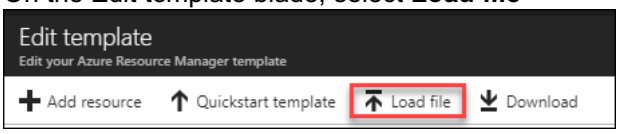
3. Select **Template deployment** and then **Create**



4. On the Custom deployment blade, select **Build your own template in the editor**



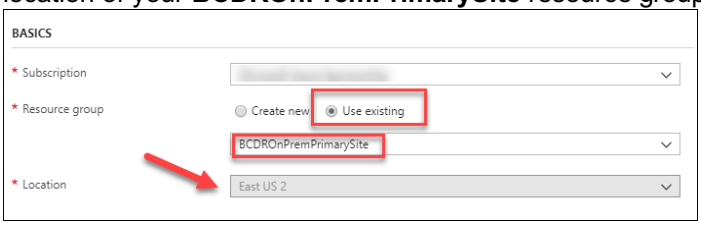
5. On the Edit template blade, select **Load file**



6. From the **C:\HOL\Deployments** directory locate the **BCDROnPremPrimarySite.json** file and select **Open**

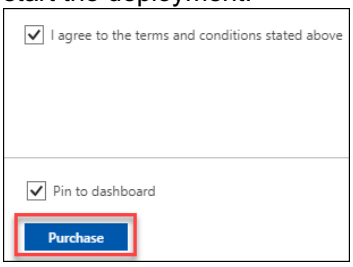
7. This will load the template into the Azure portal. Select **Save**

8. On the Custom deployment blade, next to **Resource group** select **Use existing** and then select your **BCDROnPremPrimarySite** resource group. Notice how the template picked the deployment region based on the location of your **BCDROnPremPrimarySite** resource group. Make sure that this is your **Primary** region.



9. Next, update the **Hyper-V Host DNS Name** in the **Settings** area. This will be the DNS name for the Hyper-V Host that will you will use for this simulated on-premises environment. The name will need to be lowercase and 3-24 characters consisting of letters & numbers and be unique to all of Azure. In the example, the name hypervhost8675309 was used.

10. Finally, select **I agree to the terms and conditions stated above** and **Pin to dashboard.** Select **Purchase** to start the deployment.

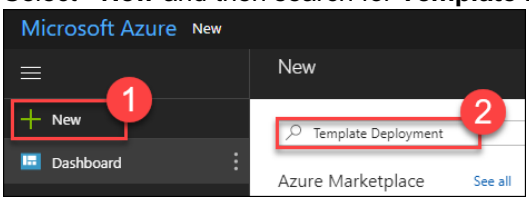


Note: This deployment will take at least 20 minutes, but you can continue to the next task.

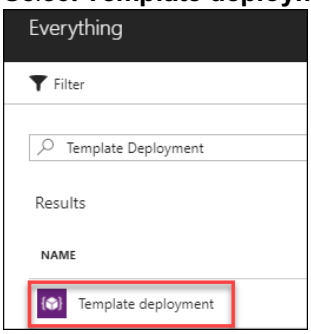
Task 3: Deploy Azure PaaS environment

1. From the **LABVM**, open Internet Explorer and connect to the Azure portal at https://portal.azure.com

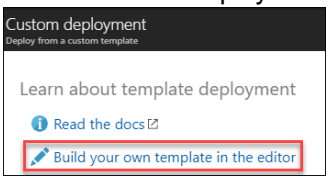
2. Select **+New** and then search for **Template Deployment**



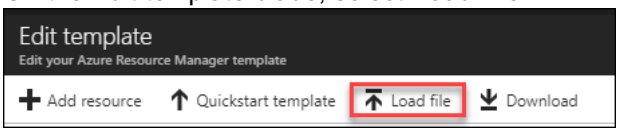
3. Select **Template deployment** and then **Create**



4. On the Custom deployment blade, select **Build your own template in the editor**



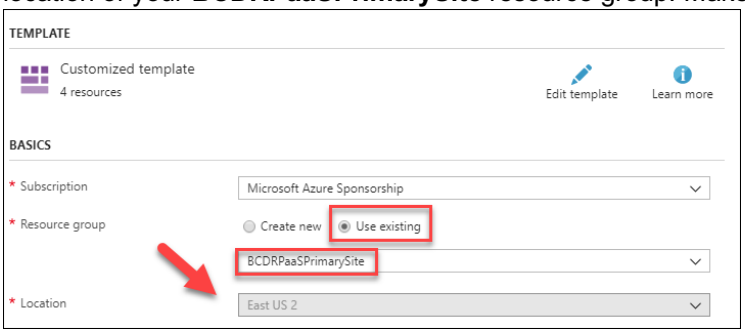
5. On the Edit template blade, select **Load file**



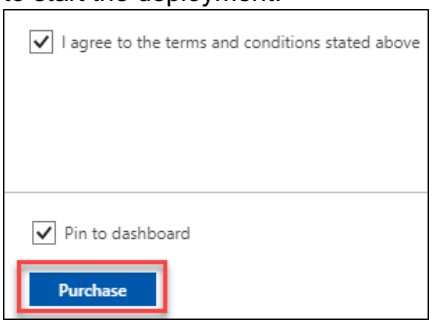
6. From the **C:\HOL\Deployments** directory locate the **BCDRPaaSPrimarySite.json** file and select **Open**

7. This will load the template into the Azure portal. Select **Save**.

8. On the Custom deployment blade, next to **Resource group** select **Use existing** and then select your **BCDRPaaSPrimarySite** resource group. Notice how the template picked the deployment region based on the location of your **BCDRPaaSPrimarySite** resource group. Make sure that this is your **Primary** region.



9. Finally, select the **I agree to the terms and conditions stated above** and **Pin to dashboard.** Select **Purchase** to start the deployment.

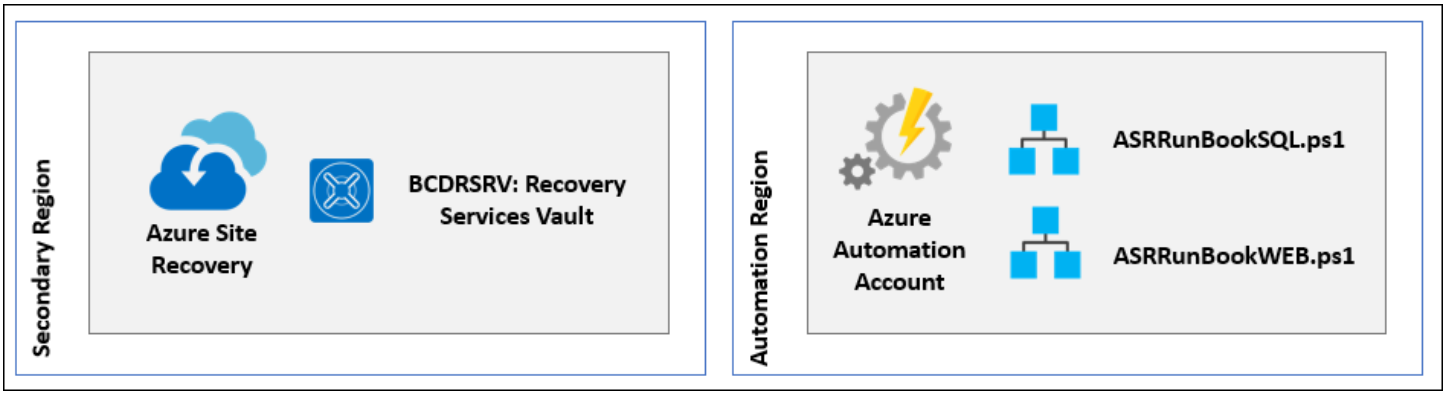


Note: This deployment will take at least 10 minutes, but you can continue to the next task.

Exercise 2: Configure BCDR services

Duration: 30 minutes

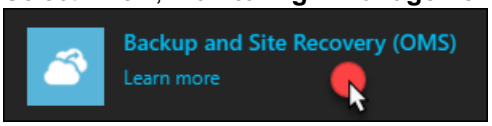
In this exercise, you will create and configure the services that will make it possible to failover both the on-premises and Azure IaaS environments. These will include a Recovery Services Vault used for Azure Site Recovery and an Azure Automation account.



Task 1: Create Azure recovery services vault

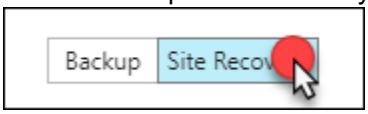
1. Using the **LABVM** connect to the Azure portal using your web browser at https://portal.azure.com

2. Select **+New**, **Monitoring + Management** and then **Backup and Site Recovery (OMS)**

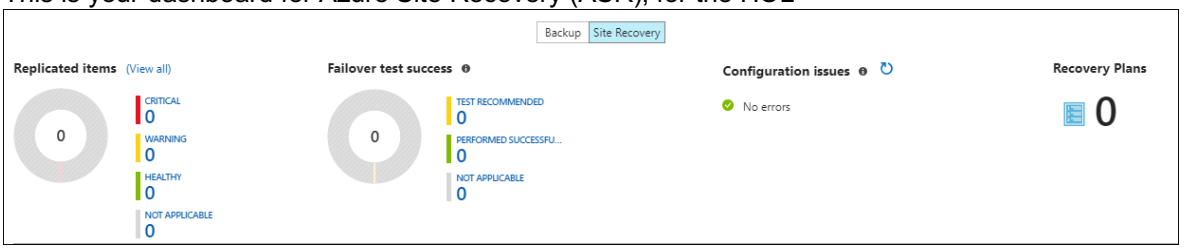


3. Complete the **Recovery Services vault** blade using the following inputs, then select **Create**
   a. **Name**: BCDRRSV
   b. **Resource Group**: Use existing / BCDRAzureSiteRecovery
   c. **Location**: Secondary Site (should be automatically selected based on your resource group)

4. Once the **BCDRRSV** Recovery Service Vault has been created, open it in the Azure portal. Toggle the switch that selects Backup / Site Recovery



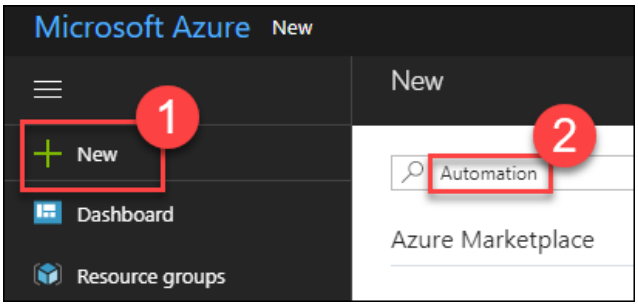
5. This is your dashboard for Azure Site Recovery (ASR), for the HOL



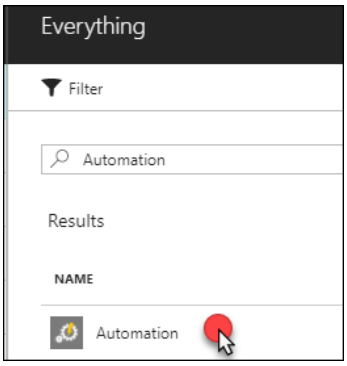
Task 2: Deploy Azure automation

1. Open the Azure portal at: https://portal.azure.com

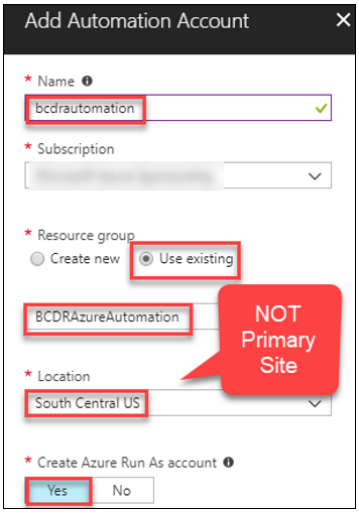
2. Select **+New** and then enter **Automation** in the search box



3. Select **Automation** and then **Create**

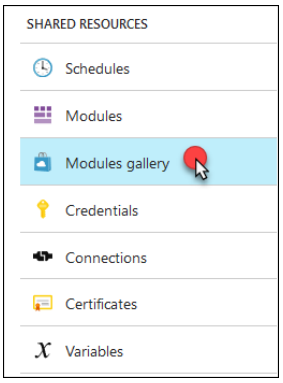


4. Complete the **Add Automation Account** blade using the following inputs and then select **Create**:
   a. **Name**: Enter a Globally unique name starting with BCDR
   b. **Resource group**: Use existing / BCDRAzureAutomation
   c. **Location**: Select a site in your Area, but NOT your Primary Site
   d. **Create Azure Run As account**: Yes

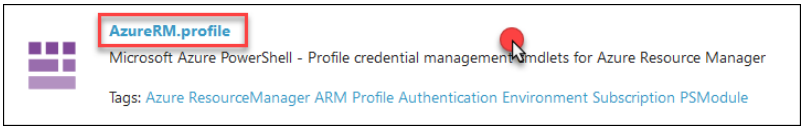


Note: Azure Automation accounts are only allowed to be created in certain Azure regions, but they can act on any region in Azure (except Government, China or Germany). It is not a requirement to have your Azure Automation account in the same region as the **BCDRAzureAutomation** resource group but **CANNOT** be in your primary site.

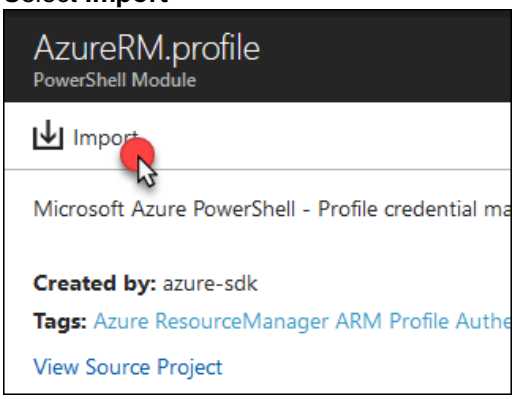
5. Once the Azure automation account has been created open the account and select **Modules gallery** under **Shared Resources**



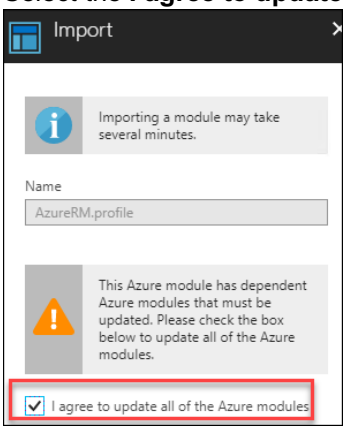
6. When the Modules load, scroll down and locate and select **AzureRM.Profile**



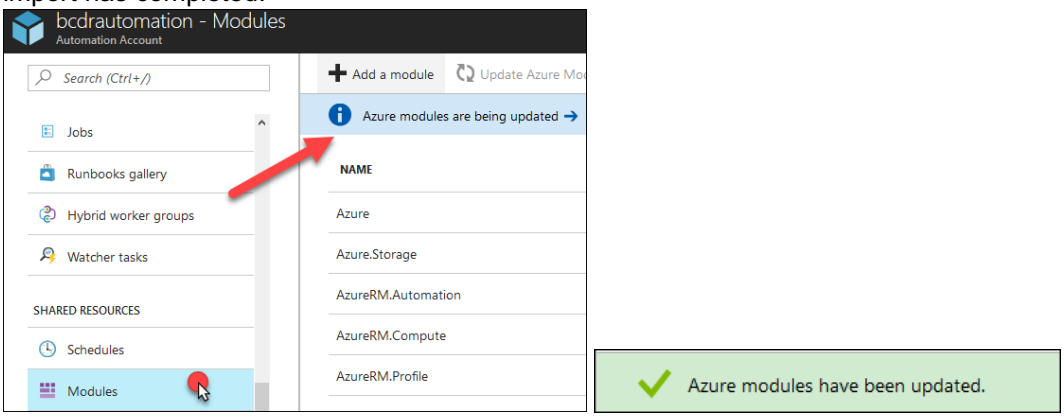
7. Select **Import**



8. Select the **I agree to update all of the Azure modules** and then select **OK**



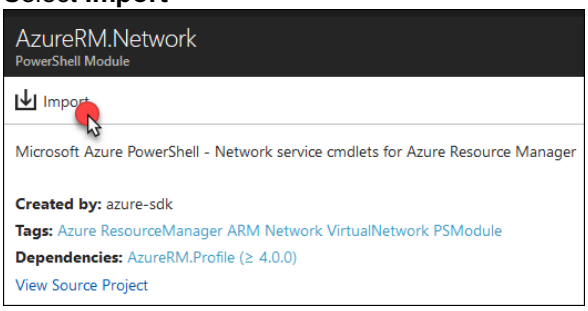
9. It will take a few minutes to update the modules. Select **Modules** under Shared Resources, and you can wait the import has completed.



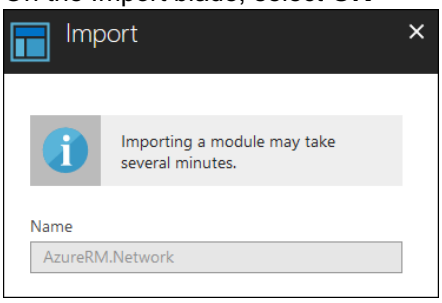
10. Select back to **Modules Gallery**

11. When the Modules load, scroll down and locate and select **AzureRM.Network**

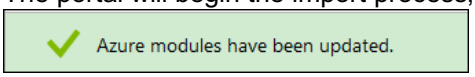
12. Select **Import**



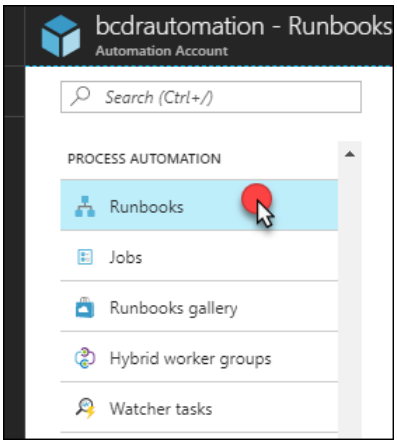
13. On the Import blade, select **OK**



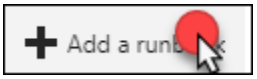
14. The portal will begin the import process, but should only take about a minute



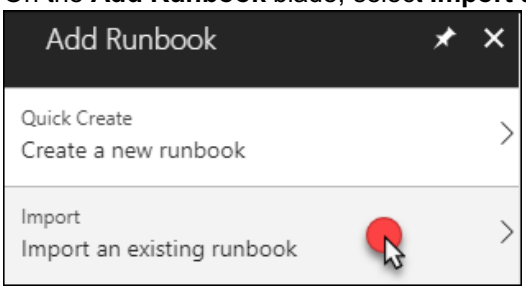
15. Next select **Runbooks**



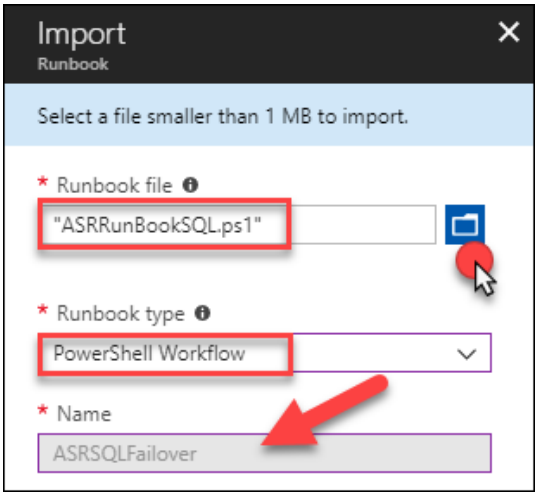
16. Select **+Add a runbook**



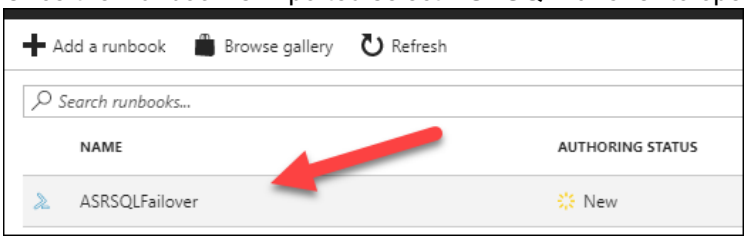
17. On the **Add Runbook** blade, select **Import an existing runbook**



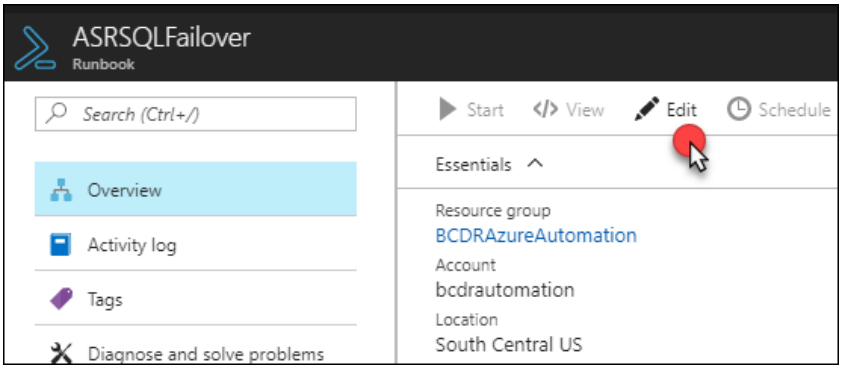
18. Select the **Folder** icon on the Import blade and select the file **ASRRunbookSQL.ps1** from the C:\HOL\Deployments directory on the **LABVM**. The Runbook type should default to **PowerShell Workflow**. Notice that the Name can't be changed. This is the name of the Workflow inside of the Runbook script. Select **Create**.



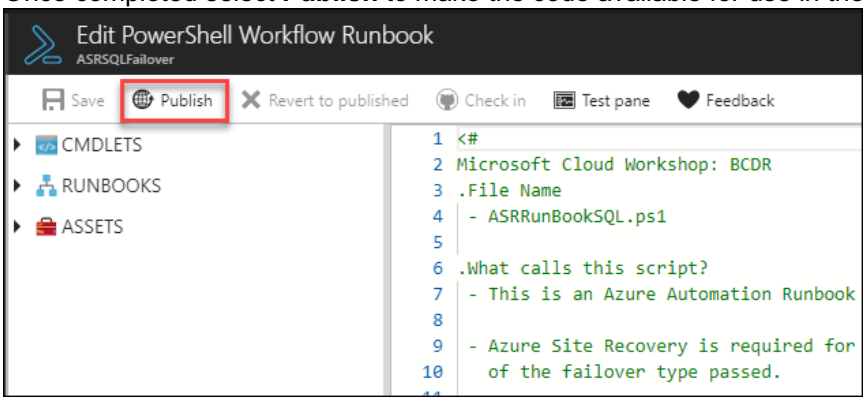
19. Once the Runbook is imported select **ASRSQLFailover** to open the runbook



20. On the **ASRSQLFailover** Runbook blade select **Edit**

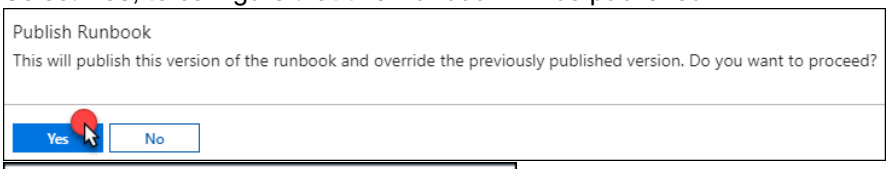


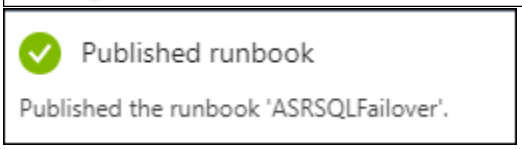
21. The PowerShell runbook will load. If you wish, you can review the comments to better understand the runbook. Once completed select **Publish** to make the code available for use in the portal.



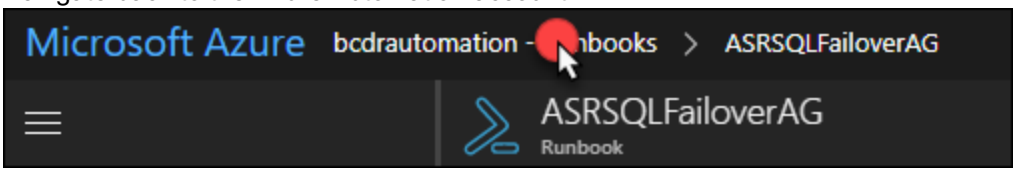
Note: You might notice the Test Pane link. This script can't be tested from here as there are more configurations required, and it relies of being called by Azure Site Recovery to feed its variables.

22. Select **Yes**, to configure that this Runbook will be published

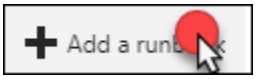




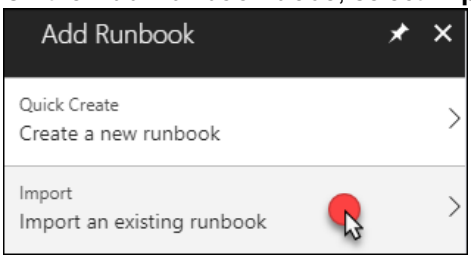
23. Navigate back to the Azure Automation account



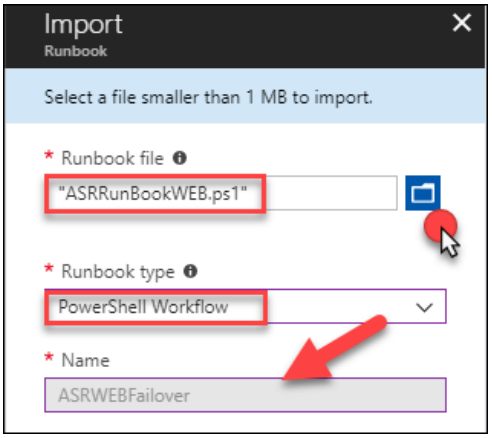
24. Select **+Add a runbook**



25. On the **Add Runbook** blade, select **Import an existing runbook**

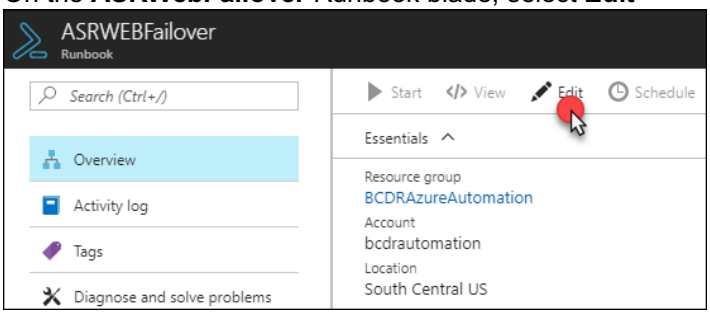


26. Select the Folder icon on the Import blade and select the file **ASRRunbookWEB.ps1** from the **C:\HOL\Deployments** directory on the **LABVM**. The Runbook type should default to **PowerShell Workflow**. Notice that the Name can't be changed. This is the name of the Workflow inside of the Runbook script. Select **Create**.

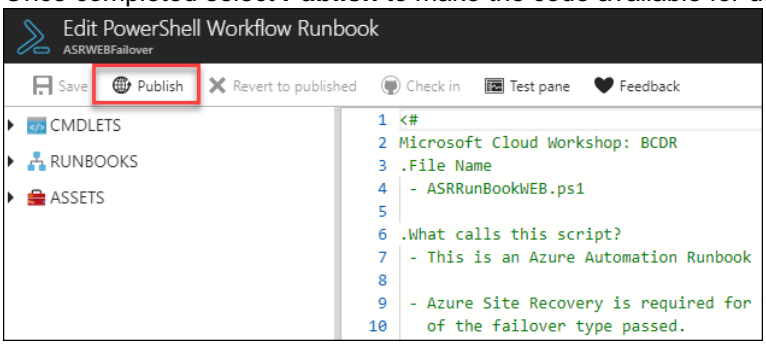


27. Once the Runbook is imported select **ASRWebFailover** to open the runbook

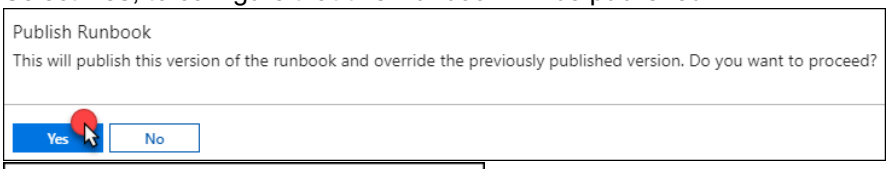
28. On the **ASRWebFailover** Runbook blade, select **Edit**



29. The PowerShell runbook will load. If you wish, you can review the comments to better understand the runbook. Once completed select **Publish** to make the code available for use in the portal.



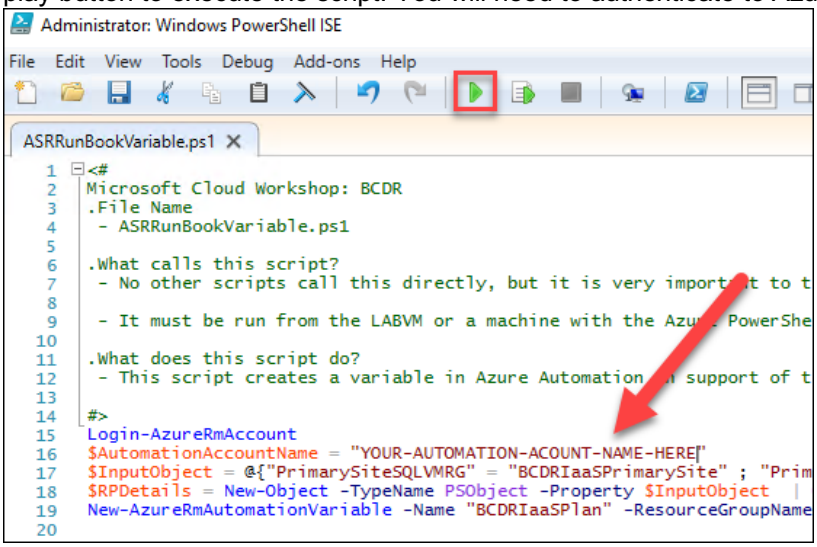
30. Select **Yes**, to configure that this Runbook will be published



31. Navigate back to **Runbooks**, make sure that both Runbooks show as "**Published"**

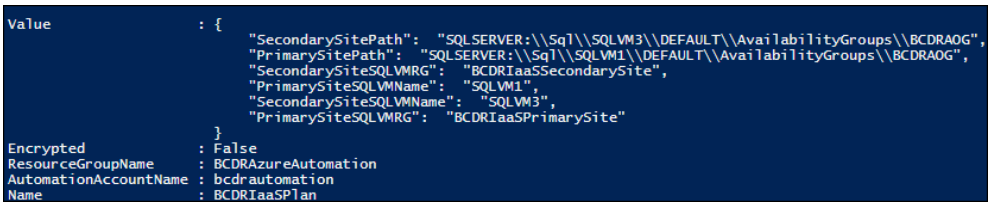
32. From **LABVM**, select **Start** and select **PowerShell ISE** (make sure to right-click and run as Administrator)

33. Open the file **C:\HOL\Deployments\ASRRunBookVariable.ps1.** Review the script and then select the green play button to execute the script. You will need to authenticate to Azure.

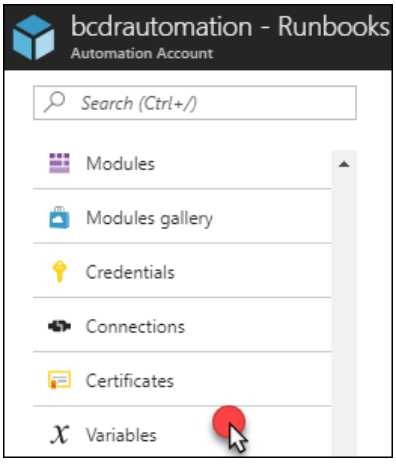


Note: If you are using an account that has multiple subscriptions you may need to change the context of your login. You can comment the **Login-AzureRmAccount** command in the script and complete the login and the use the **Get-AzureRmSubscription** and **Select-AzureRmSubscription** cmdlets to create the proper context for this HOL. For help, you can review this article: https://docs.microsoft.com/en-us/powershell/azure/authenticate-azureps?view=azurermps-5.1.1 & https://docs.microsoft.com/en-us/powershell/azure/manage-subscriptions-azureps?view=azurermps-5.1.1

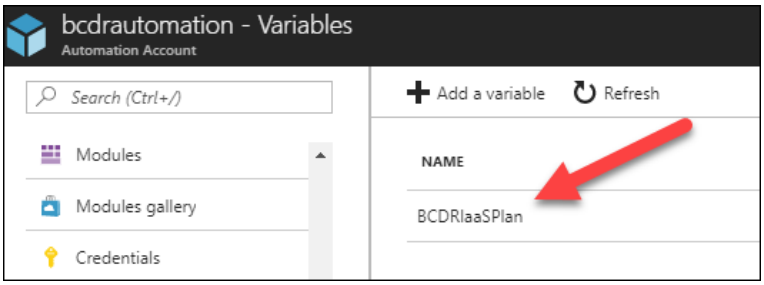
34. Once the script has run, you will see the following output from PowerShell ISE. This script created a variable that will be used with the PowerShell Runbook in Azure Automation to help with the Failover and Failback of the Azure IaaS environment.



35. Move back to the Azure portal and locate the **Azure Automation** account. Select the **Variables** link in the **Shared Resources** section.



36. Notice that the variable **BCDRIaaSPlan** has been created. This variable will be used along with ASR to orchestrate part of the failover.



Note: When you configure the ASR Recovery Plan for the IaaS deployment you will use the SQL Runbook as a Pre-Failover Action and the Web Runbook as a Post-Failover action. They will run both ways and have been written to take the "Direction," of the failover into account when running.

Exercise 3: Configure environments for failover

Duration: 90 minutes

In this exercise, you will configure the three environments to use BCDR technologies found in Azure. Each environment has unique configurations that must be completed to ensure their availability in the event of a disaster.

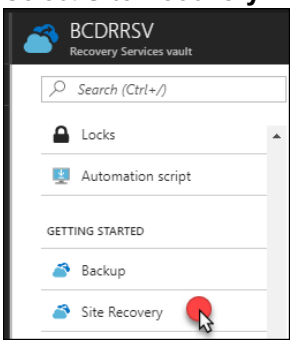
> **Note**: Make sure prior to starting each task that the deployment that you started in Exercise 1 has completed for each as you come to that task. This can be determined, but reviewing the deployments for each Resource group in the Azure portal. If it says Succeeded, then you can begin the task.

Task 1: Configure on-premises to Azure IaaS failover for migration

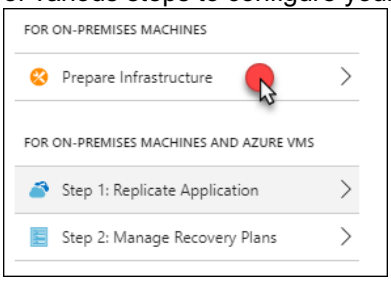
In this task, the **OnPremVM** will be configured to replicate to Azure and be ready to failover to the **BCDRIaaSSecondarySite**. This will consist of configuring your Hyper-V host with the ASR provider and then enabling replication of the VM to the Recovery Service Vault.

1. From the Azure portal, open the **BCDRRSV** Recovery Services Vault located in the **BCDRAzureSiteRecovery** resource group

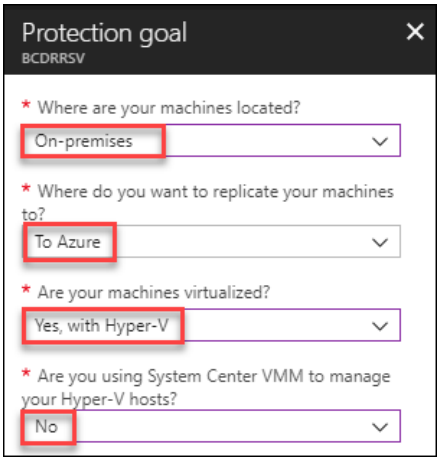
2. Select **Site Recovery** in the **Getting Started** area of **BCDRRSV** blade



3. Next, select **Prepare Infrastructure** in the **For On-Premises Machines** section. This will start you down a path of various steps to configure your VM that is running on Hyper-V on-premises to be replicated to Azure.

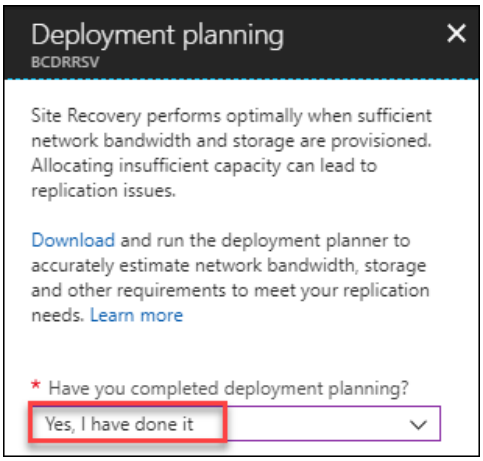


4. On **Step 1 Protection Goal** select the following inputs and then select **OK**:
    a. **Where are your machines located?**: On-premises
    b. **Where do you want to replicate your machines to?**: To Azure
    c. **Are your machines virtualized?**: Yes, with Hyper-V (your VM is running as a nested VM in Azure)
    d. **Are you using System Center VMM to manage your Hyper-V hosts?**: No

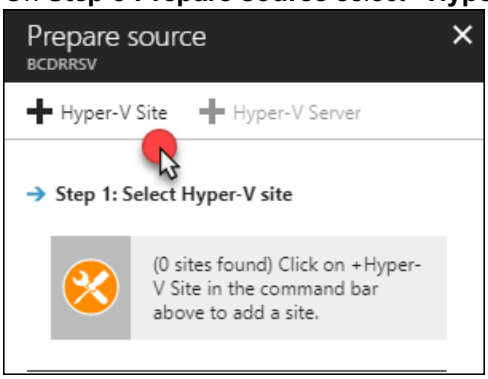


5. On **Step 2 Deployment planning** select the following inputs and then select **OK**:
    a. Have you completed deployment planning: **Yes, I have done it**

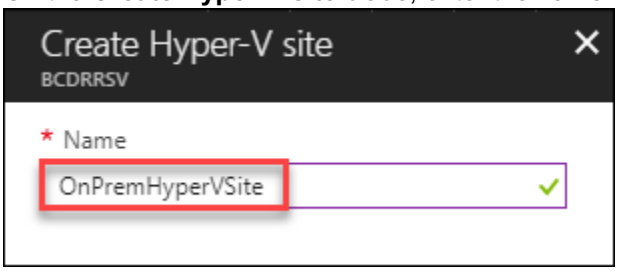
Note: You can read more about planning an ASR to deployment here: https://docs.microsoft.com/en-us/azure/site-recovery/site-recovery-hyper-v-deployment-planner



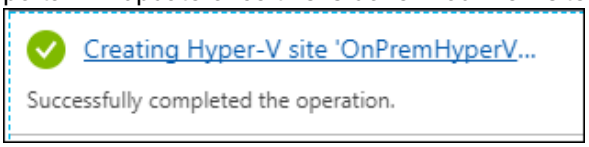
6. On **Step 3 Prepare source** select **+Hyper-V Site**

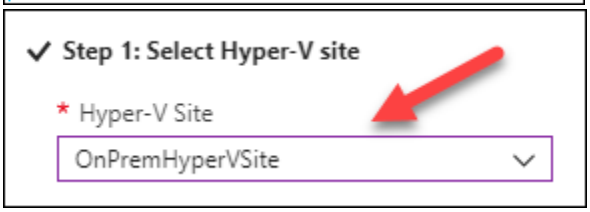


7. On the **Create Hyper-V site** blade, enter the name: **OnPremHyperVSite** and select **OK**

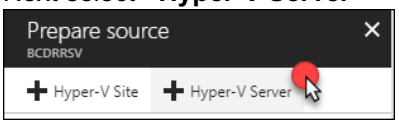


8. The portal will deploy the site providing you notifications. Wait for the creation process to complete, and the ASR portal will update once this is done. Your new site is now shown under **Step 1: Select Hyper-V site**.

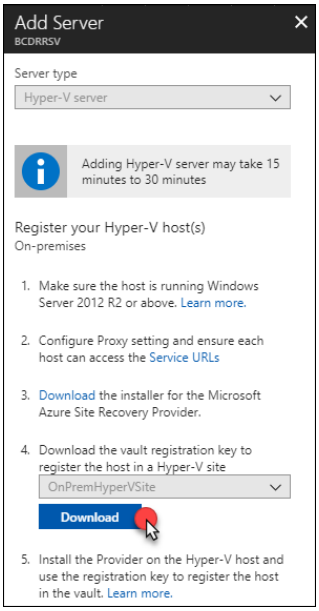




9. Next select **+Hyper-V Server**

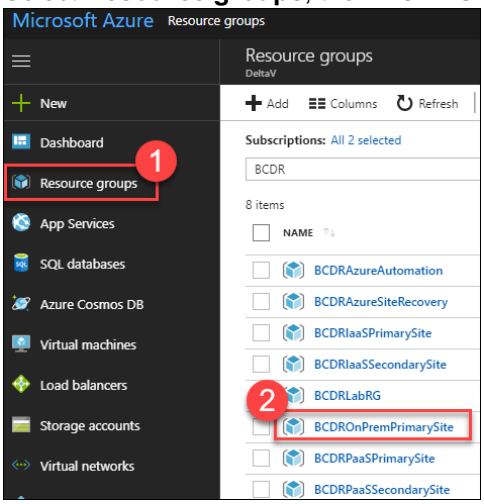


10. A new blade will appear. You will need to download the vault registration key to register the host in the Hyper-V site of ASR. Select the Download button which will save the file to your Downloads folder on the **LABVM**.



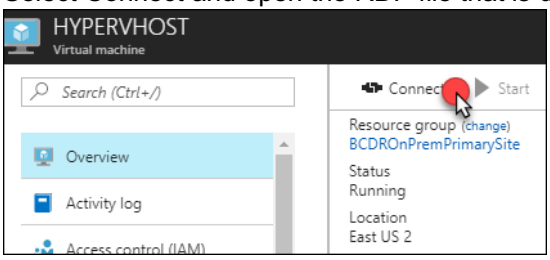
11. Open a **NEW** tab in your web browser and connect again to the Azure Portal at https://portal.azure.com

12. Select **Resource groups**, then **BCDROnPremPrimarySite**



13. Locate and select on the **HYPERVHOST** VM object

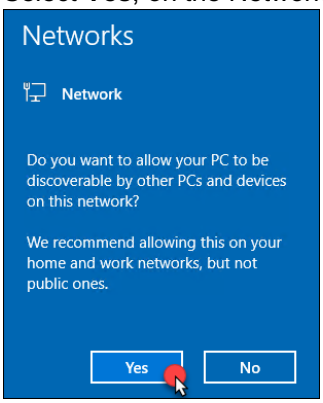
14. Select Connect and open the RDP file that is downloaded



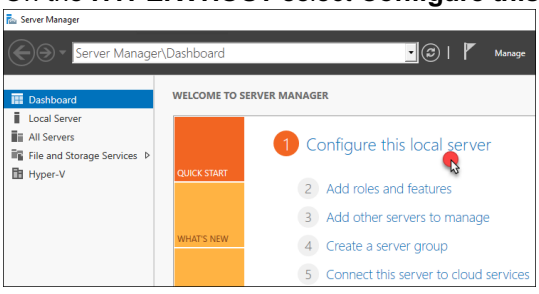
15. Enter the credentials for the VM:
    b. **User Name**: mcwadmin
    c. **Password**: demo@pass123

16. You will be prompted with a warning about a certificate. Select **Yes** to connection (you can always select yes to these prompts during this HOL).

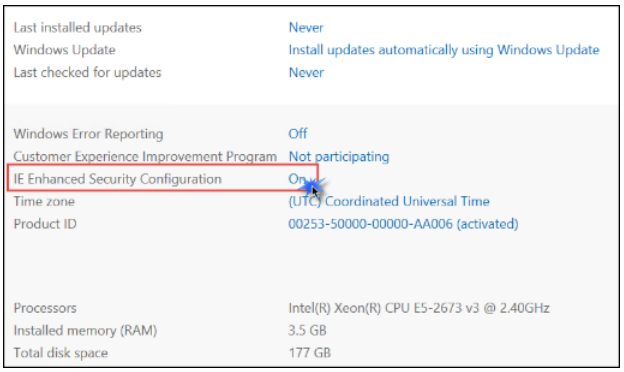
17. Select **Yes**, on the Networks prompt



18. On the **HYPERVHOST** select **Configure this local server** in the Server Manager Dashboard



19. On the right side of the pane, select **On** by **IE Enhanced Security Configuration**.

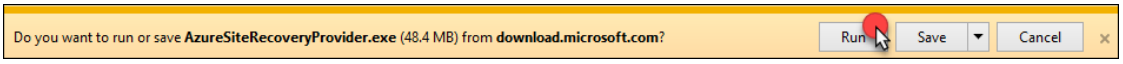


20. Change to **Off** for Administrators and select **OK**

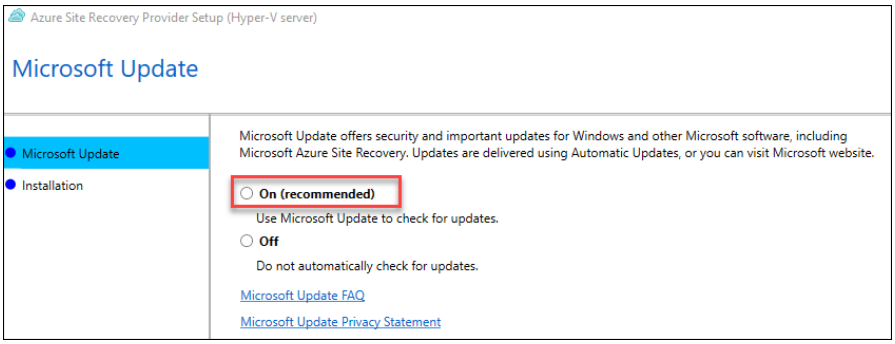
21. Open Internet Explorer on **HYPERVHOST** and browse to the following URL. This will download the Azure Site Recovery Provider for Hyper-V

```
http://aka.ms/downloaddra_cus
```

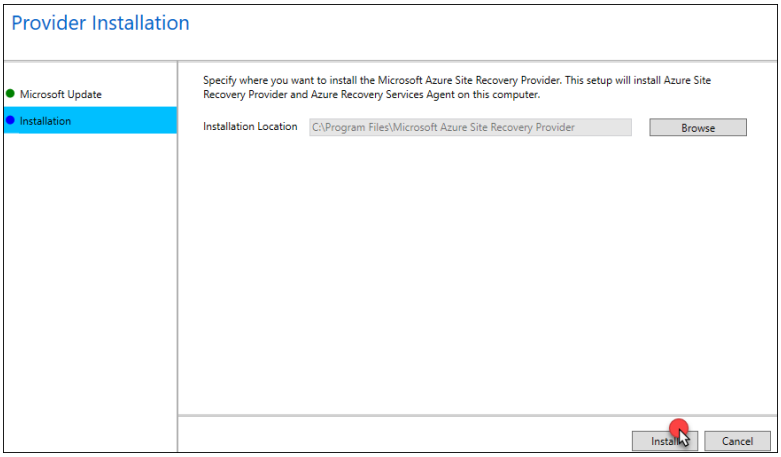
22. Select **Run**



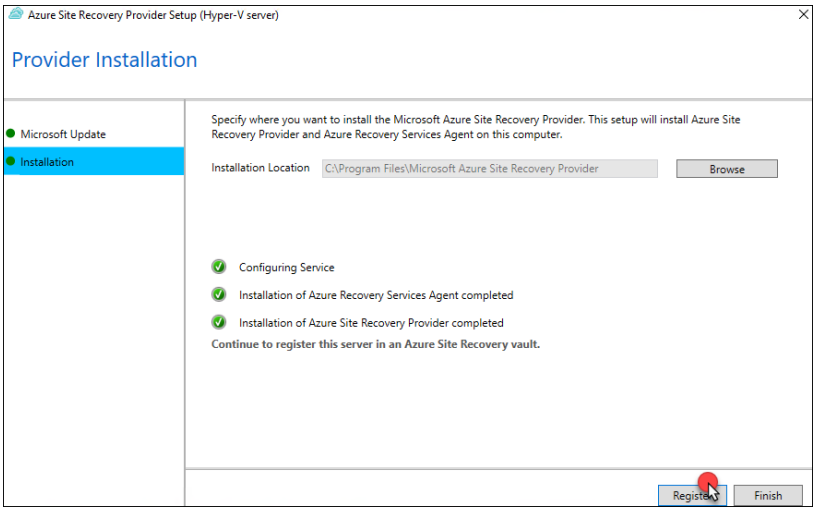
23. Select **On** and then **Next**



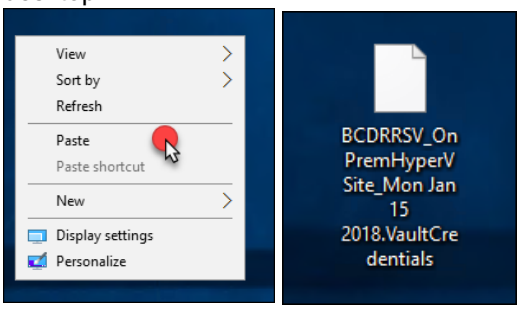
24. Select **Install** on the Provider Installation screen



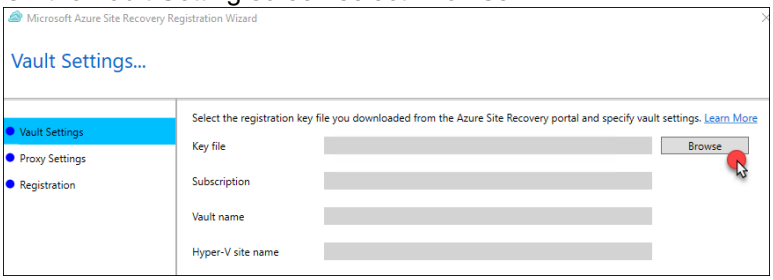
25. Once the Provider has been installed you will come to a screen that will request for you **Register** or **Finish**. Select **Register**.



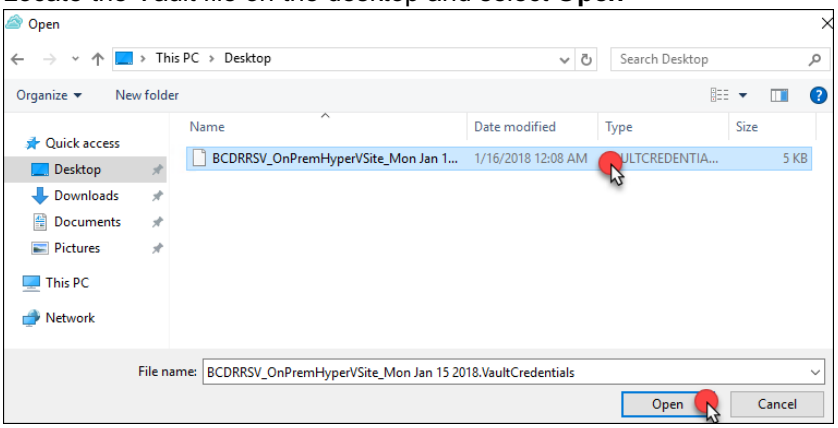
26. Minimize your Remote Desktop window and locate the vault registration key which is in the **Downloads** directory of your **LABVM**. Right-click the file and copy it. Move back to your **HYPERVHOST** and paste the file to the desktop.



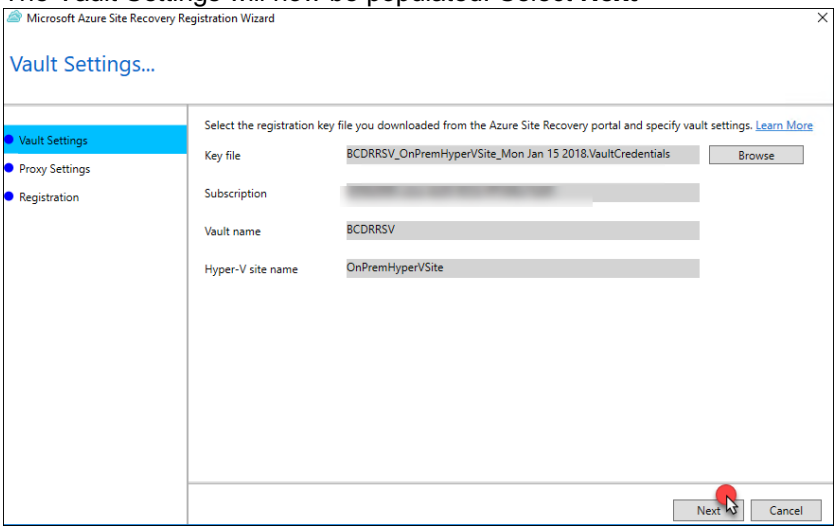
27. On the Vault Setting screen select **Browse**



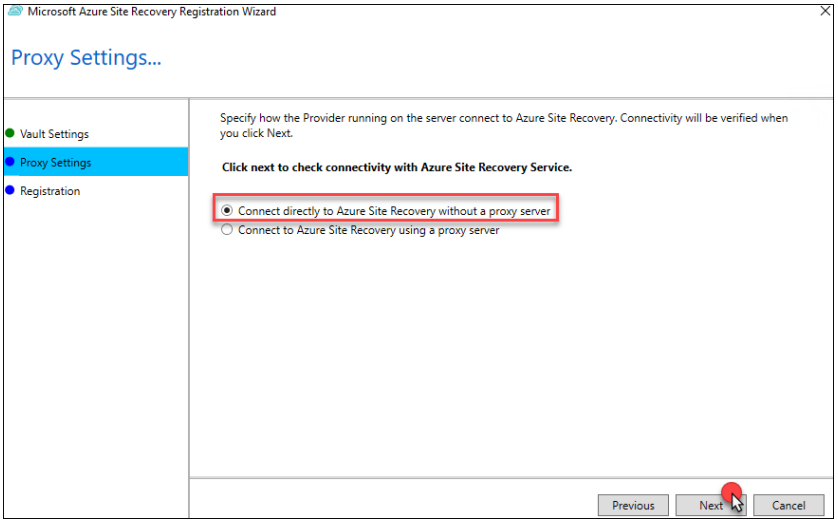
28. Locate the Vault file on the desktop and select **Open**



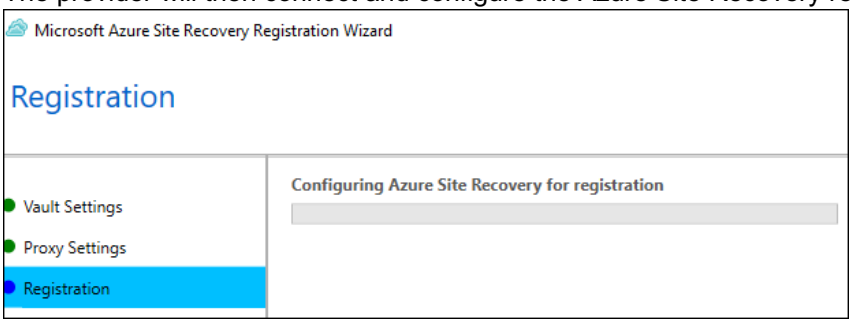
29. The Vault Settings will now be populated. Select **Next**



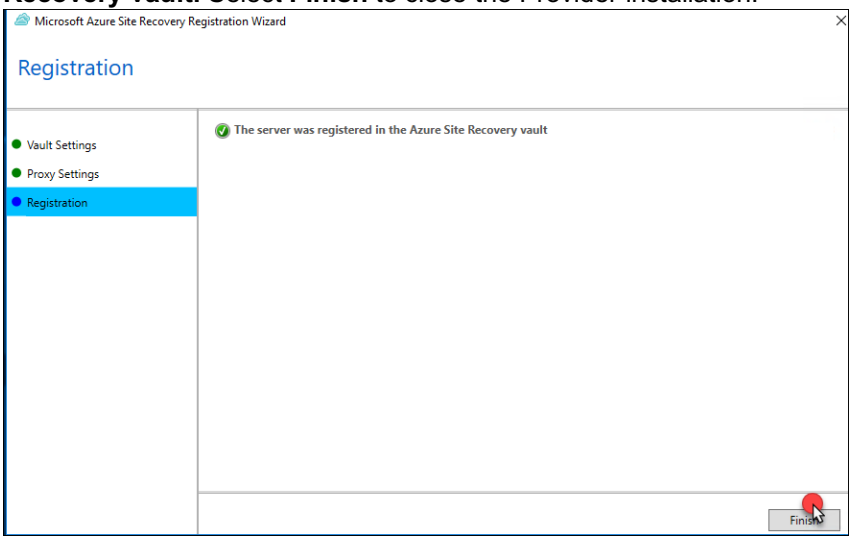
30. On the Proxy settings screen retain the settings **Connect directly to Azure Site Recovery without a proxy server** and select **Next**



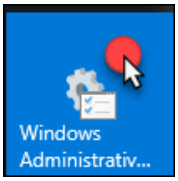
31. The provider will then connect and configure the Azure Site Recovery registration for the Hyper-V Server.



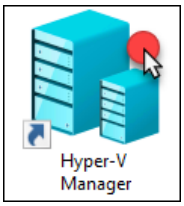
32. After a few minutes, the wizard will be complete with the message **The Server was registered in the Azure Site Recovery vault.** Select **Finish** to close the Provider installation.



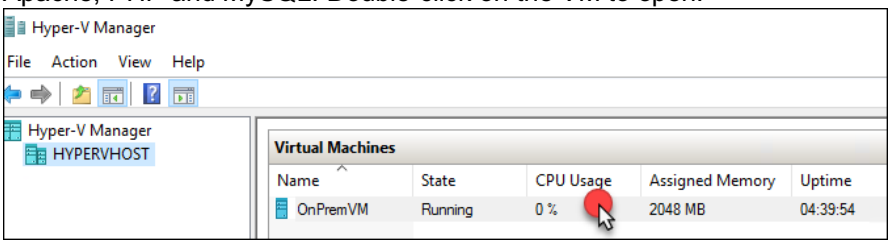
33. Select **Start**, **Windows Administrative Tools**



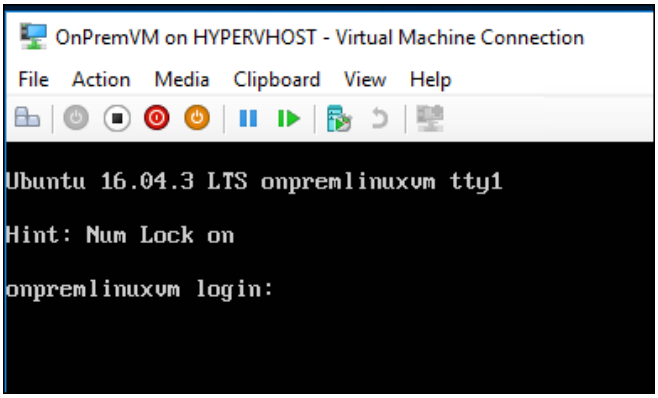
34. Locate then double-click the **Hyper-V Manager**



35. When the Hyper-V Manager opens select the Server name in the left pane and then you will see that the **OnPremVM** virtual machine is running on your **HYPERVHOST**. This is an Ubuntu 16.04.3 LTS server running Apache, PHP and MySQL. Double-click on the VM to open.



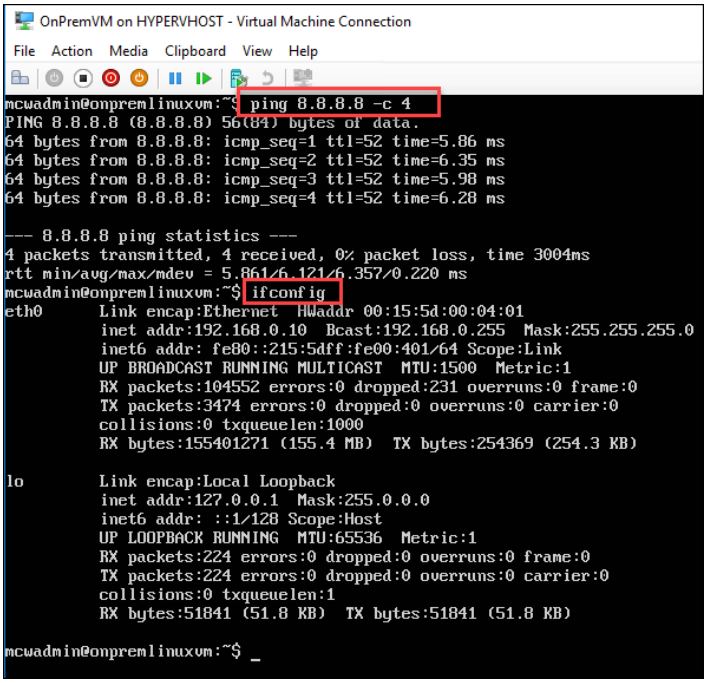
36. The console for the **OnPremVM** will load. Press **Enter** to get a login prompt.



37. Login to the VM using the following credentials:
    a. **User Name**: mcwadmin
    b. **Password**: demo@pass123

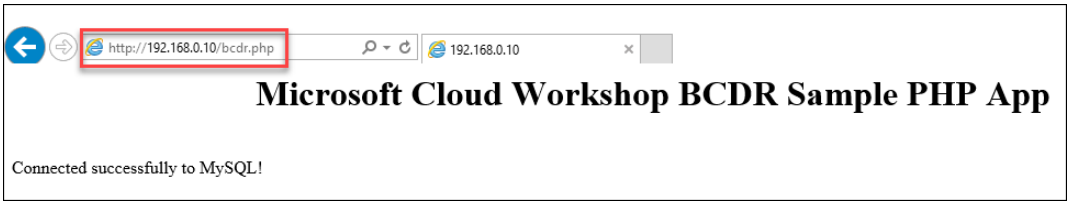
38. Once logged in enter a few commands and notice that you can get to the internet and that the **local IP address of the VM is currently 192.168.0.10**

```
ping 8.8.8.8 -c 4
ifconfig
```



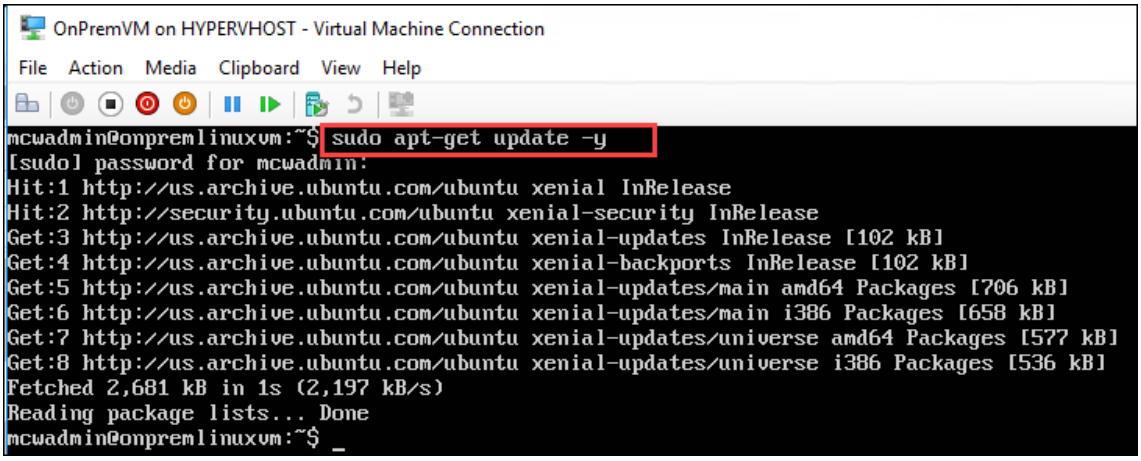
39. Open Internet Explorer on **HYPERVHOST** and browse to the following URL of the **OnPremVM**. This is very simple PHP application running on the **OnPremVM** connected to the MySQL Server that is running locally on the VM. This is to simulate an application is that running on one VM in the on-premises data center.

```
http://192.168.0.10/bcdr.php
```



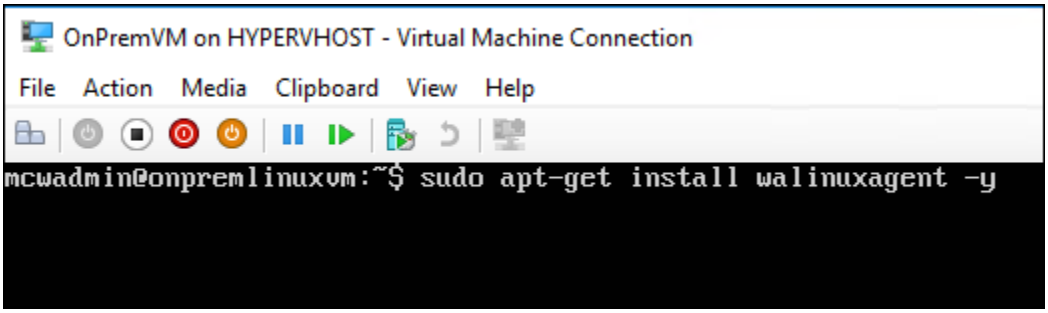
40. From the command prompt of the OnPremVM update the OS with the latest patches by using the following command. You will need to enter the password again.

```
sudo apt-get update -y
```



41. After the updates complete, install the Azure Guest Agent for Linux on the VM using the following commands

```
sudo apt-get install walinuxagent -y
```

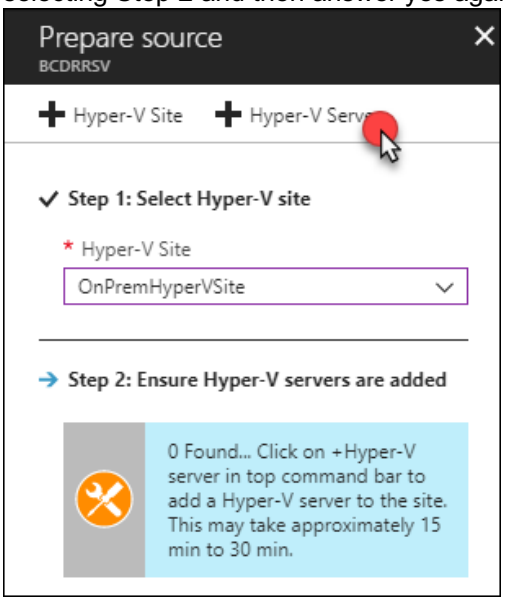


Note: You may see some errors, but this is normal behavior since the VM now thinks it should be in Azure.

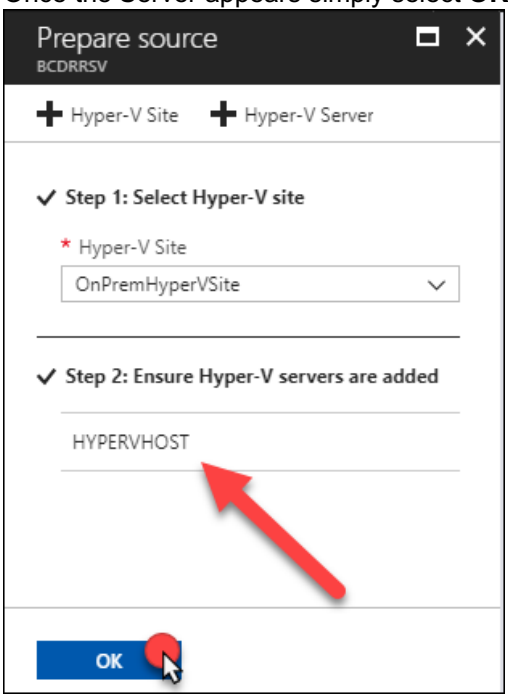
42. Enter exit to log out of the **OnPremVM**

43. Sign out of **HYPERVHOST** and return to the Azure portal running on your **LABVM**

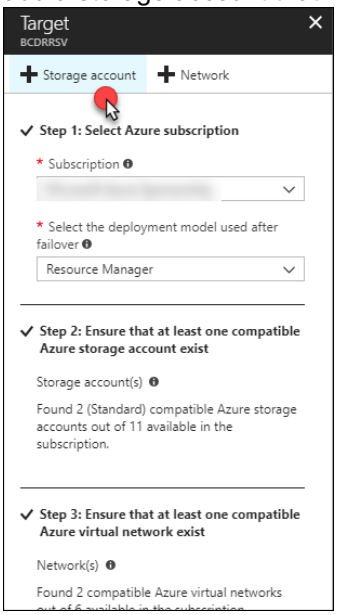
44. You will need to Return to the Prepare Source screen and select **+Hyper-V Server**. Notice the warning that it could take up to 30 mins for this server to appear, but in practice you should cancel out of this window by selecting Step 2 and then answer yes again with **I have done it** to the question of Deployment planning.



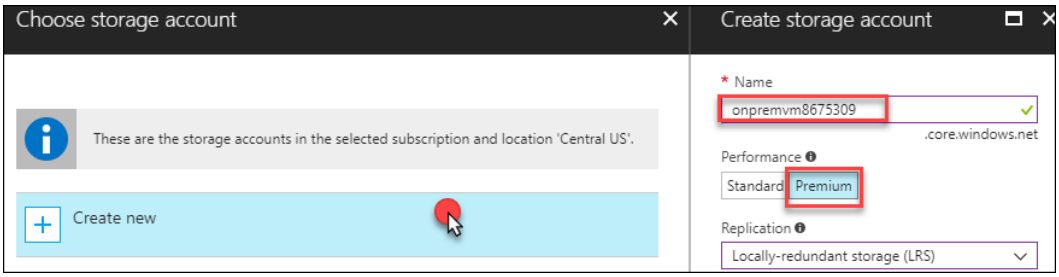
45. Once the Server appears simply select **OK**



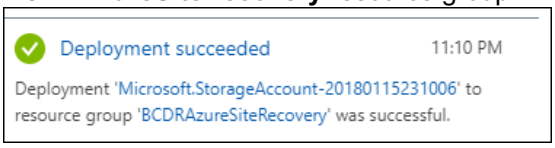
46. On **Step 4 Target Prepare** review the screen to better understand the various steps. Select **+Storage account** to add a storage account that will be used for the **OnPremVM** when it is failed over to Azure.



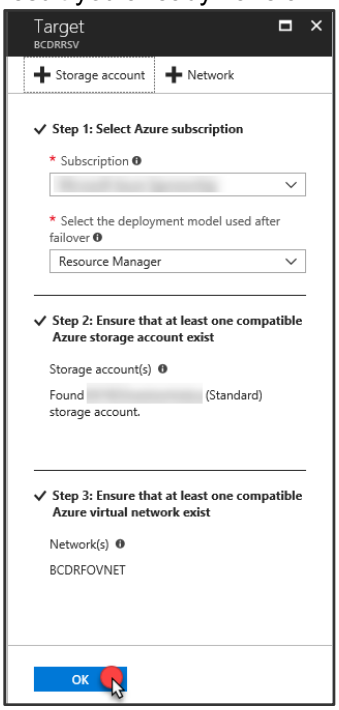
47. Select **Create New** and provide a unique name for your storage account containing the name of the VM **OnPremVM** with added characters to make it unique. Also, select the Premium tier for the storage account and select **OK**.



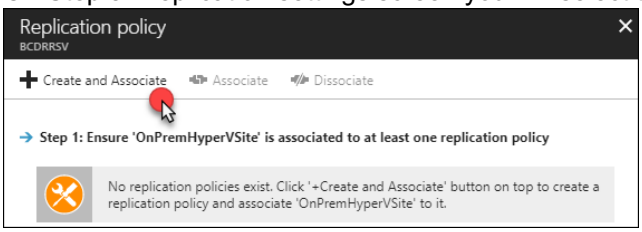
48. The portal will submit a deployment, and you must wait until this completes. It will be created in the **BCDRAzureSiteRecovery** resource group.



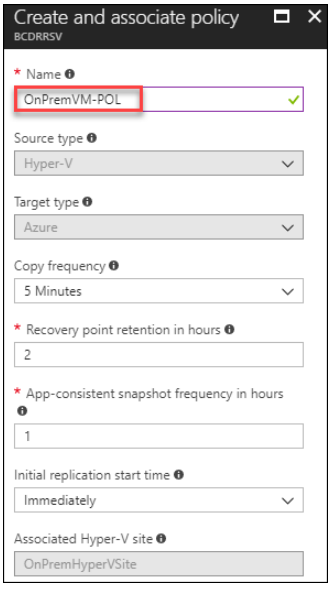
49. You will be failing the **OnPremVM** into the **Secondary** site that was deployed with your IaaS installation, so as a result you already have a Virtual Network created. Select **OK** on the Target blade to continue.



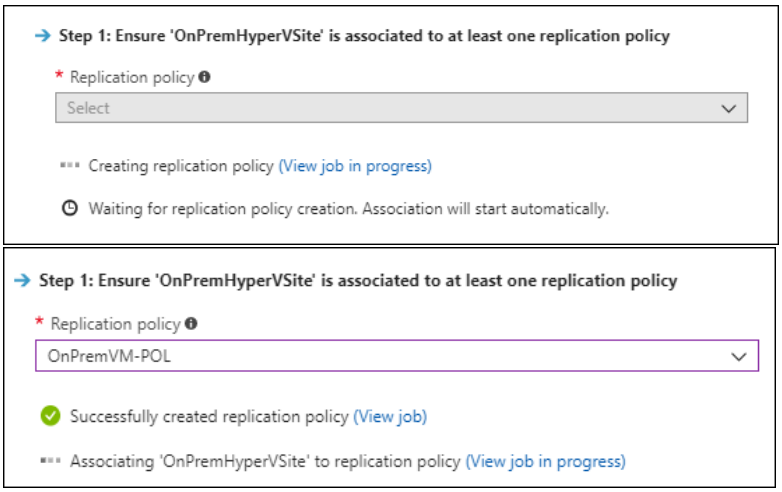
50. On Step 5: Replication settings screen you will select the **+Create and Associate** to a **Replication policy**



51. Enter the Name: **OnPremVM-POL**, review the settings that you can configure and then select **OK**

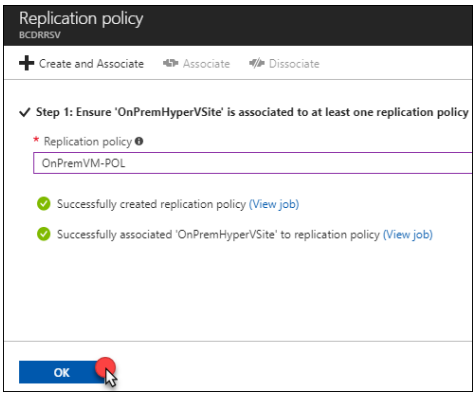


52. Azure will run a deployment and start the process of creating and then associating the Hyper-V Site with the replication policy

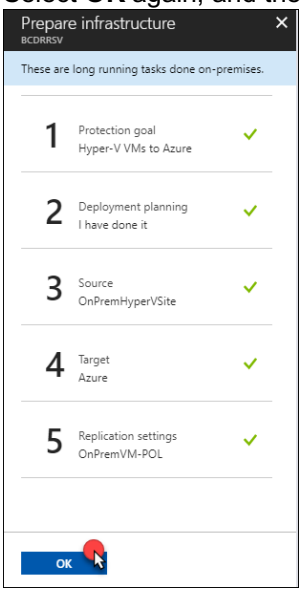


Note: This will take a couple of minutes to complete. Please wait until this completes prior to moving on.

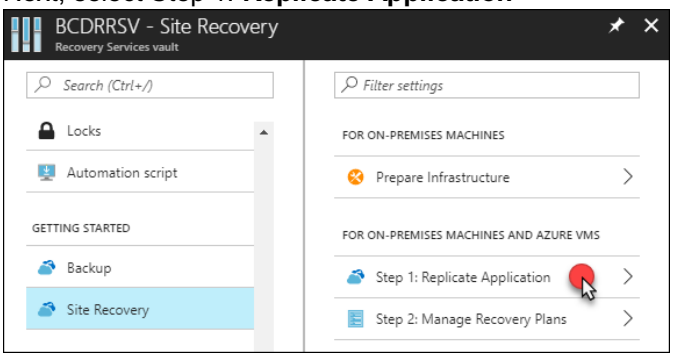
Once complete select **OK**



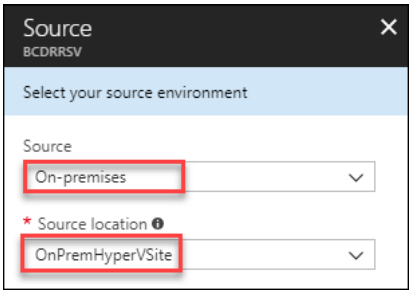
53. Select **OK** again, and the process for adding the Hyper-V Server to the Recovery Services Vault will be complete



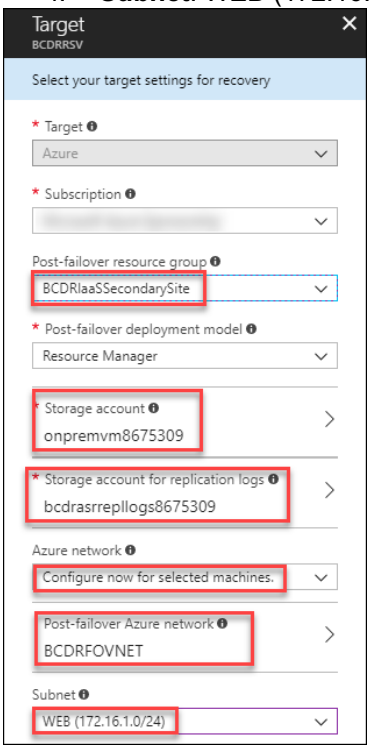
54. Next, select Step 1: **Replicate Application**



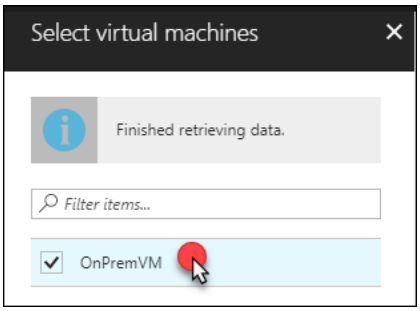
55. On the Source blade select: Source: **On-premises** and Source location: **OnPremHyperVSite** and select **OK**.



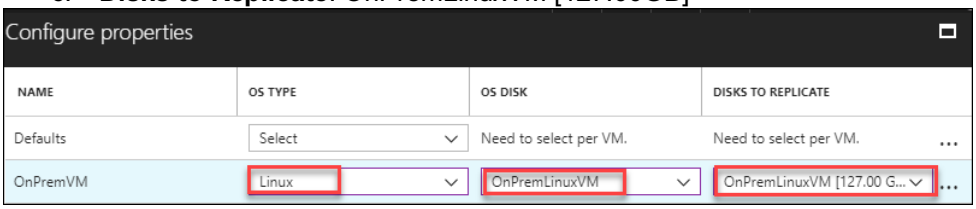
56. Complete the Target blade using the following inputs:
    a. **Post-failover resource group**: BCDRIaaSSecondarySite
    b. **Storage Account**: Select the account you just created (onpremvm8675309)
    c. **Storage Account for replication logs**: create a new one using the prefix bcdrasrrepllogs
    d. **Azure network**: configure now for selected machines
    e. **Post-failover Azure network**: BCDRFOVNET
    f. **Subnet**: WEB (172.16.1.0/24)



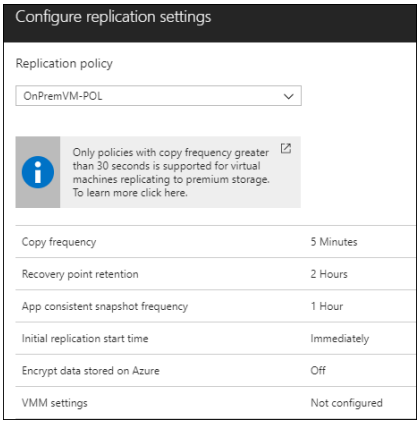
57. Next on the **Select virtual machines** select **OnPremVM** and then select **OK**



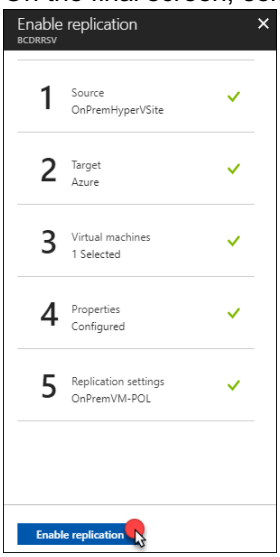
58. Complete the **OnPremVM** selections of the **Configure properties** blade using these inputs and then select **OK**
    a. **OS Type**: Linux
    b. **OS Disk**: OnPremLinuxVM
    c. **Disks to Replicate**: OnPremLinuxVM [127.00GB]



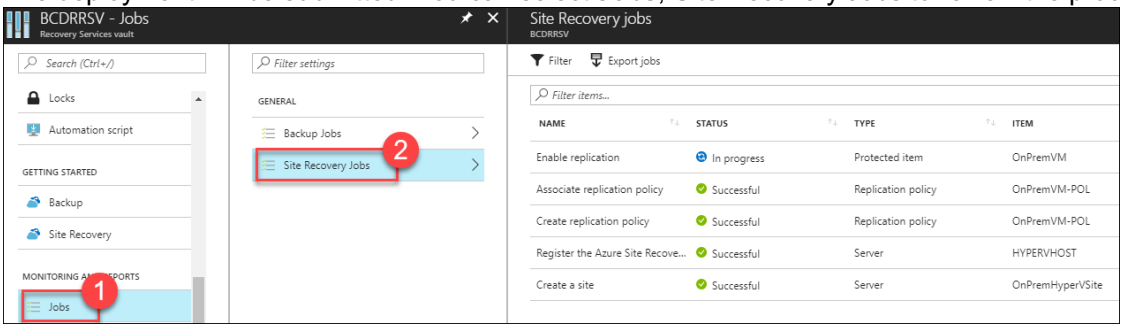
59. On the **Configure replication settings** blade review the selections and notice that the **OnPremVM-POL** that you created has been selected. Select **OK**.



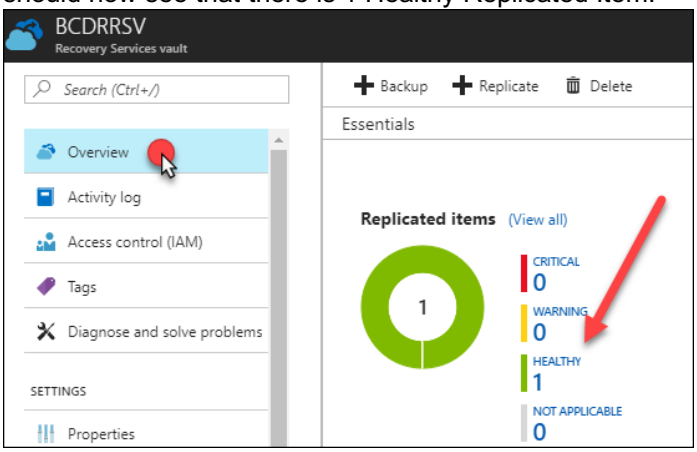
60. On the final screen, select **Enable replication**



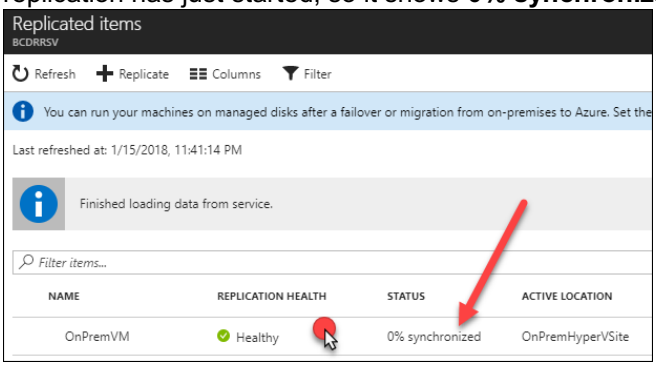
61. The deployment will be submitted. You can select **Jobs**, Site Recovery Jobs to review the process



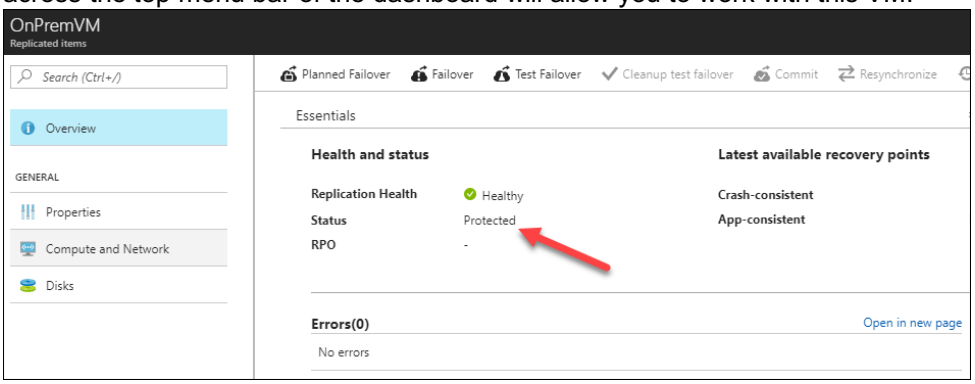
62. After a few minutes, the Enable replication will move to Successful. Select **Overview** on the **BCDRRSV,** and you should now see that there is 1 Healthy Replicated Item.



63. Select **Healthy 1** and you will see the replicated items list. Notice it shows that the VM is healthy, but the replication has just started, so it shows **0% synchronized**. It will take a few minutes for the VM to replicate.

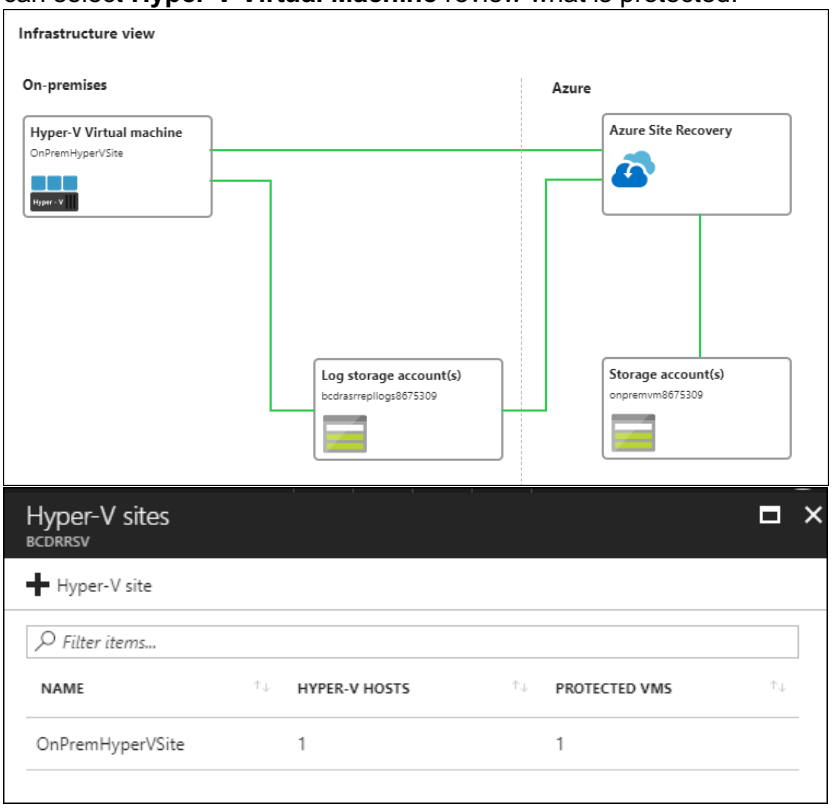


64. Select **OnPremVM**. Review the Replication details for **OnPremVM**. Once the VM has replicated the selections across the top menu bar of the dashboard will allow you to work with this VM.

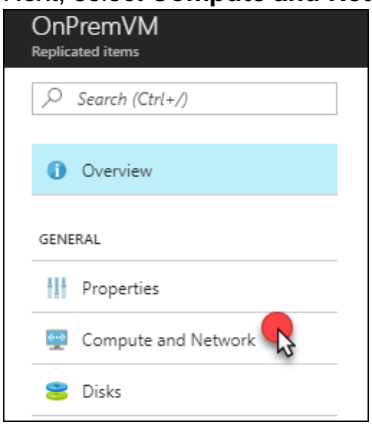


Note: It will take about 15 minutes for the VM to replicate, so you can move on and come back to review the environment later.

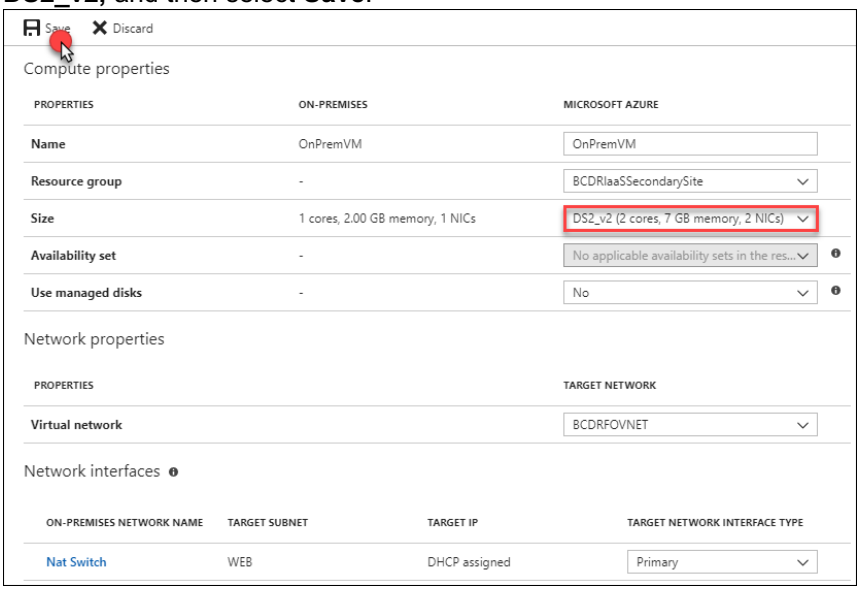
65. Scroll down and review the Infrastructure view of the BCDR environment you have created for **OnPremVM**. You can select **Hyper-V Virtual Machine** review what is protected.



66. Next, select **Compute and Network**



67. Details of the VM will be shown, and you can make configuration changes. Change the **Size** of the VM to **DS2_v2,** and then select **Save**.

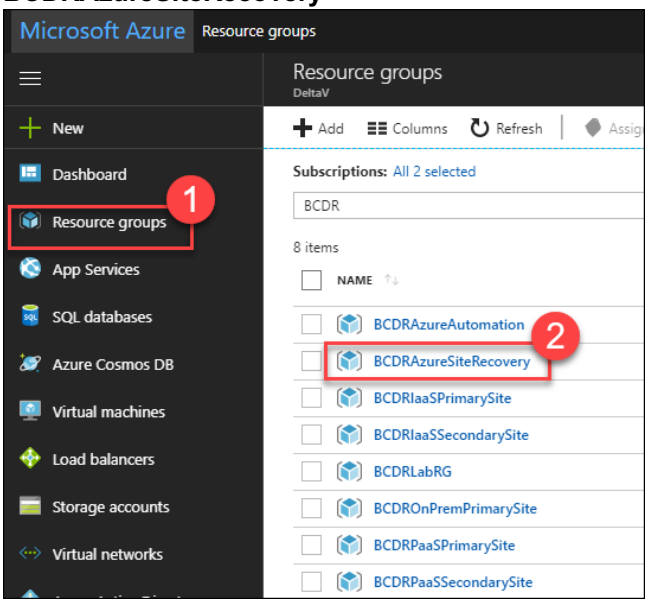


Note: It could take a few minutes for these screens to populate, so be patient. You can come back to this step later to adjust the size if you wish.

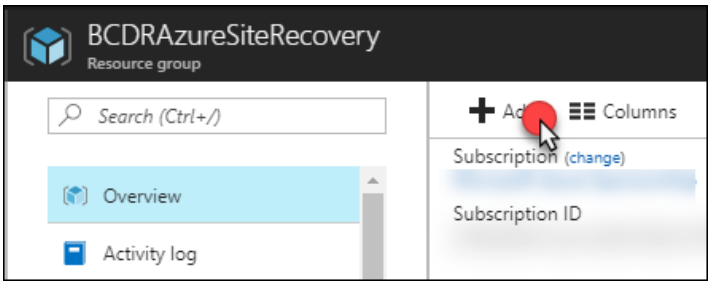
Task 2: Configure IaaS SQL Always On availability groups for region to region failover

In this task, you will build a Windows Failover Cluster and configure SQL Always On Availability Groups. This will be in place to ensure that if there is an issue in the **Primary** site in Azure you can failover to the **Secondary** site and have access to the data for the application. You will also configure the Traffic Manager to ensure that the Web Application will always answer to the same DNS name even when it is failed over to the **Secondary** site.

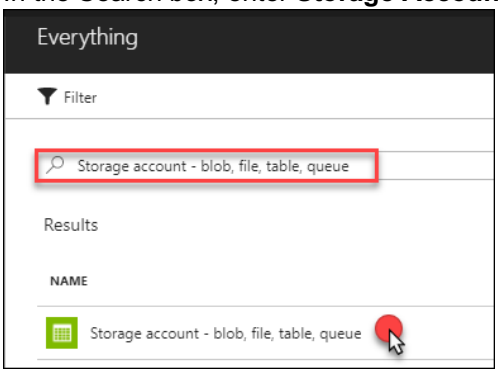
1. From the **LABVM** navigate to the Azure Portal, and navigate to **Resource Groups** and then **BCDRAzureSiteRecovery**



2. Select **+Add**

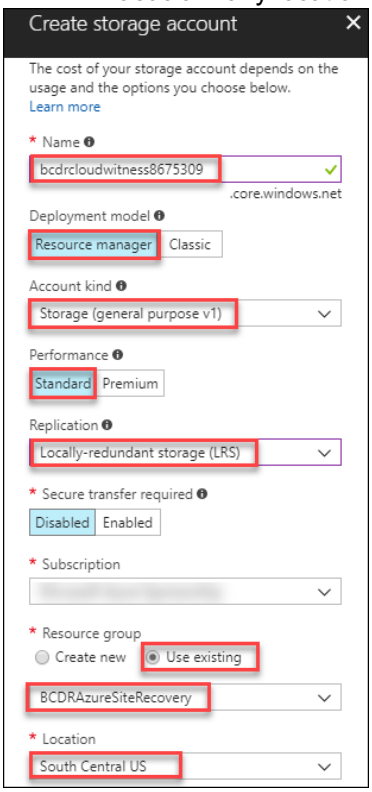


3. In the Search box, enter **Storage Account,** and then select **Storage account – blob, file, table, queue**

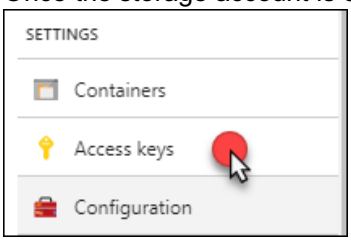


4. Select **Create**

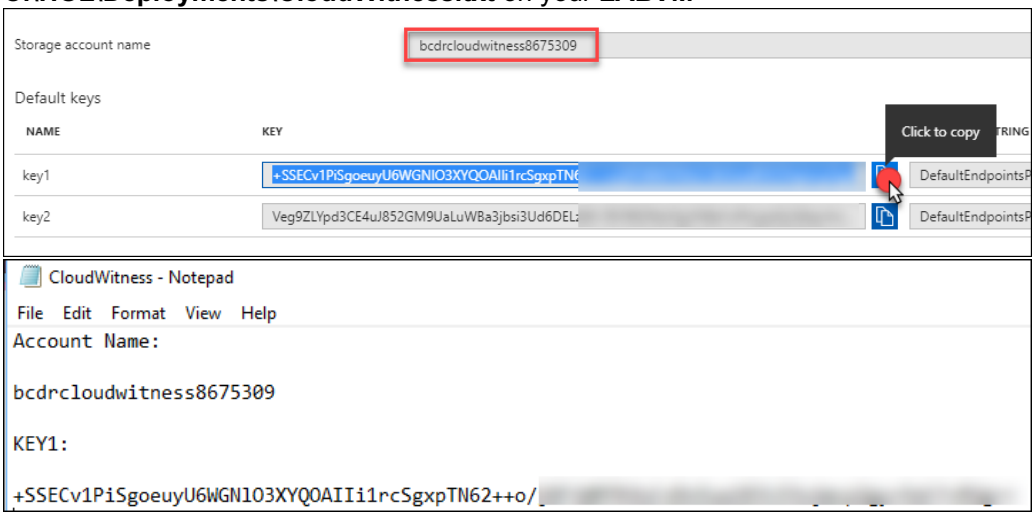
5. Complete the **Create storage account** blade using the following details, then select **Create**:
    a. **Name:** Unique name starting with bcdrcloudwitnessxxx
    b. **Deployment model**: Resource manager
    c. **Account kind**: Storage (general purpose v1)
    d. **Performance**: Standard
    e. **Replication**: Locally-redundant storage (LRS)
    f. **Access tier**: Hot
    g. **Resource group**: Use existing / BCDRAzureSiteRecovery
    h. **Location**: any location in your area that is NOT your Primary or Secondary Site



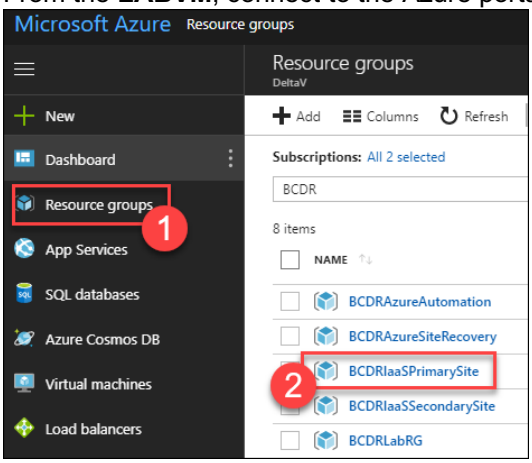
6. Once the storage account is created, locate and select **Access keys** under **Settings**



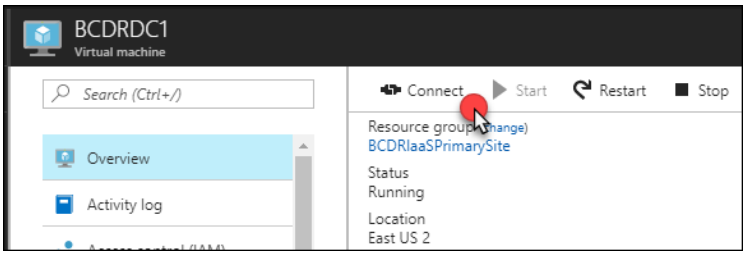
7. Copy the name of the account and the first access key to notepad and save the file as **C:\HOL\Deployments\CloudWitness.txt** on your **LABVM**



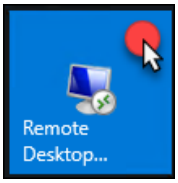
8. From the **LABVM**, connect to the Azure portal and locate the **BCDRIaaSPrimarySite**



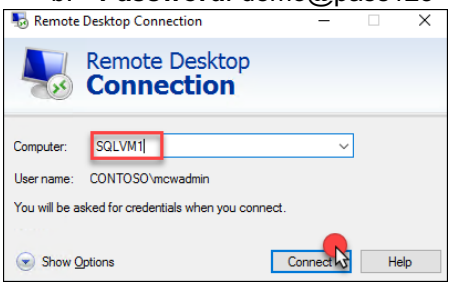
9. Select **BCDRDC1** and then select **Connect**



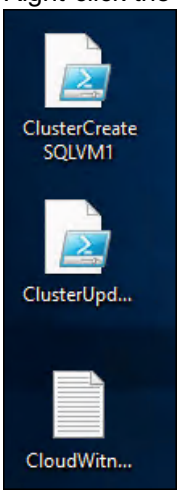
10. Once in the RDP session to **BCDRDC1**, select **Start** and then **Remote Desktop**. You will need to connect to **SQLVM1** for the next steps



11. Connect to **SQLVM1** using the following credentials:
    a. **User Name**: CONTOSO\mcwadmin
    b. **Password**: demo@pass123

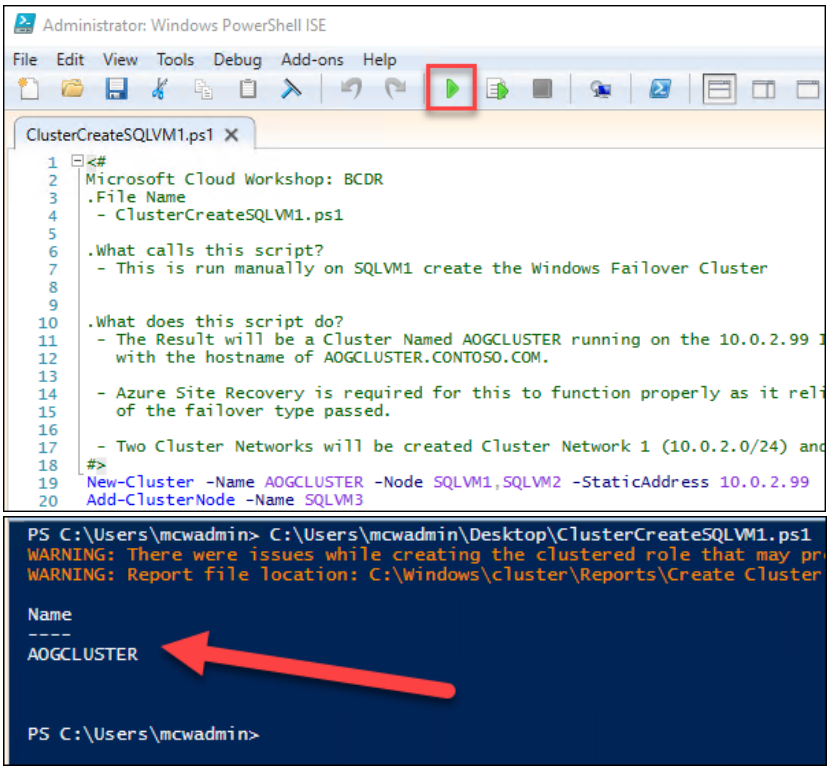


12. Minimize your Remote Desktop window and locate three files: **ClusterCreateSQLVM1.ps1**, **ClusterUpdateSQLVM1.ps1** and **CloudWitness.txt** in the **C:\HOL\Deployments** directory of your **LABVM**. Right-click the files and copy them, then move back to your **SQLVM1** and **paste the files to the desktop**.



13. On **SQLVM1**, select Start and then select **PowerShell ISE**

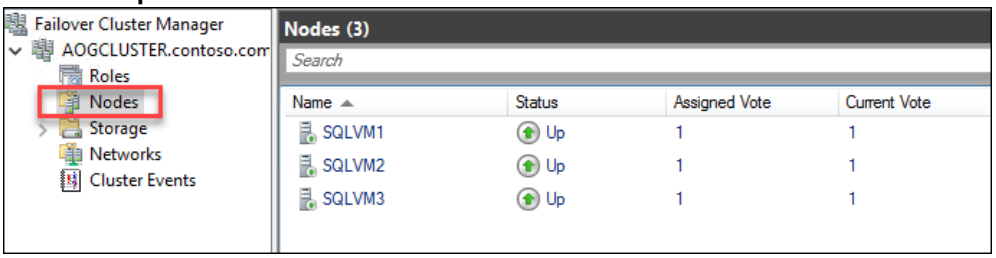
14. Open the **ClusterCreateSQLVM1.ps1** in the PowerShell ISE window. Then select the **green play button**. This script will create the Windows Failover Cluster and add all the SQL VMs as nodes in the cluster. It will also assign a static IP address of 10.0.2.99 to the new Cluster named **AOGCLUSTER**.



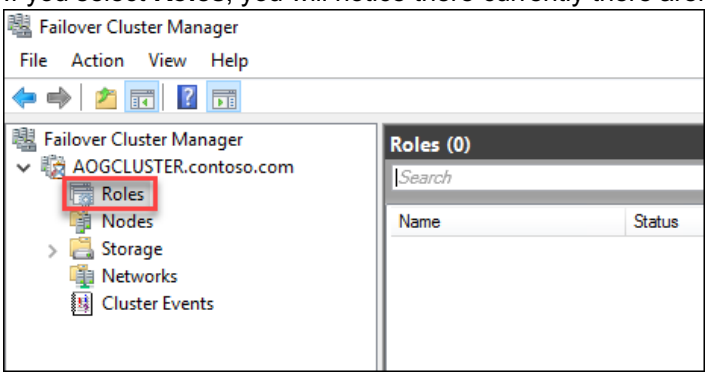
Note: It is possible to use a wizard for this task, but the resulting cluster will require additional configuration to set the static IP address in Azure.

15. After the cluster has been created, select **Start** and then **Windows Administrative Tools**. Locate and open the **Failover Cluster Manager**.

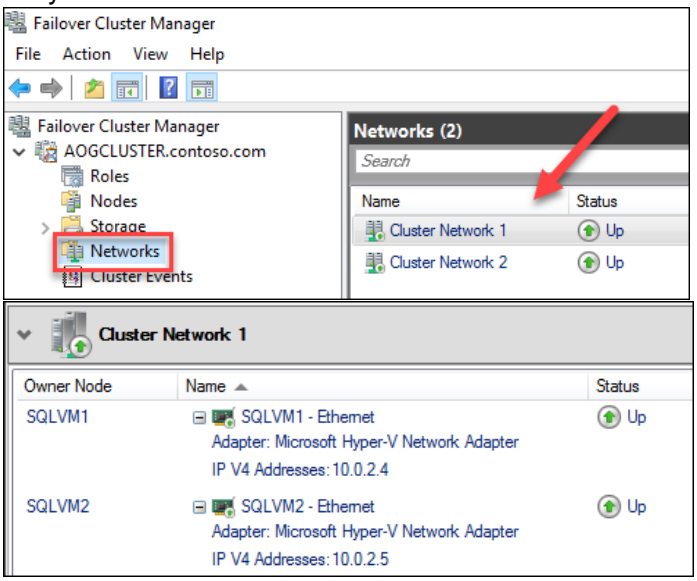
16. When the cluster opens, select **Nodes** and the SQL Server VMs will show as nodes of the cluster and show their status as **Up**



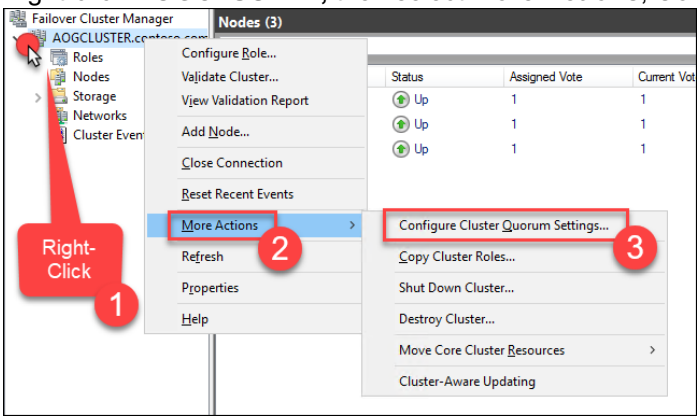
17. If you select **Roles**, you will notice there currently there aren't any roles assigned to the cluster



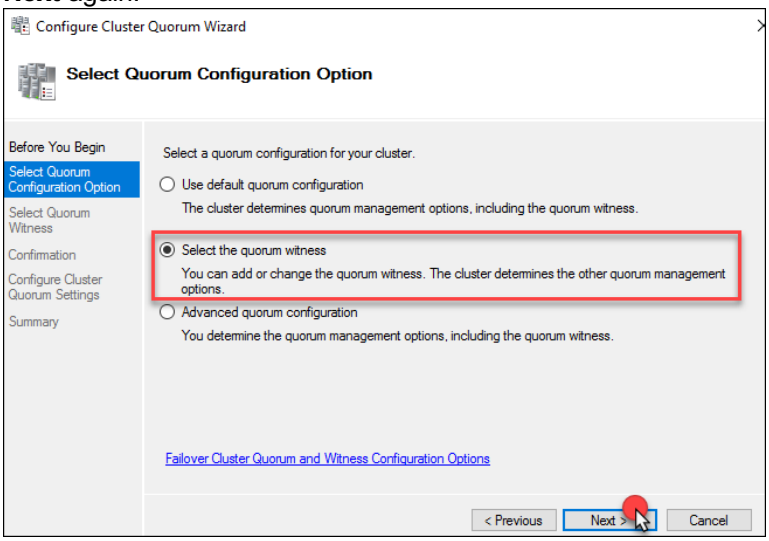
18. Select Networks, and you will see two networks: **Cluster Network 1** and **Cluster Network 2**. Both should show the status of **Up**. If you navigate through to the networks, you will see their IP address spaces, and on the lower tab you can select **Network Connections** and review the nodes.



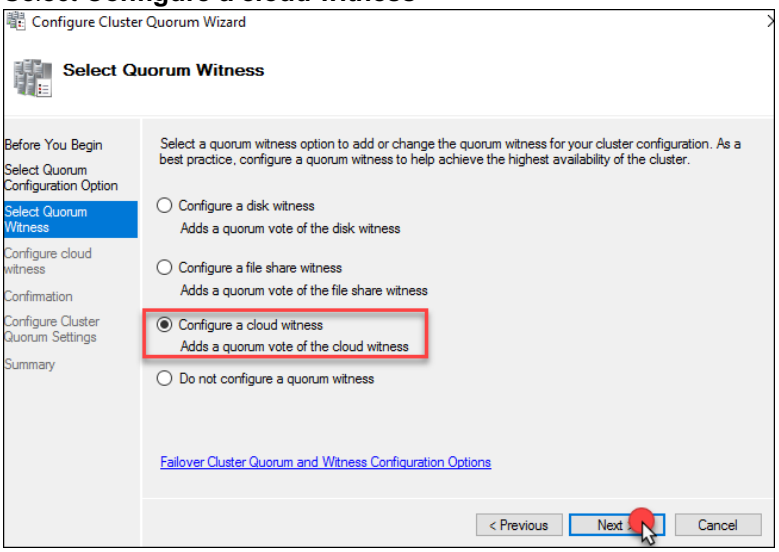
19. Right-click **AOGCLUSTER,** then select **More Actions**, **Configure Cluster Quorum Settings**



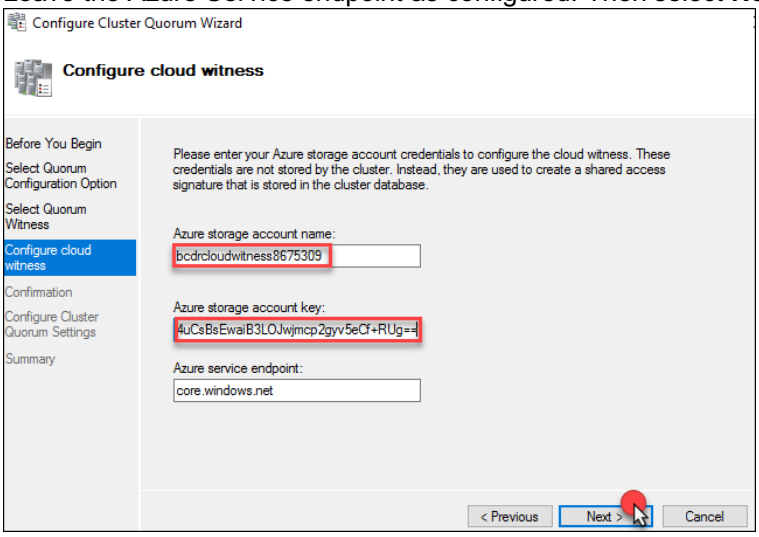
20. On the **Configure Cluster Quorum Wizard** select **Next**, then select **Select the quorum witness**. Then select **Next** again.



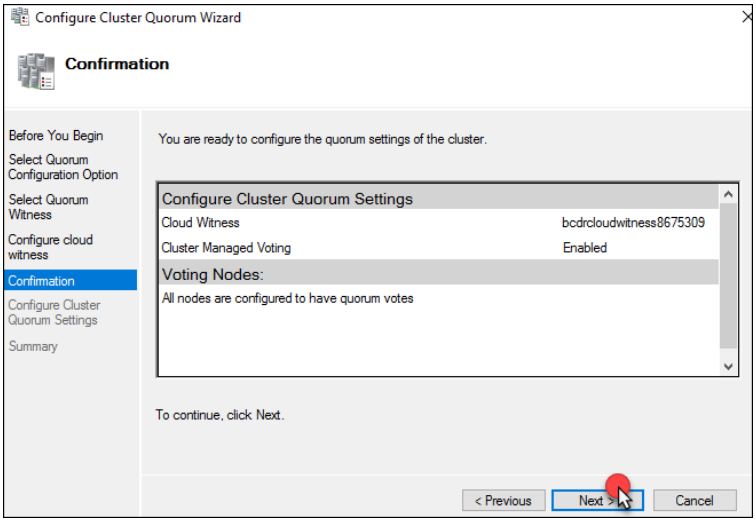
21. Select **Configure a cloud witness**



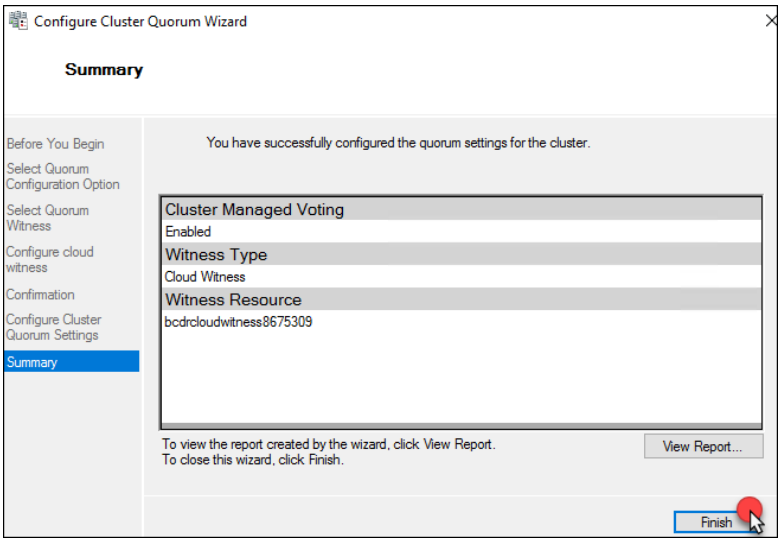
22. Open the **CloudWitness.txt** file on the desktop of **SQLVM1** and copy the **Storage account name** and **KEY1**. Leave the Azure Service endpoint as configured. Then select **Next**.



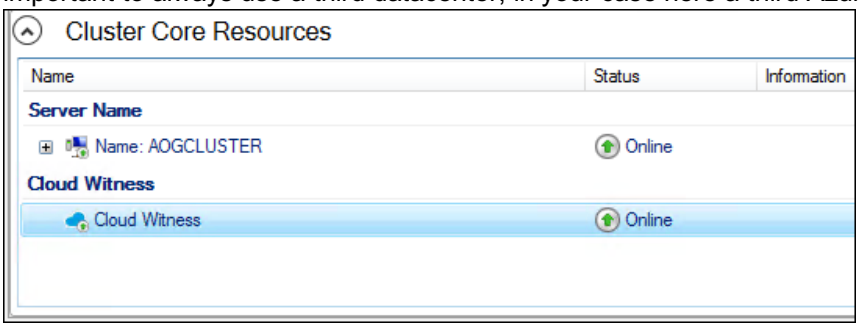
23. Select **Next** on the Confirmation screen



24. Select **Finish**

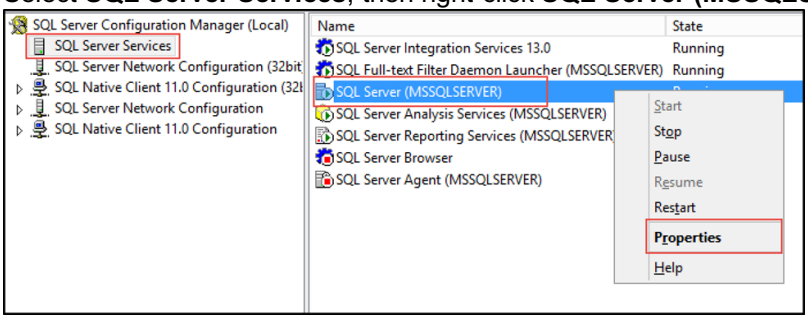


25. Select the name of the Cluster again and the **Cloud Witness** should now appear in the **Cluster Resources**. It's important to always use a third datacenter, in your case here a third Azure Region to be your Cloud Witness.

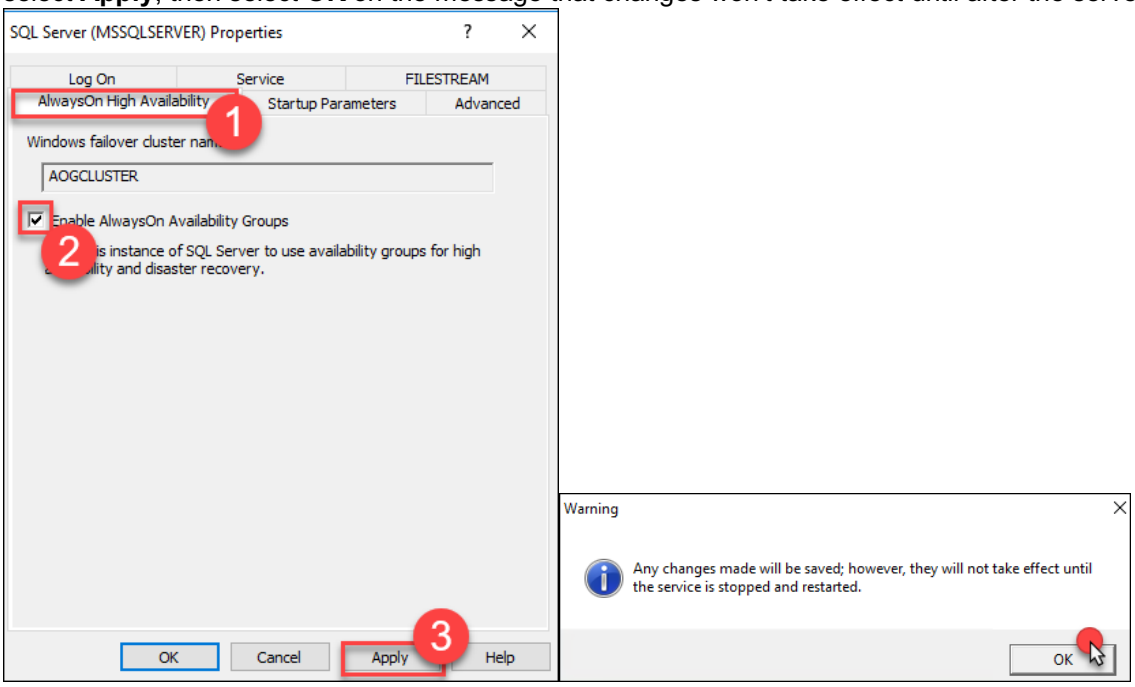


26. Select **Start** and Launch **SQL Server 2017 Configuration Manager**

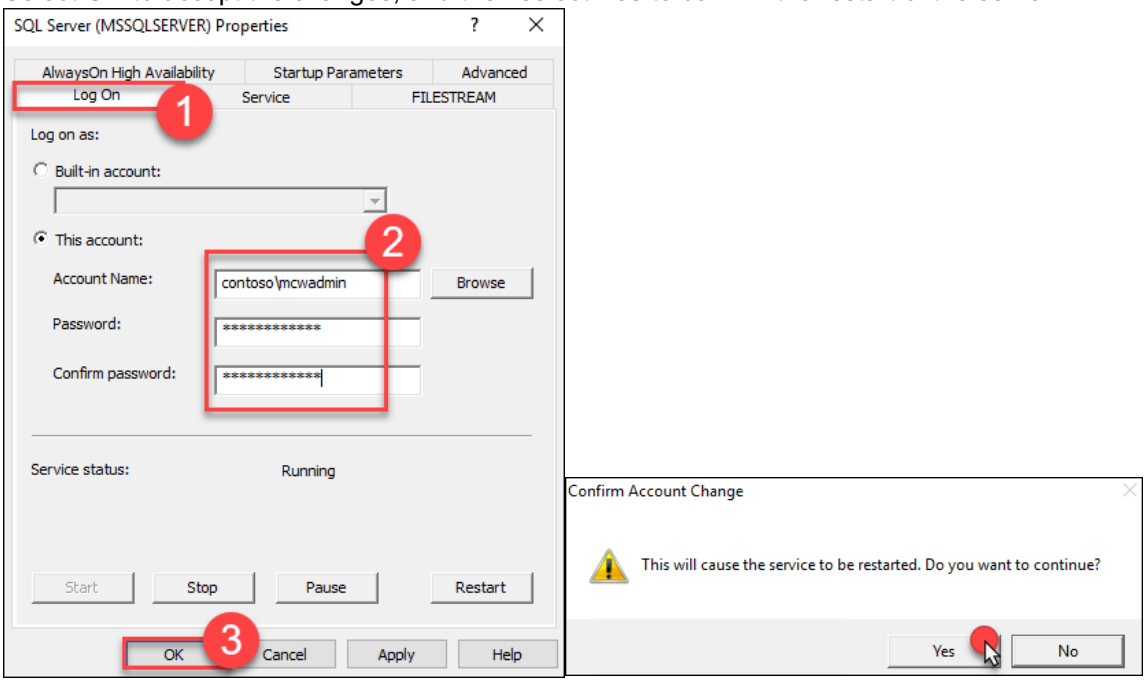
27. Select **SQL Server Services**, then right-click **SQL Server (MSSQLSERVER)** and select **Properties**



28. Select the **AlwaysOn High Availability** tab and check the box for **Enable AlwaysOn Availability Groups**, then select **Apply**, then select **OK** on the message that changes won't take effect until after the server is restarted



29. On the **Log On** tab. Change the service account to **contoso\mcwadmin**, with the password **demo@pass123**. Select **OK** to accept the changes, and then select **Yes** to confirm the restart of the server.



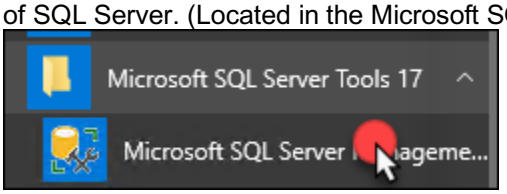
30. Open a new Remote desktop session (this can be done from within SQLVM1), and repeat these steps to **Enable SQL Always On** and change the User name to **contoso\mcwadmin** on each of the other nodes **SQLVM2**, and **SQLVM3.** Make sure that you have restarted the SQL Service on each node prior to moving to the next node.

Note: If you get confused what server you are on open a command prompt and simply enter the command *hostname.*

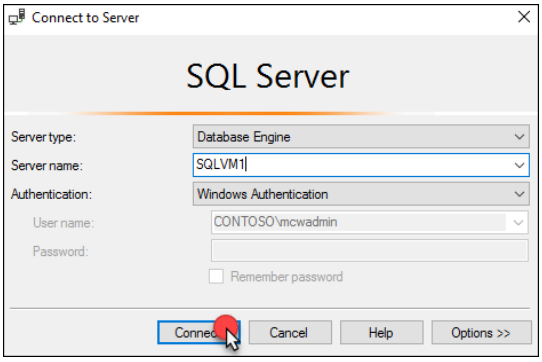
31. After you have completed the process on each SQLVM Node, reconnect to **SQLVM1** using Remote Desktop

Note: Remember that you must use the BCDRDC1 VM as your jumpbox to get into the environment. You can use the Azure portal to connect to BCDRDC1 and then use Remote desktop form there to SQLVM1.

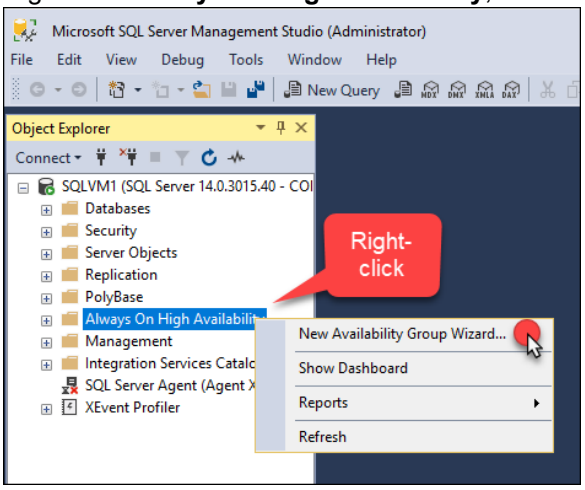
32. Use the Start menu to launch **Microsoft SQL Server Management Studio 17** and connect to the local instance of SQL Server. (Located in the Microsoft SQL Server Tools 17 folder)



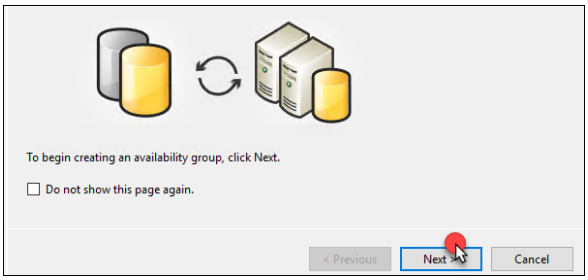
33. Select **Connect** to Sign On to **SQLVM1**



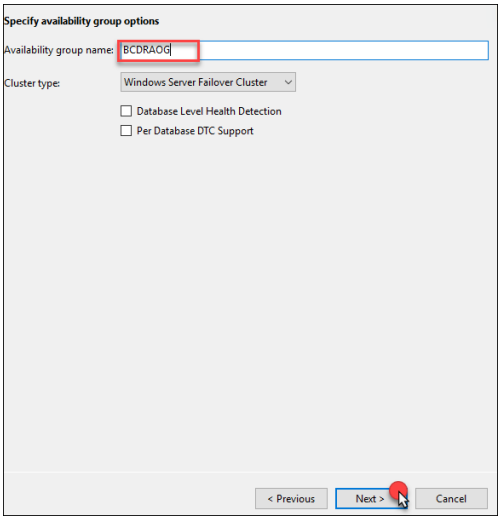
34. Right-click **Always On High Availability**, then select **New Availability Group Wizard**



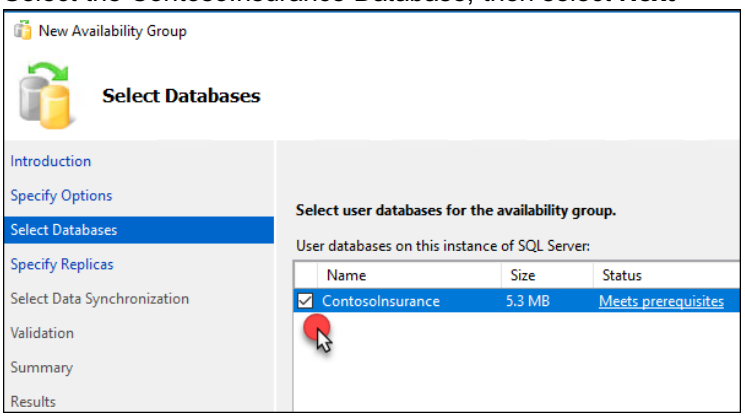
35. Select **Next** on the Wizard



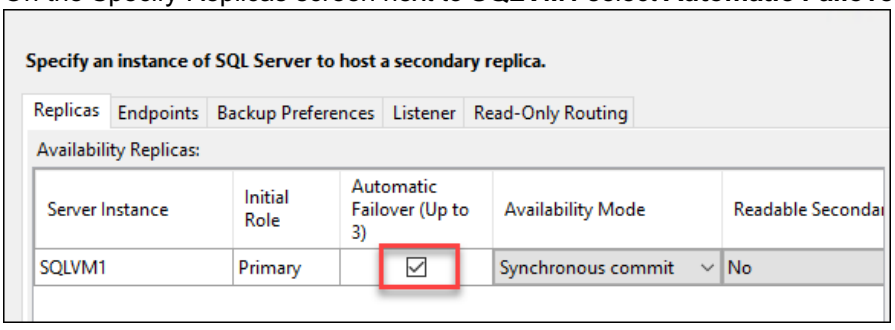
36. Provide the name **BCDRAOG** for the **Availability group name**, then select **Next**



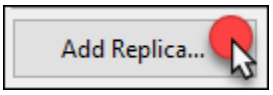
37. Select the ContosoInsurance Database, then select **Next**



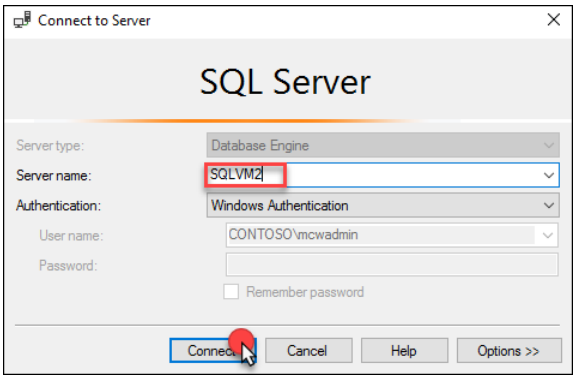
38. On the Specify Replicas screen next to **SQLVM1** select **Automatic Failover**



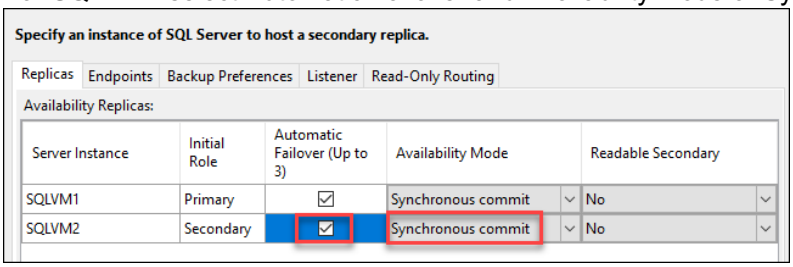
39. Select **Add Replica**



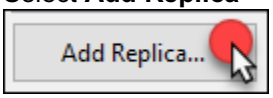
40. On the Connect to Server enter the Server Name of **SQLVM2** and select **Connect**



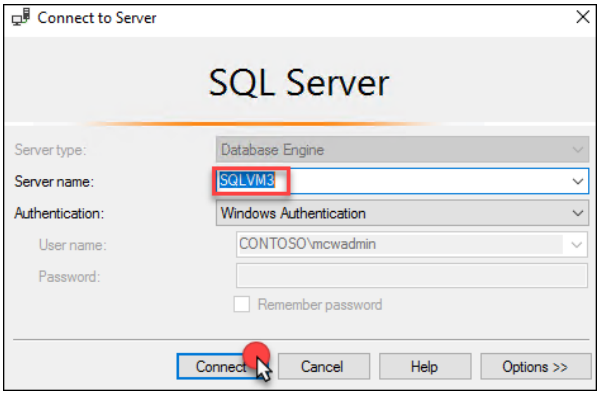
41. For **SQLVM2** select Automatic Failover and Availability Mode of Synchronous commit



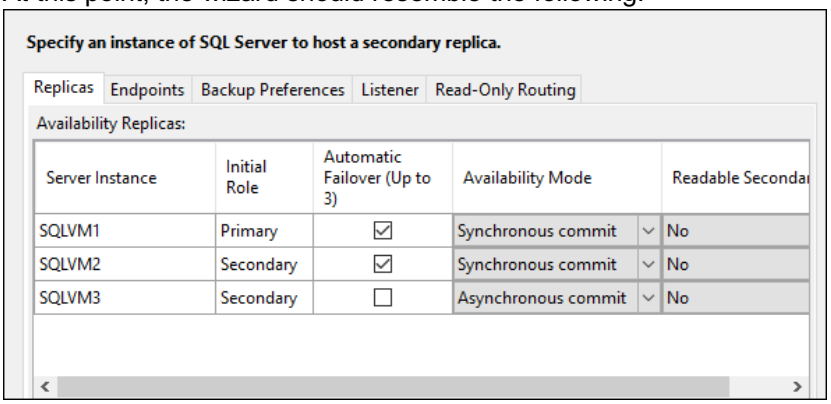
42. Select **Add Replica**



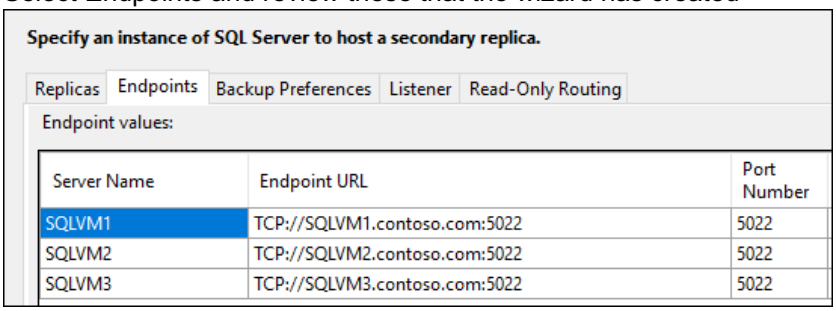
43. On the Connect to Server enter the Server Name of **SQLVM3** and select **Connect**



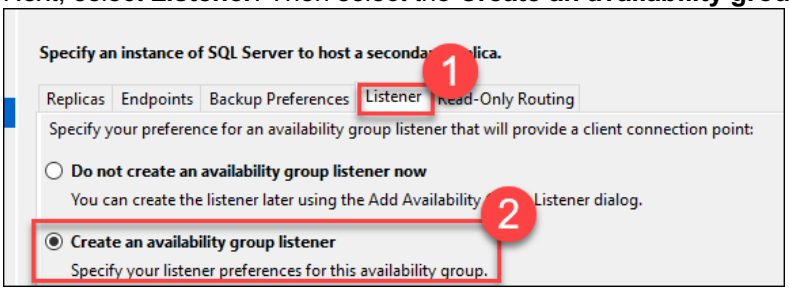
44. At this point, the wizard should resemble the following:



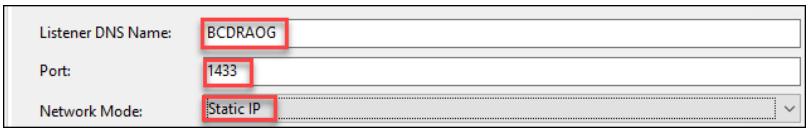
45. Select Endpoints and review these that the wizard has created



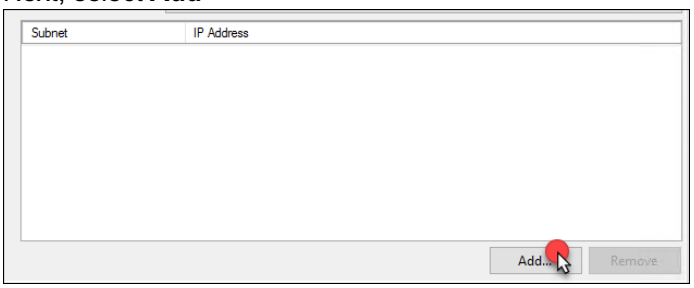
46. Next, select **Listener**. Then select the **Create an availability group listener**



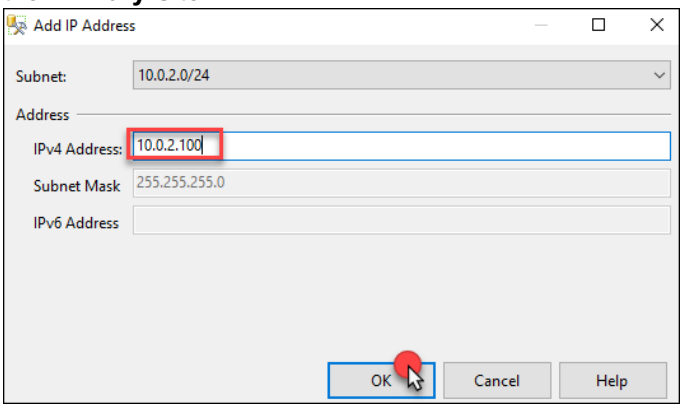
47. Add the following details:
    c. **Listener DNS Name**: BCDRAOG
    d. **Port**: 1433
    e. **Network Mode**: Static IP



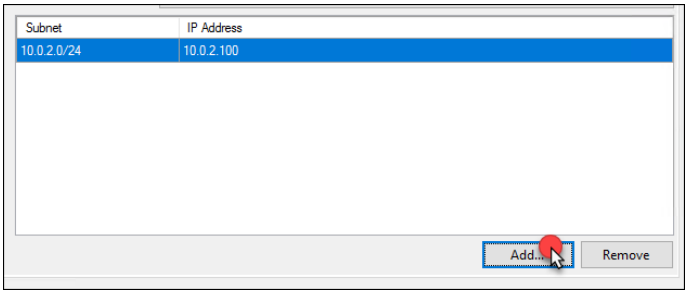
48. Next, select **Add**



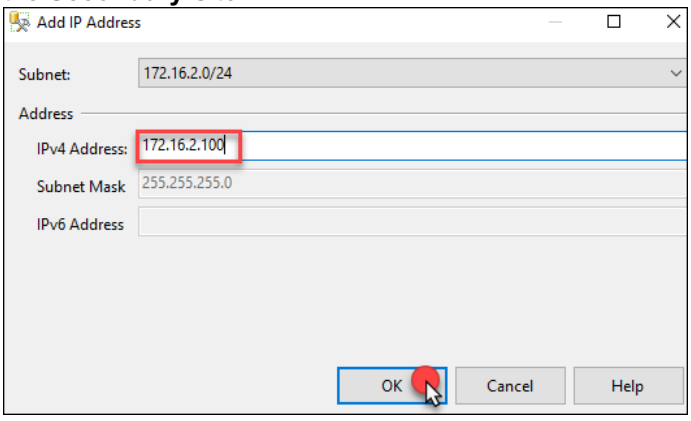
49. Select the Subnet of **10.0.2.0/24** and then add IPv4 **10.0.2.100** and select **OK**. This is the IP address of the Internal Load Balancer that is in front of the **SQLVM1** and **SQLVM2** in the **BCDRVNET Data** Subnet running in the **Primary** Site.



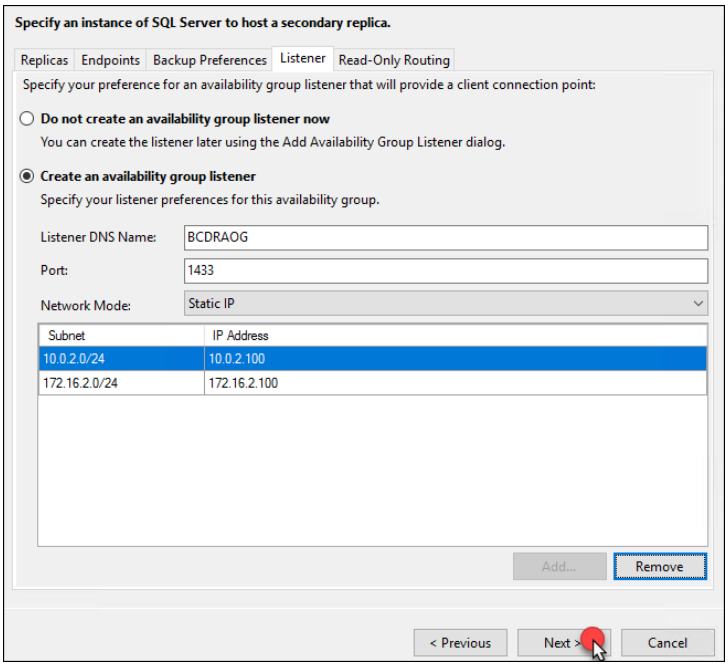
50. Select **Add**



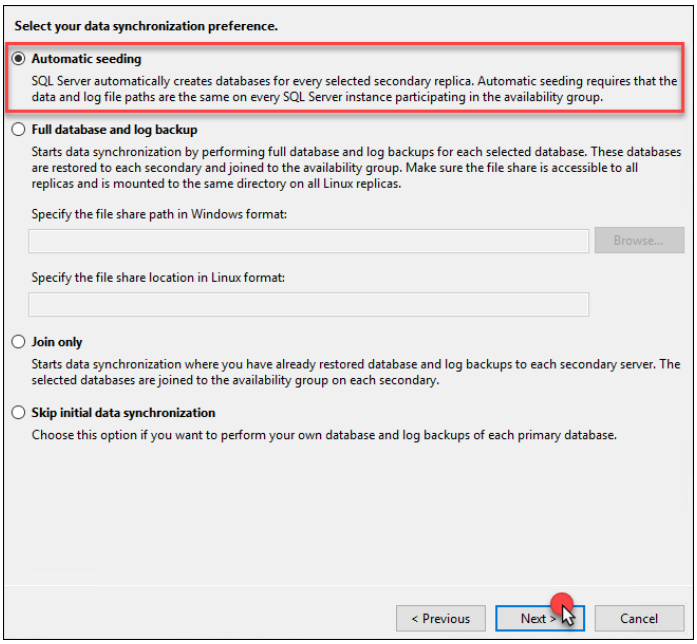
51. Select the Subnet of **172.16.2.0/24** and then add IPv4 **172.16.2.100** and select **OK**. This is the IP address of the Internal Load Balancer that is in front of the **SQLVM3** and **SQLVM4** in the **BCDRFOVNET Data** Subnet running in the **Secondary** Site.



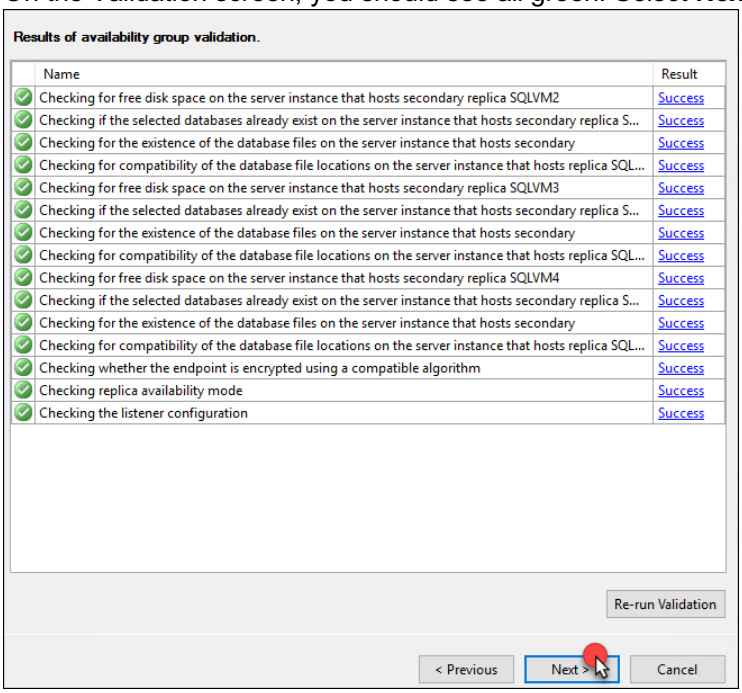
52. Select **Next**



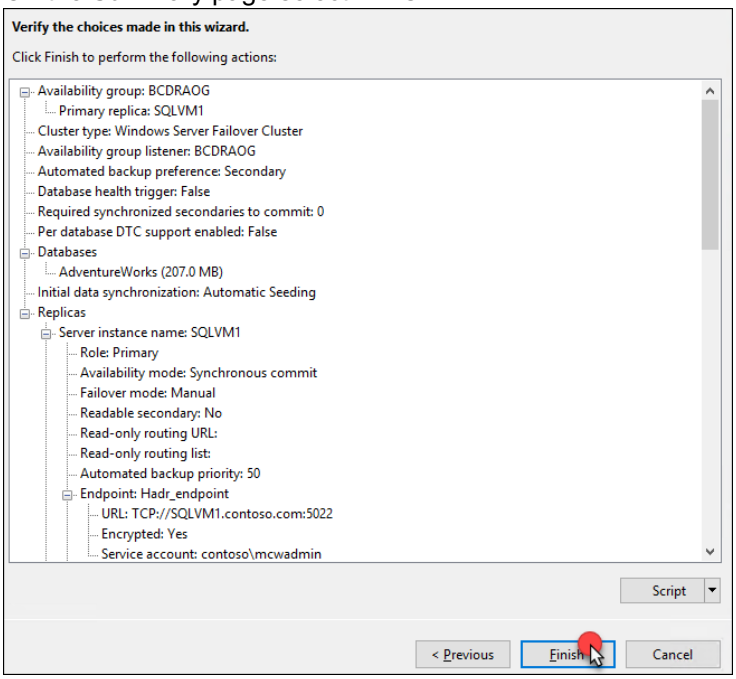
53. On the Select Initial Data Synchronization screen, make sure that **Automatic seeding** is selected and select **Next**



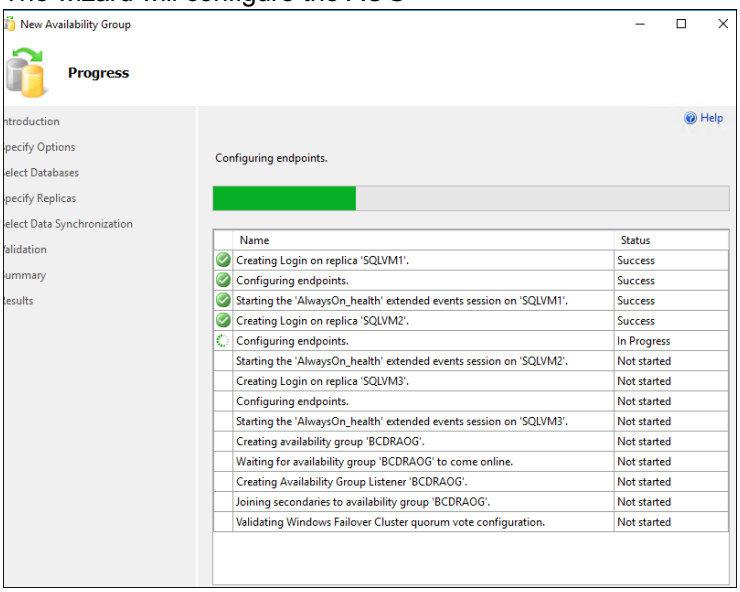
54. On the Validation screen, you should see all green. Select **Next**



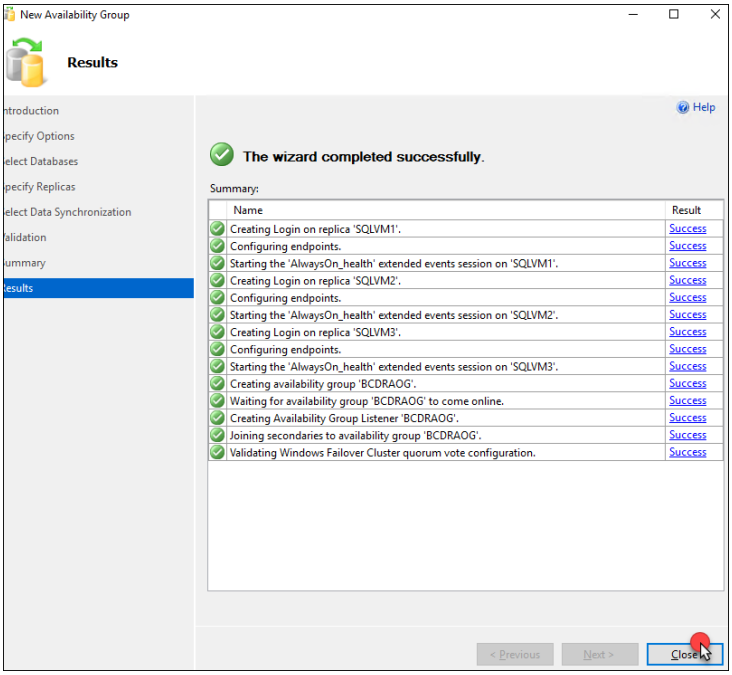
55. On the Summary page select **Finish**



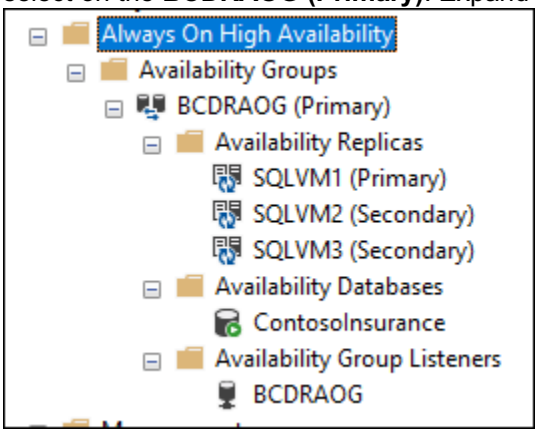
56. The wizard will configure the AOG



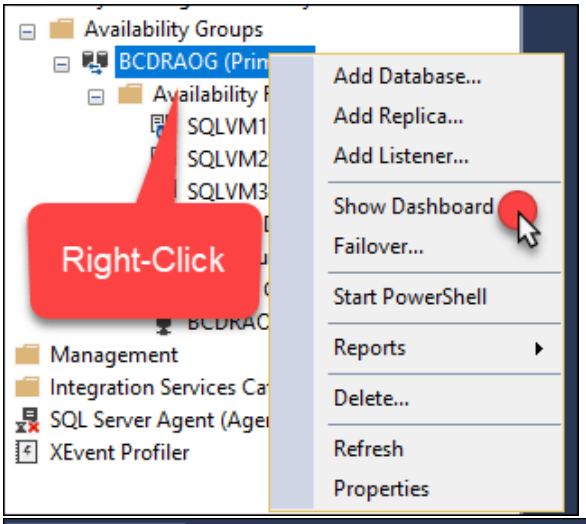
57. Once the AOG is built, select **Close**

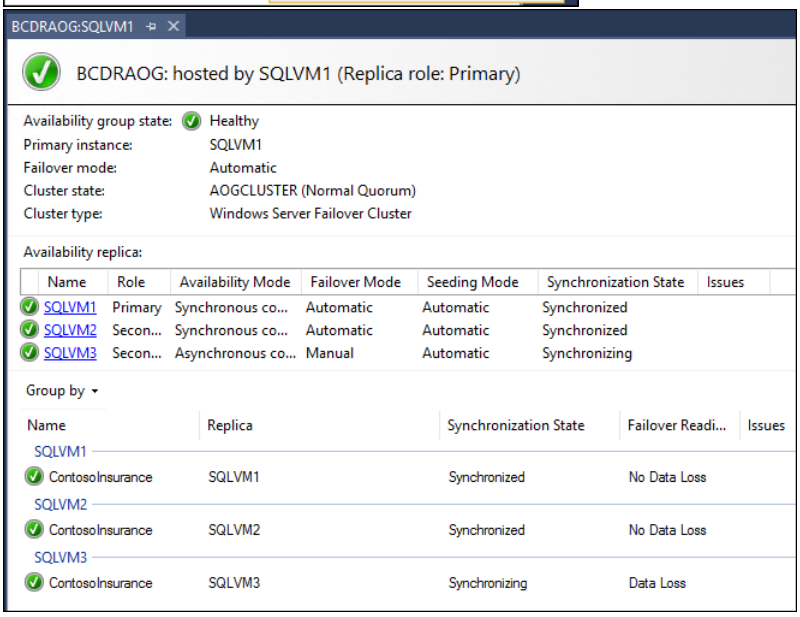


58. Move back to **SQL Management Studio** on **SQLVM1** and open the **Always On High Availability** and then select on the **BCDRAOG (Primary)**. Expand the areas and review them.

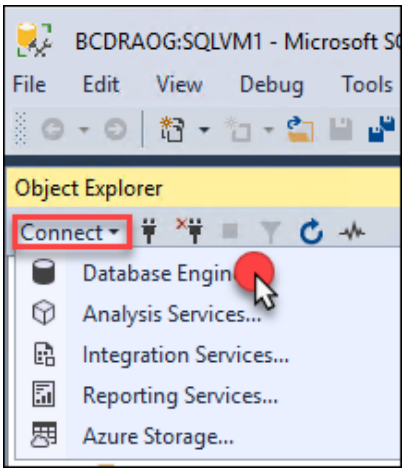


59. Right-click **BCDRAOG (Primary)** and then **Show Dashboard.** You should see that all the nodes have been added and are now “Green”.

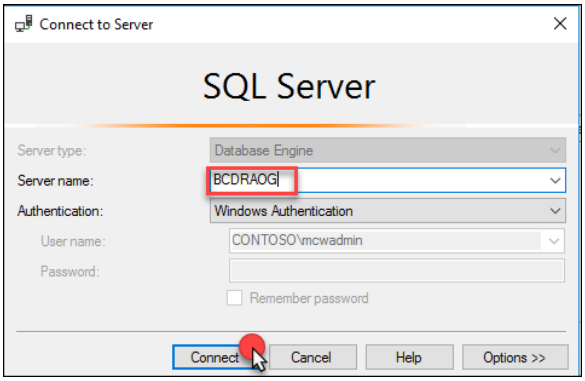




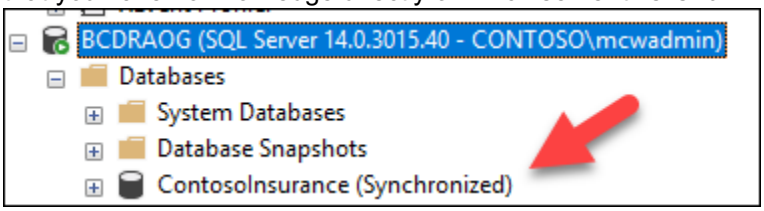
60. Next, select **Connect** and then **Database Engine** in SQL Management Studio



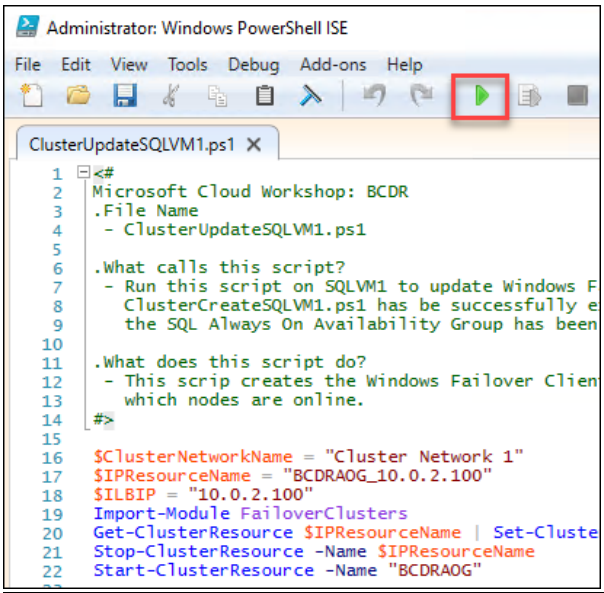
61. Enter **BCDRAOG** as the Server Name. This will be connected to the listener of the group that you created.

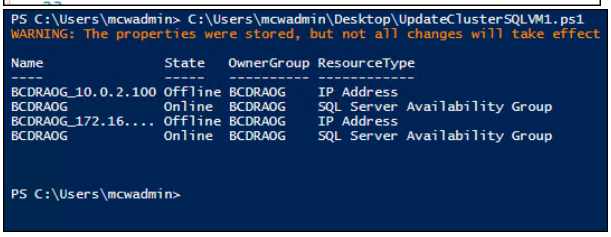


62. Once connected to the **BCDRAOG** you can select **Databases** and will be able to see the database there. Notice that you have no knowledge directly of which server this is running on.

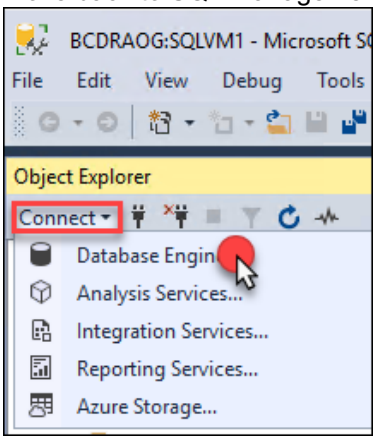


63. Move back to PowerShell ISE on **SQLVM1** and open the PowerShell script named **ClusterUpdateSQLVM1.ps1**. Select the **Play** button. This will update the Failover cluster with the IP Addresses of the Listener that you created for the AOG.

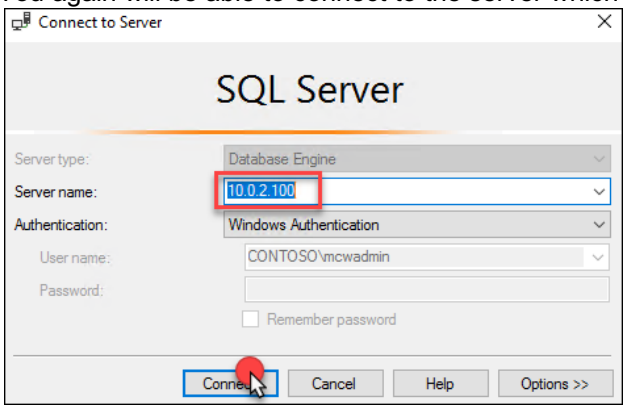




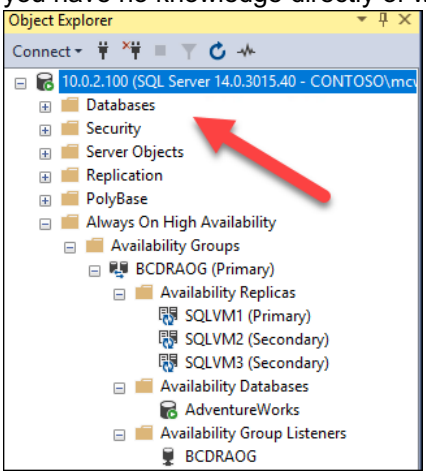
64. Move back to SQL Management Studio and select **Connect** and then **Database Engine**



65. This time put in the IP address of the Internal Load balancer of the **Primary** Site AOG Load Balancer: **10.0.2.100**. You again will be able to connect to the server which is up and running as the master.

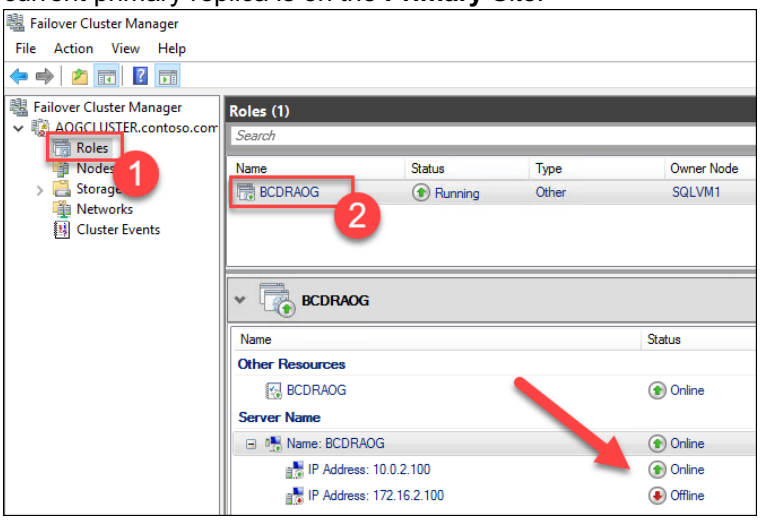


66. Once connected to **10.0.2.100** you can select **Databases** and will be able to see the database there. Notice that you have no knowledge directly of which server this is running on.



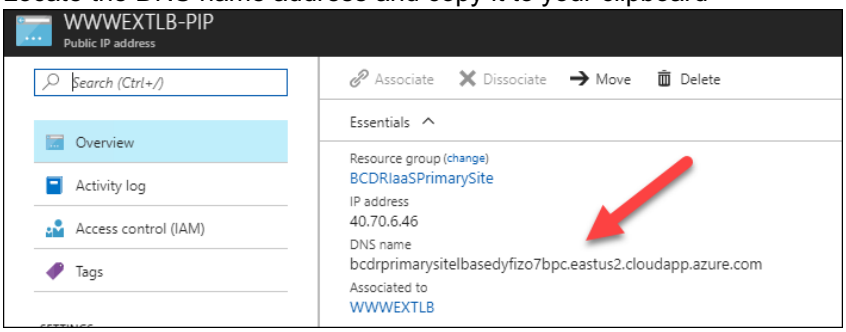
Note: It could take a minute to connect the first time as this is going through the Azure Internal Load Balancer.

67. Move back to Failover Cluster Manager on **SQLVM1**, and you can review the IP Addresses that were added by selecting Roles and **BCDRAOG** and viewing the Resources. Notice how the **10.0.2.100** is Online since the current primary replica is on the **Primary** Site.

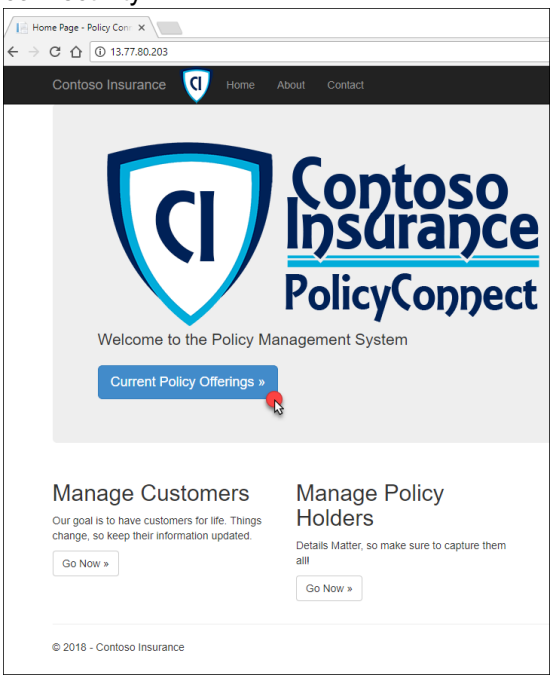


68. Now that the AOG is up and running online the Contoso Insurance Web Application should be available. Minimize the RDP window and from the **LABVM** open the Azure portal and navigate to the resource group **BCDRIaaSPrimarySite**. Select **WWWEXTLB-PIP** which is the Public IP address for the external load balancer **WWWEXTLB** in front of **WEBVM1** and **WEBVM2** in your Primary region.

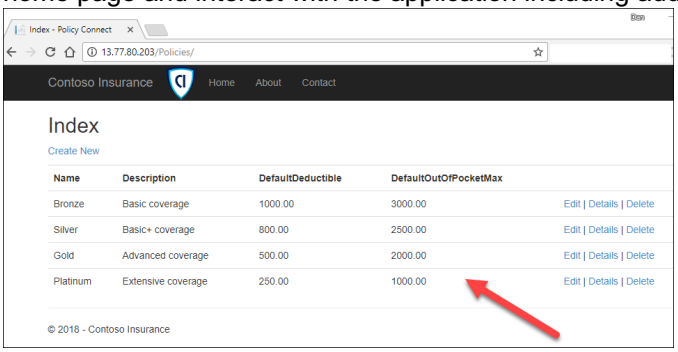
69. Locate the DNS name address and copy it to your clipboard



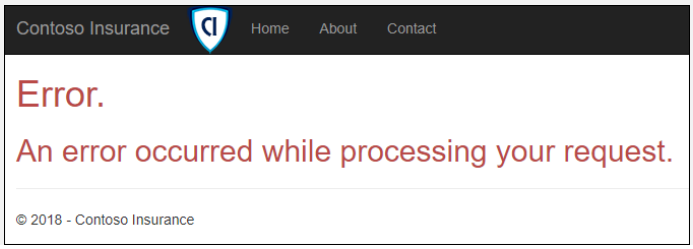
70. Open a new tab in the browser and navigate to the DNS name. The **Contoso Insurance PolicyConnect** application should load in your browser. Select the **Current Policy Offerings** button to check for database connectivity.



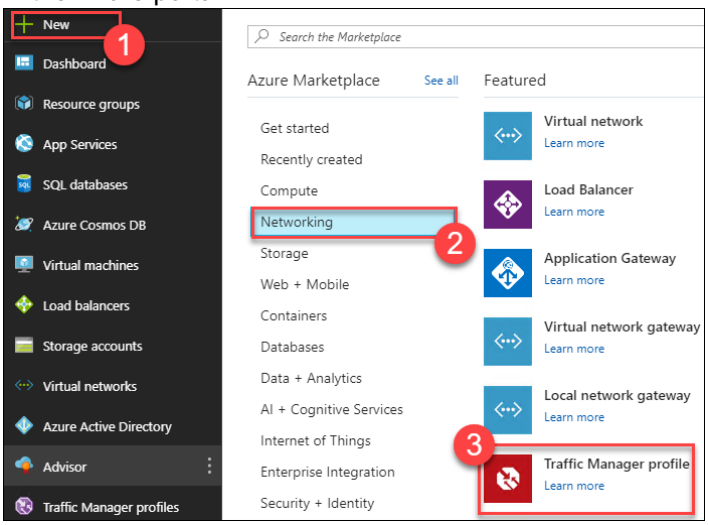
71. If you can see the offerings, then the application is able to access the database. You can also go back to the home page and interact with the application including adding, editing or deleting data.



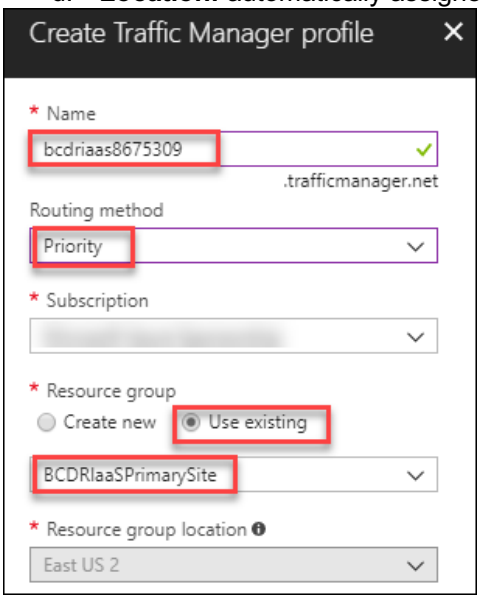
Note: If you see the following screen shot then something is not configured correctly with your environment. The connection string of the application is configured to use the name of **bcdraog.contoso.com** which is the name for the SQL AOG listener. This configuration is part of the connection string located in the **web.config** file which is on **WEBVM1** and **WEBVM2** in the **C:\Inetpub\wwwroot** directory. If you for some reason you named something incorrectly you can make a change to this file and then iisreset /restart from the command line on the WEBVMs.



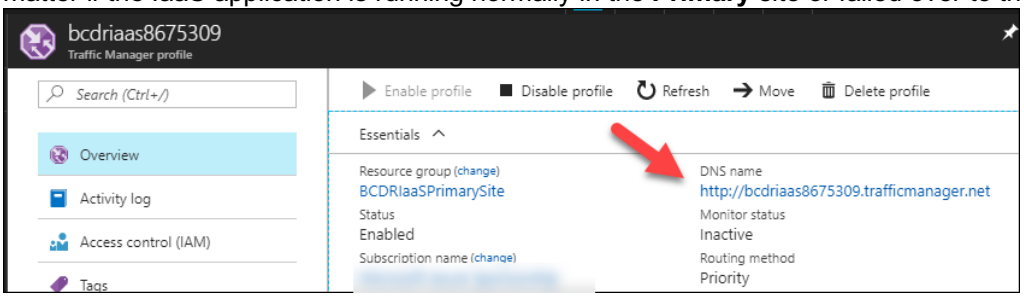
72. Once you have verified that the application is up and running, you will need to build a Traffic Manager to direct traffic to the edge of your Primary and Secondary Site. Select **+NEW, Networking then Traffic Manager profile** in the Azure portal.



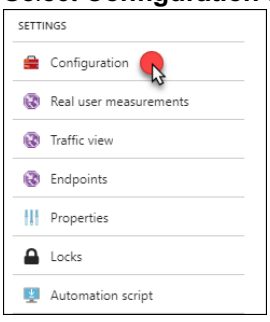
73. Complete the **Create Traffic Manager profile** using the following inputs, then select **Create**:
    a. **Name:** unique name all lowercase using bcdriaasxxx
    b. **Routing method:** Priority
    c. **Resource group:** Use existing / BCDRIaasPrimarySite
    d. **Location:** automatically assigned based on the BCDRIaaSPrimarySite



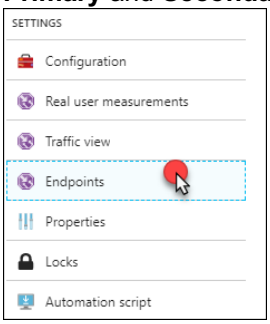
74. Once the Traffic Manager profile is created, open it in the Azure portal. Notice the DNS name. This is the URL that you will use to connect to the application. Once configured this DNS name will always respond and doesn't matter if the IaaS application is running normally in the **Primary** site or failed over to the **Secondary site.**



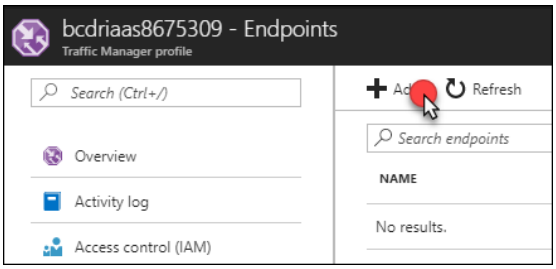
75. Select **Configuration** and review the configurations



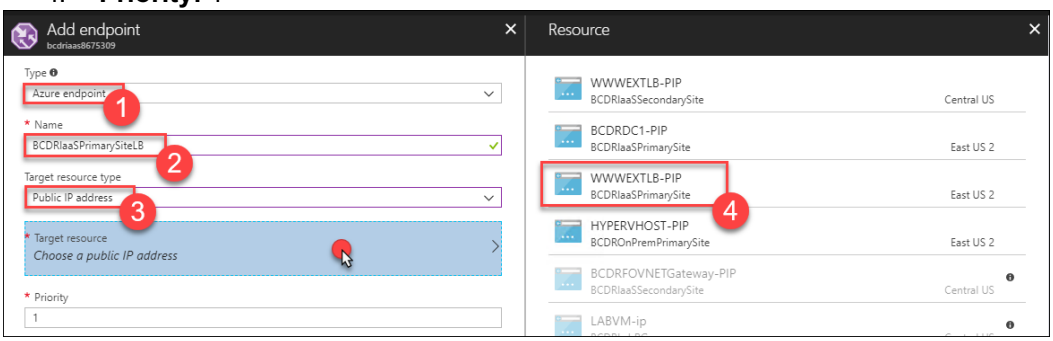
76. Next select **Endpoints**. This is where you will configure the two external load balancers that are located your **Primary** and **Secondary** sites



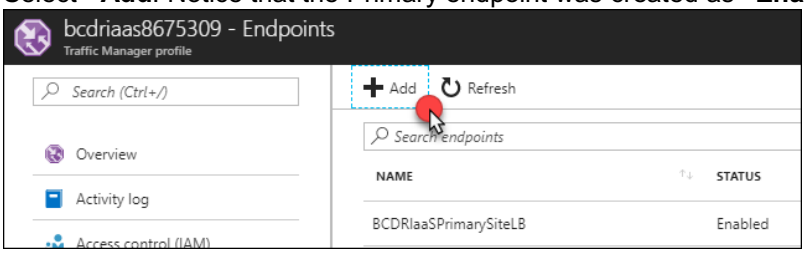
77. Select **+Add**



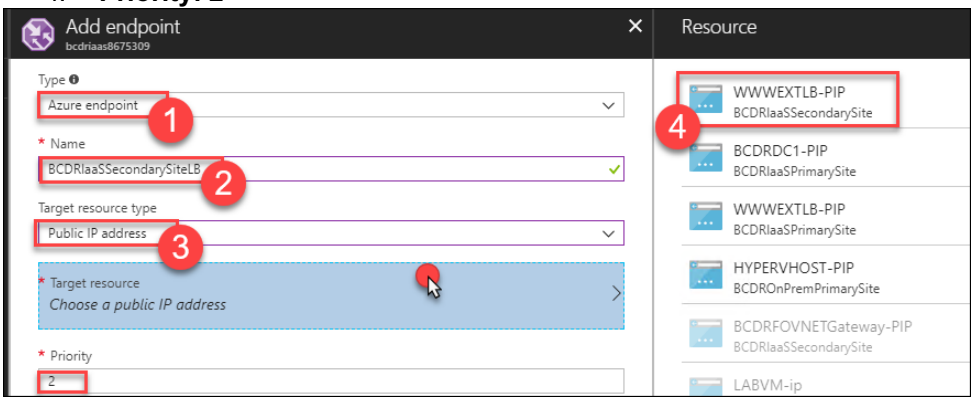
78. Complete the **Add endpoint** using the following inputs and then select **OK:**
    a. **Type:** Azure endpoint
    b. **Name:** BCDRIaaSPrimarySiteLB
    c. **Target resource type:** Public IP address
    d. **Target resource:** Choose a public IP address
    e. **Resource:** WWWEXTLB-PIP in the BCDRIaaSPrimarySite
    f. **Priority:** 1



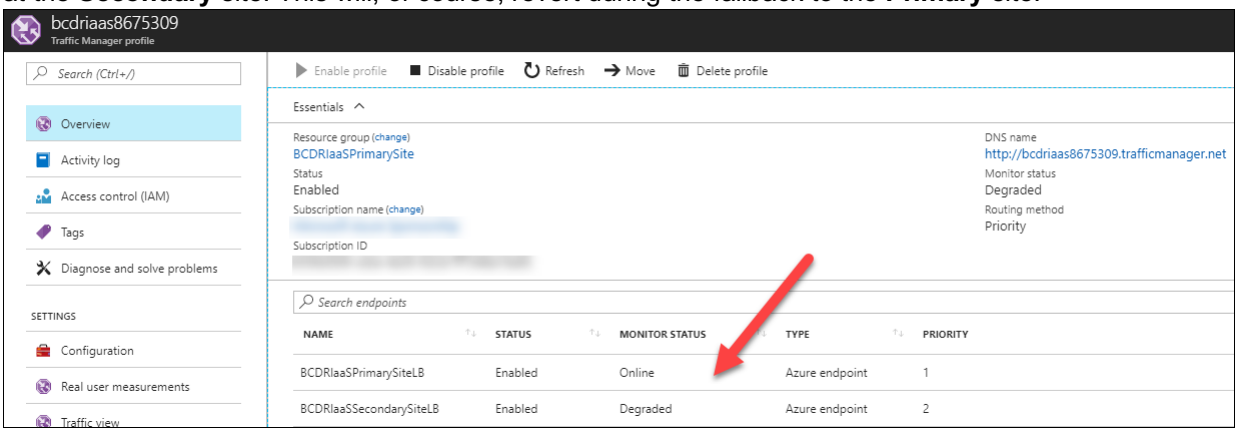
79. Select **+Add**. Notice that the Primary endpoint was created as **"Enabled"**



80. Complete the **Add endpoint** using the following inputs and then select **OK:**
    a. **Type:** Azure endpoint
    b. **Name:** BCDRIaaSSecondarySiteLB
    c. **Target resource type:** Public IP address
    d. **Target resource:** Choose a public IP address
    e. **Resource:** WWWEXTLB-PIP in the BCDRIaaSSecondarySite
    f. **Priority:** 2

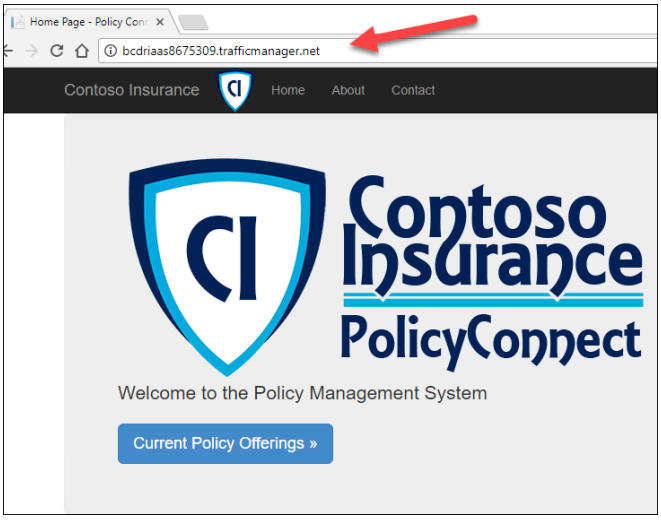


81. Once the second endpoint has been added, select **Overview**. Notice that the Primary endpoint is set to **Priority 1**. This means that traffic will always be directed to the **Primary** site unless it is down. The Traffic Manager will monitor the Endpoints, and if the **Primary** site moves to a **Monitor Status** of **Degraded**, then the Traffic Manager will direct traffic to the **Secondary** site. The current **Monitor Status** shows that the **Primary** site is **Online** and the Secondary site is **Degraded**. During a failover using ASR, the **Primary** site will move to **Degraded,** and the **Secondary** will move to **Online**. This will allow for traffic to flow to the failed over IaaS infrastructure now running at the **Secondary** site. This will, of course, revert during the failback to the **Primary** site.



Note: All of this is automatic and easily configured with a vanity domain by adding a C NAME record in DNS to point to the DNS name of the Traffic Manager. This would allow for a site like www.contoso.com to resolve to the DNS name of the traffic manager. The users will never know that the site is failed over or failed back.

82. Select the DNS name of the Traffic manager the Policy Connect web application will load. This is connecting to the **WWWEXTLB** External Load Balancer that is in front of **WEBVM1** and **WEBVM2** running in the **Primary** Site in **BCDRIaaSPrimarySite** resource group and connecting to the SQL Always On Listener at the same location.

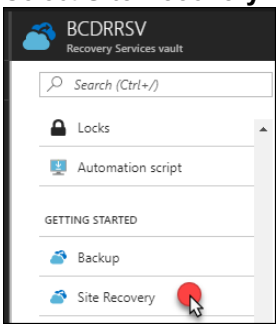


Task 3: Configure IaaS for region to region failover

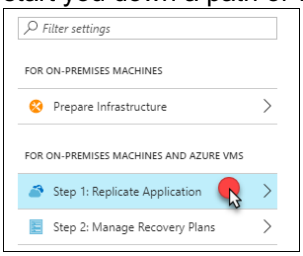
In this task the WEBVM1 and WEBVM2 will be configured to replicate from the Primary Site to the Secondary site to support an Azure region to region failover. This will consist of configuring the VMs to replicate and integrating with the Azure Automation to failover the SQL Always On group from the Primary Site to the Secondary. Once the failover is complete the website will again answer to the Traffic Manager DNS name.

1. From the Azure portal on **LABVM**, open the **BCDRRSV** Recovery Services Vault located in the **BCDRAzureSiteRecovery** resource group.

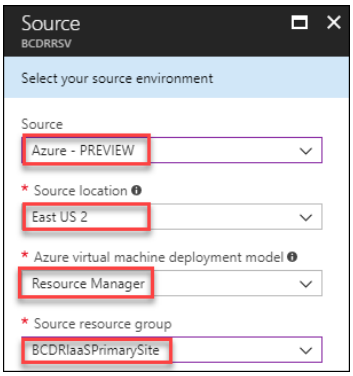
2. Select **Site Recovery** in the **Getting Started** area of **BCDRRSV** blade



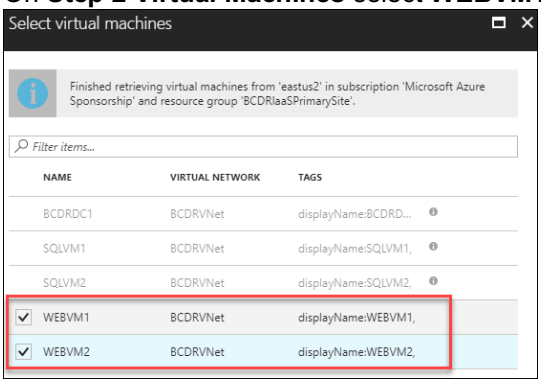
3. Next, select **Step 1: Replicate Application** in the **For On-Premises Machines and Azure VMs** section. This will start you down a path of various steps to configure your WEBVMs that are running on Azure at your Primary Site.



4. On **Step 1 Source** select the following inputs and then select **OK**:
    a. **Source**: Azure-PREVIEW
    b. **Source Location**: Your Primary Region
    c. **Azure virtual machine deployment model**: Resource Manager
    d. **Source resource group**: BCDRIaaSPrimarySite

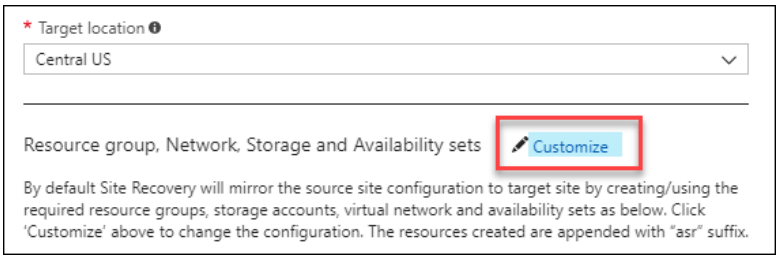


5. On **Step 2 Virtual Machines** select **WEBVM1** and **WEBVM2** and select **OK**

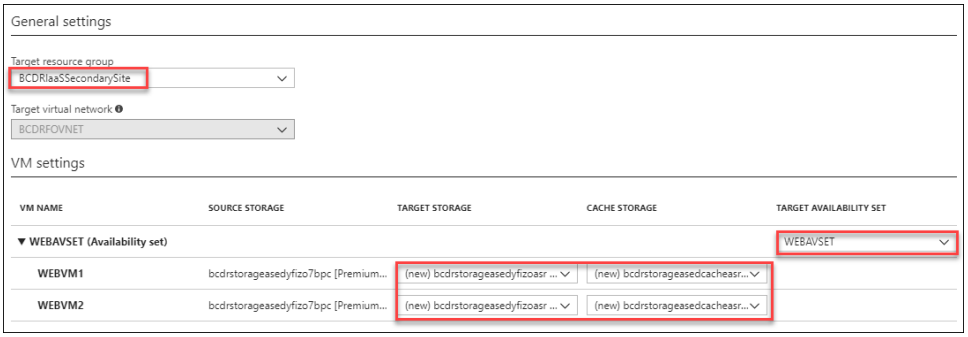


6. On the **Configure settings** blade, first select the **Target location** as your **Secondary** Site Azure Region

7. Select **Customize**

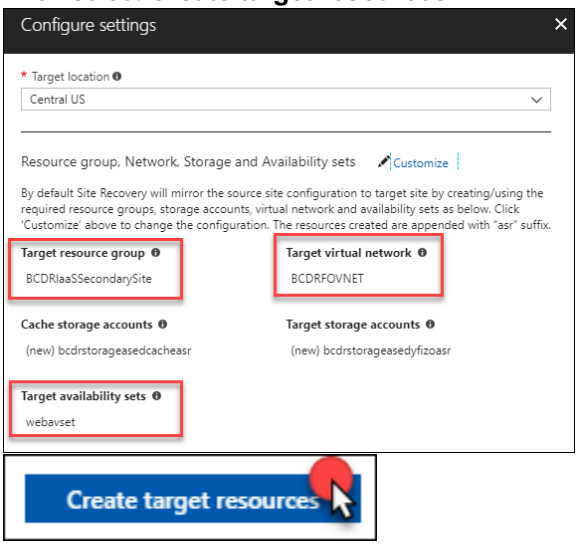


8. Update the rest of the blade using the following inputs and the select **OK**:
   a. **Target resource group:** BCDRIaaSSecondarySite
   b. **Target virtual network:** BCDRFOVNET
   c. **Target storage:** accept the new account
   d. **Target storage:** accept the new account
   e. **Target Availability Set:** WEBAVSET



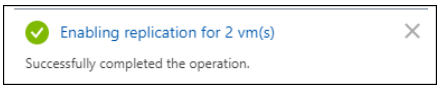
Note: Double check these selections. They are critical to your Failover.

9. Then select **Create target resources**

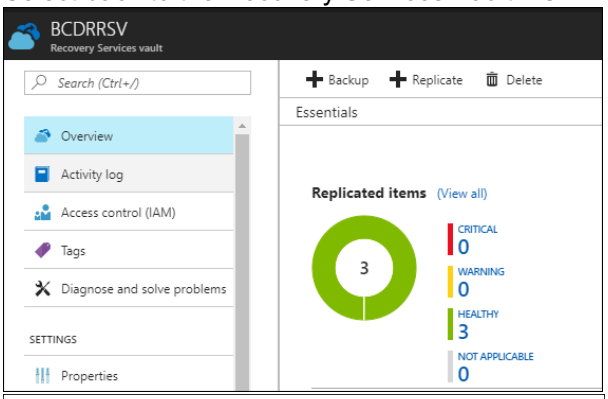


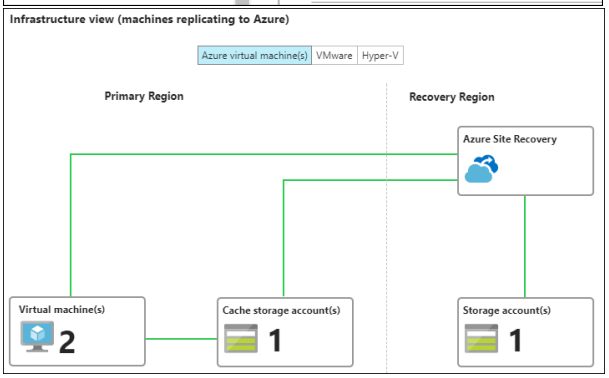
10. Then select **Enable replication**

11. The Azure portal will start the deployment. This will take about 10 minutes to complete and you will receive a notification once it has

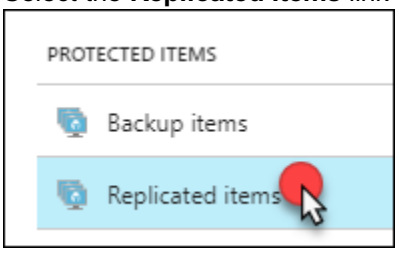


12. Select back to the Recovery Services Vault **BCDRRSV** and you will now see that 3 items are replicated

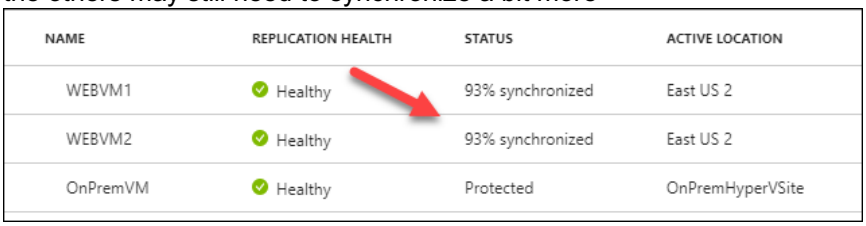




13. Select the **Replicated Items** link under **Protected Items**

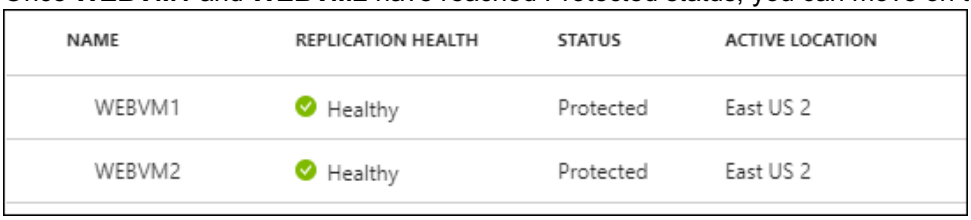


14. You should see three items: **OnPremVM**, **WEBVM1** and **WEBVM2**. The **OnPremVM** will show as protected and the others may still need to synchronize a bit more



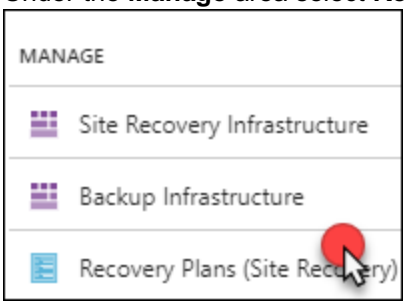
NAME	REPLICATION HEALTH	STATUS	ACTIVE LOCATION
WEBVM1	Healthy	93% synchronized	East US 2
WEBVM2	Healthy	93% synchronized	East US 2
OnPremVM	Healthy	Protected	OnPremHyperVSite

15. Once **WEBVM1** and **WEBVM2** have reached Protected status, you can move on to the next step

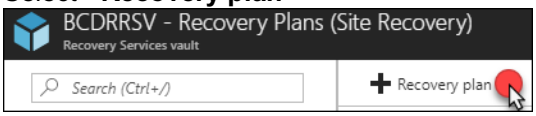


Note: It can take up to 30 minutes for this action to complete.

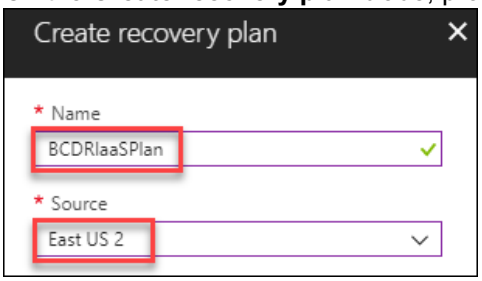
16. Under the **Manage** area select **Recovery Plans (Site Recovery)**



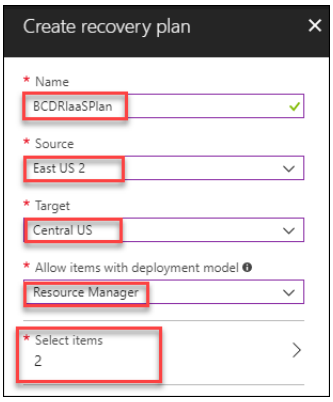
17. Select **+Recovery plan**



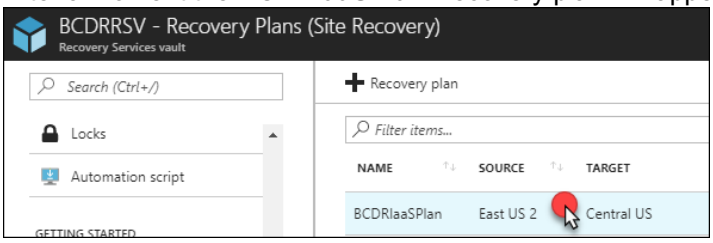
18. On the **Create recovery plan** blade, provide the Name: **BCDRIaaSPlan** and Source: select your **Primary** site



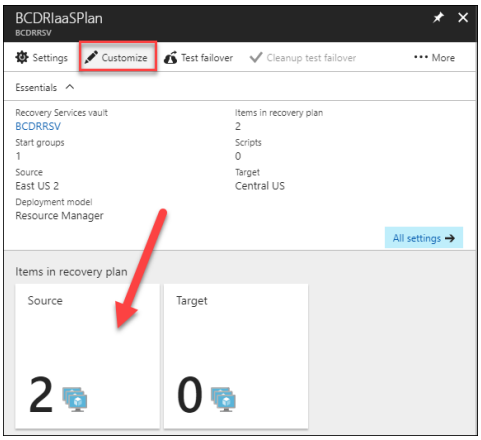
19. Complete the rest of the blade using the following inputs and then select **OK**:
    a. **Target:** Secondary region
    b. **Allow items with deployment model:** Resource Manager
    c. **Select Items:** Select WEBVM1 and WEBVM2



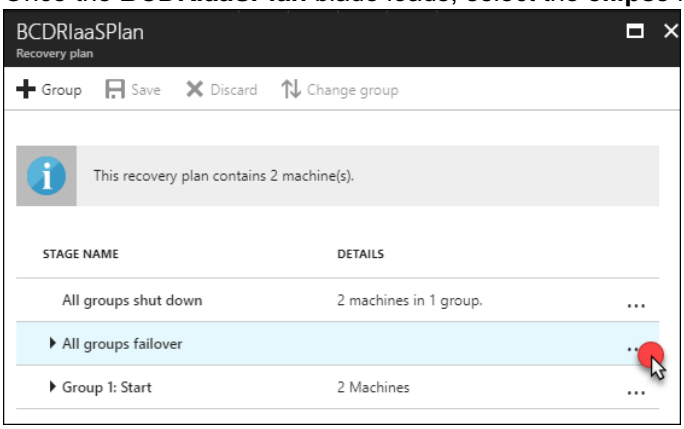
20. After a moment the **BCDRIaaSPlan** Recovery plan will appear, select it to review



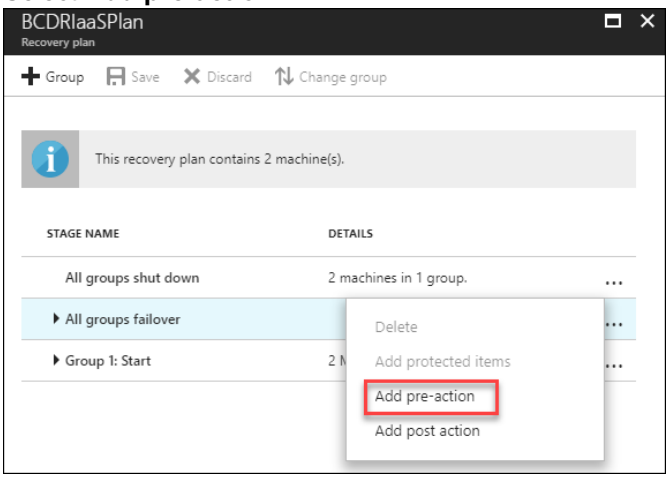
21. When the **BCDRIaaSPlan** loads **notice** that it shows **2 VMs in the Source** which is your **Primary** Site. Then select **Customize**.



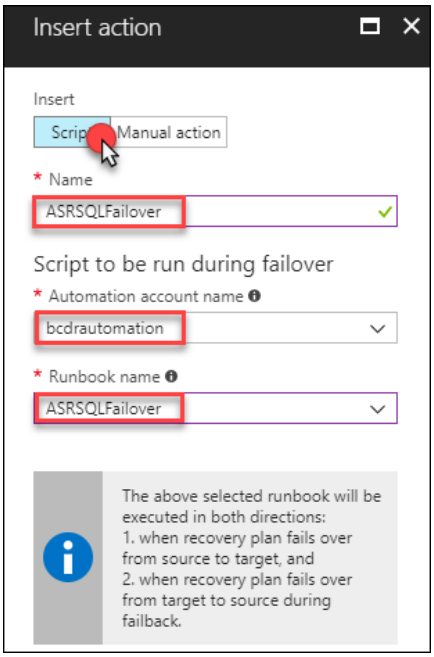
22. Once the **BCDRIaaSPlan** blade loads, select the **ellipse** next to **All groups failover**



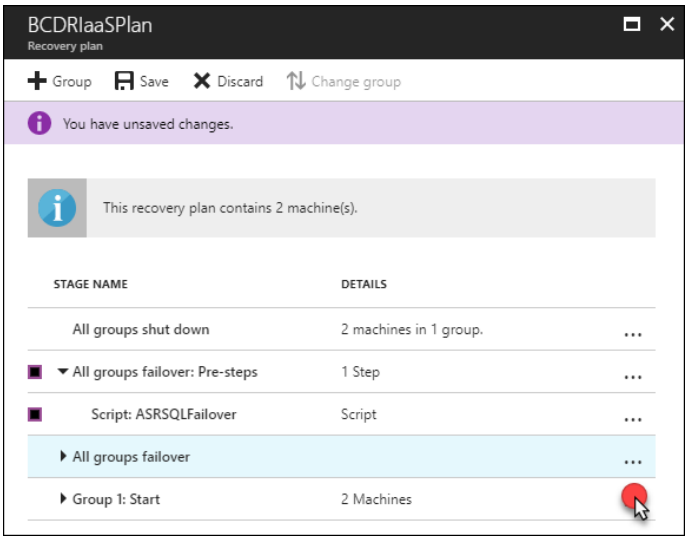
23. Select **Add pre-action**



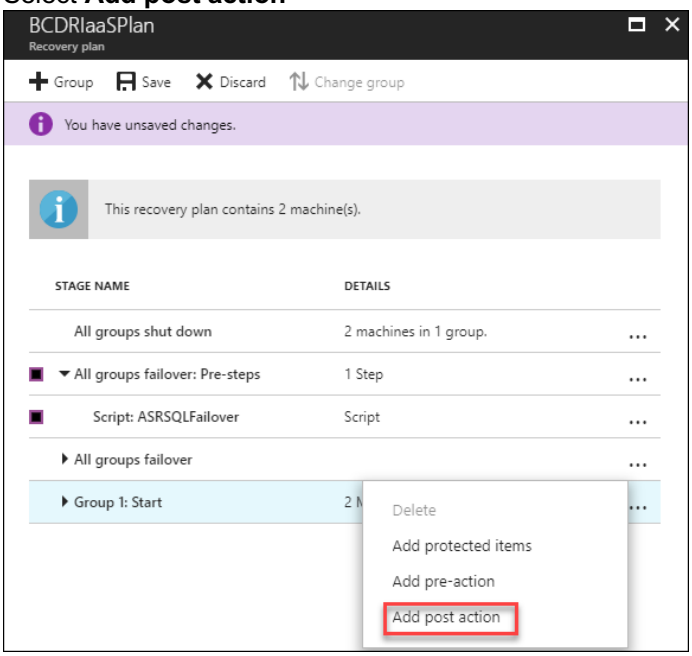
24. On the **Insert action** blade, select **Script** and then provide the name: **ASRSQLFailover.** Ensure that your Azure Automation account is selected and then chose the Runbook name: **ASRSQLFailover**. Select **OK**.



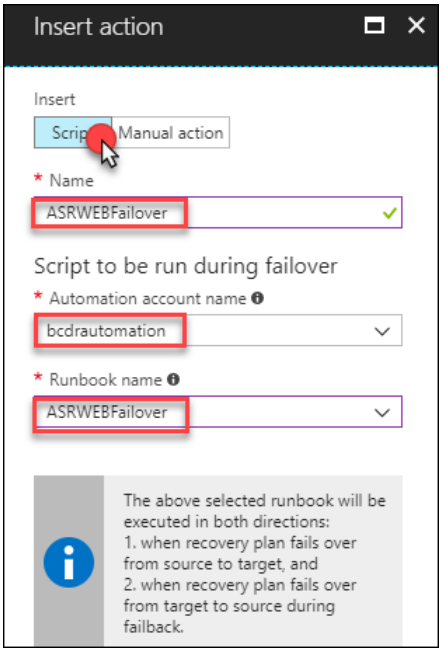
25. Once the **BCDRIaaSPlan** blade loads, select the **ellipse** next to **Group 1: Start**



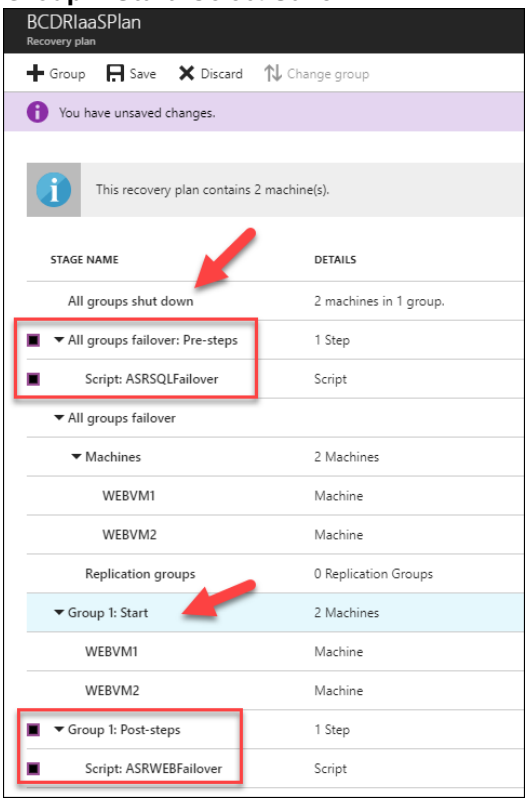
26. Select **Add post action**



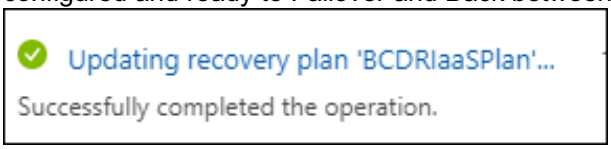
27. On the **Insert action** blade, select **Script** and then provide the name: **ASRWEBFailover.** Ensure that your Azure Automation account is selected and then chose the Runbook name: **ASRWEBFailover**. Select **OK**.



28. Make sure that your **Pre-steps** are running under **All groups failover** and the **Post-steps** are running under the **Group1: Start.** Select **Save**.



29. After a minute, the portal will provide a successful update notification. This means that your machines are fully configured and ready to Failover and Back between the **Primary** and **Secondary** regions.

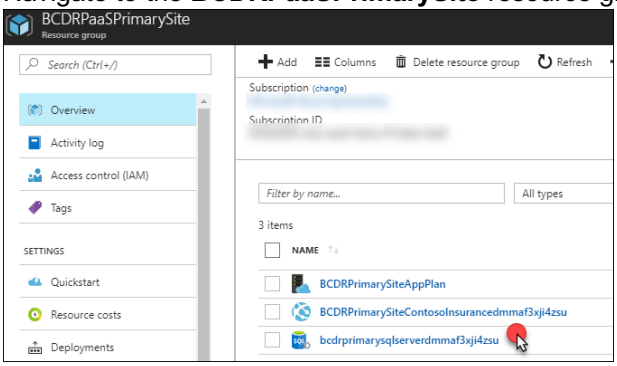


Task 4: Configure PaaS for region to region failover

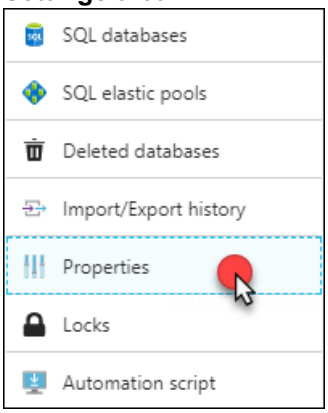
In this task you will deploy the website to App Services using Visual Studio, migrate a database to Azure SQL Database and configure it for high-availability using an Azure SQL Database Failover Group. The Traffic Manager will be used to direct traffic to the closest front end to the user. If there is a failover of the database it will happen transparently, and the users will never know there was an outage. There is no reconfiguration required for this to function properly.

1. From the **LABVM**, open the Azure portal at: https://portal.azure.com

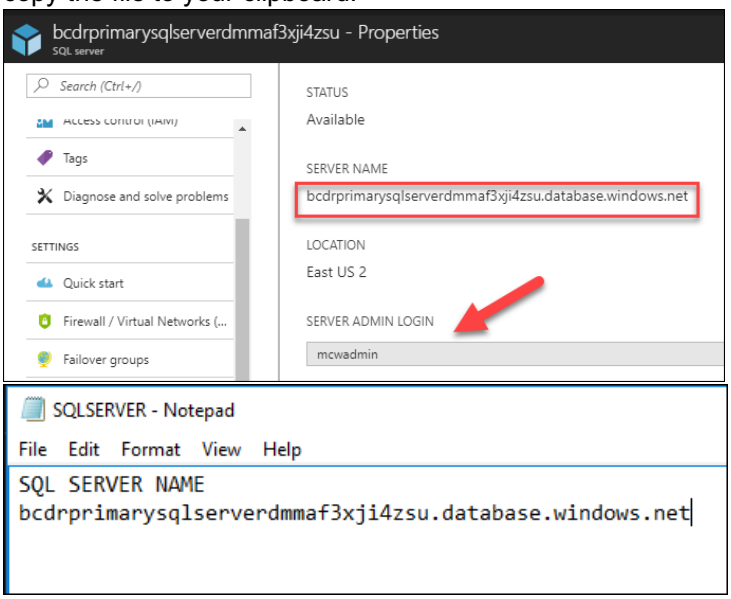
2. Navigate to the **BCDRPaaSPrimarySite** resource group. Select the SQL Server resource



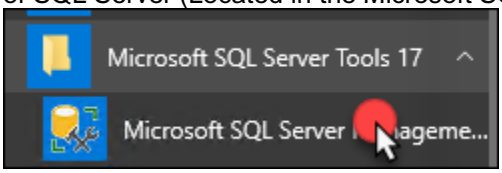
3. This server will host your SQL Database for the Contoso Insurance Web App. Select **Properties** under the **Settings** area.



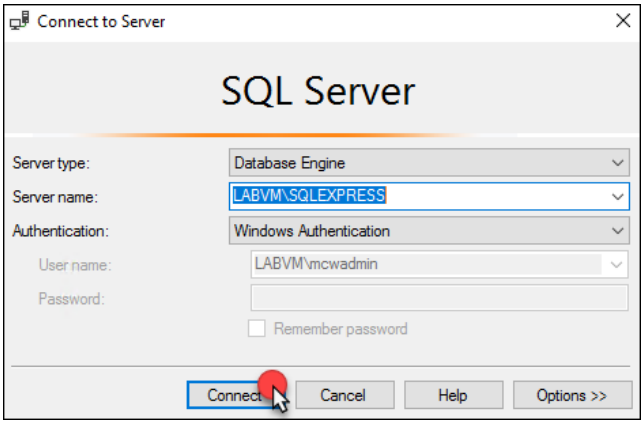
4. Copy the name of the SQL Server to notepad. Also, notice that the Server Admin Login is the same. Save this file as **C:\HOL\Deployments\SQLSERVER.txt.** Navigate to this folder using Windows Explorer and right-click and copy the file to your clipboard.



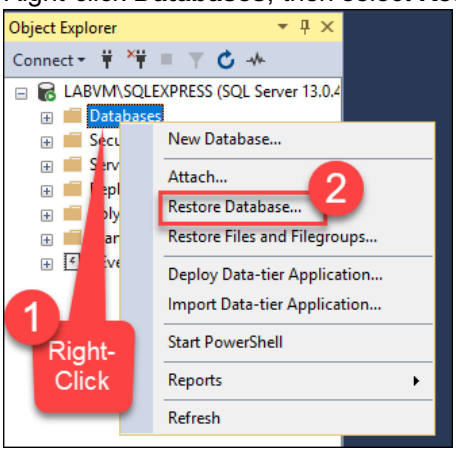
5. Use the Start menu to launch **Microsoft SQL Server Management Studio 17** and connect to the local instance of SQL Server (Located in the Microsoft SQL Server Tools 17 folder)



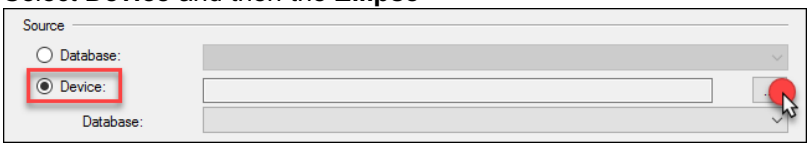
6. Select **Connect** to Sign On to the Local **SQLEXPRESS** on **LABVM**



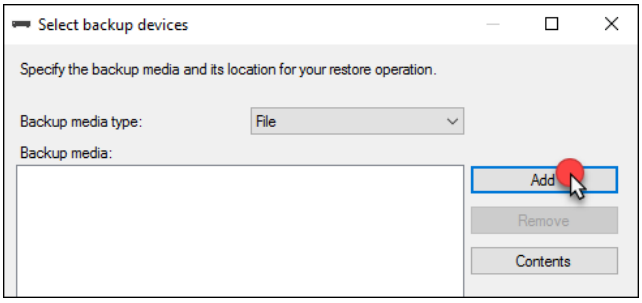
7. Right-click **Databases**, then select **Restore Database**



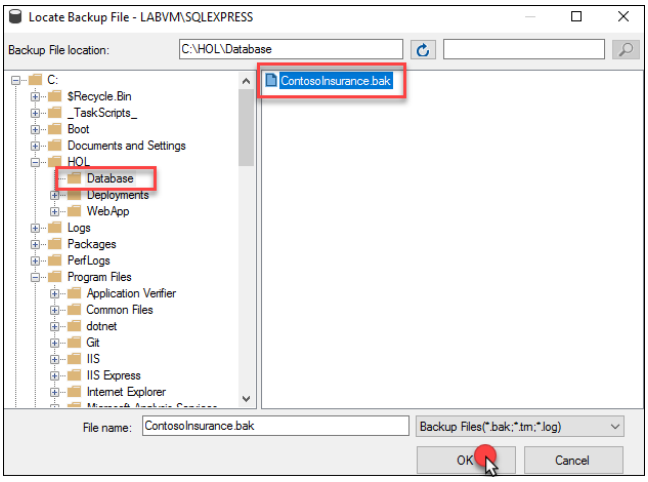
8. Select **Device** and then the **Ellipse**



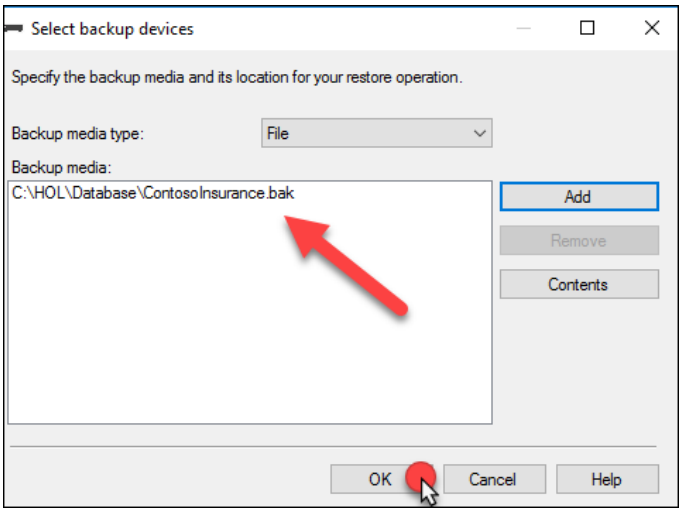
9. Select **Add**



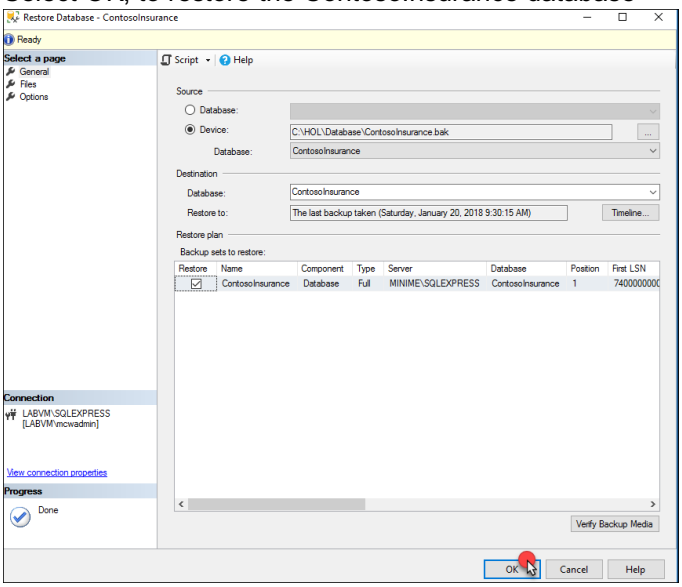
10. Navigate the folder menu and locate the **C:\HOL\Database** folder and then select on **ContosoInsurnace.bak** and then **OK**



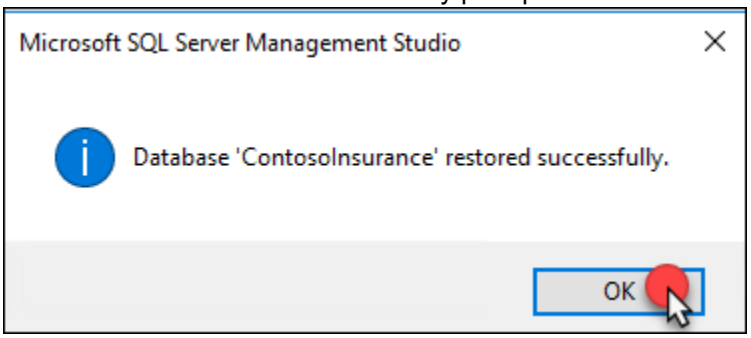
11. Select **OK**



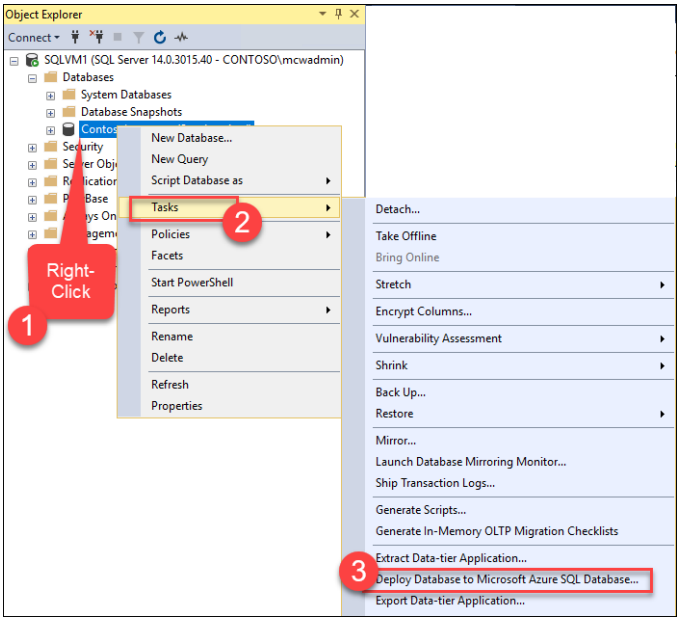
12. Select OK, to restore the ContosoInsurance database



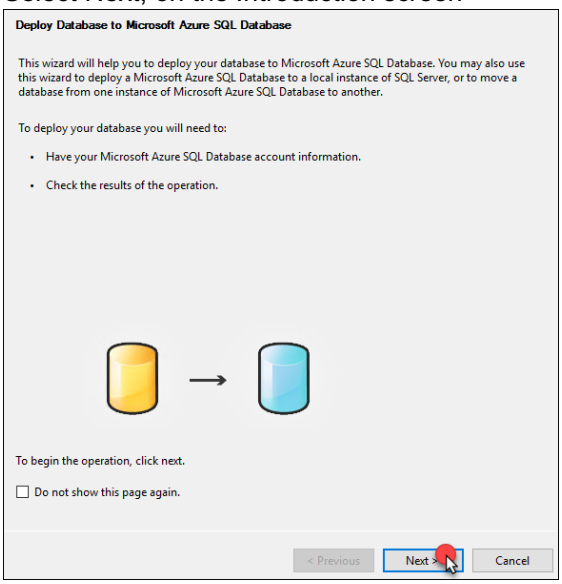
13. Select OK at the restored successfully prompt



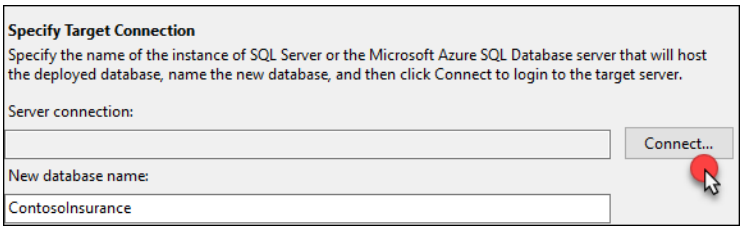
14. **Expand Databases** and then right-click on the **ContosoInsurance** database, select **Tasks**, then **Deploy Database to Microsoft Azure SQL Database**



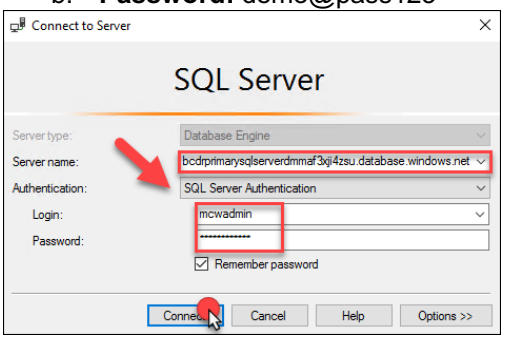
15. Select Next, on the Introduction screen



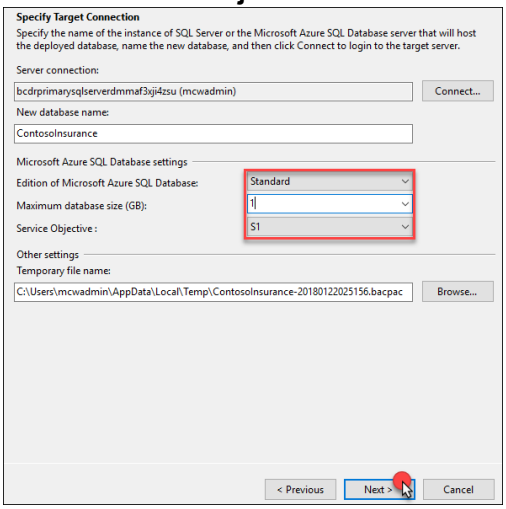
16. Select **Connect**



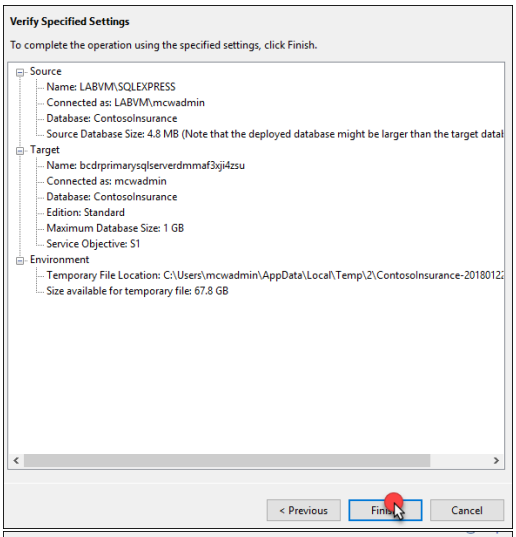
17. In the **Connect to SQL Server** screen, copy the name of your Azure SQL Server from the **SQLSERVER.TXT**. Change the Authentication to **SQL Server Authentication** and enter the credentials for the server then select **Connect.**
    a. **Login:** mcwadmin
    b. **Password:** demo@pass123

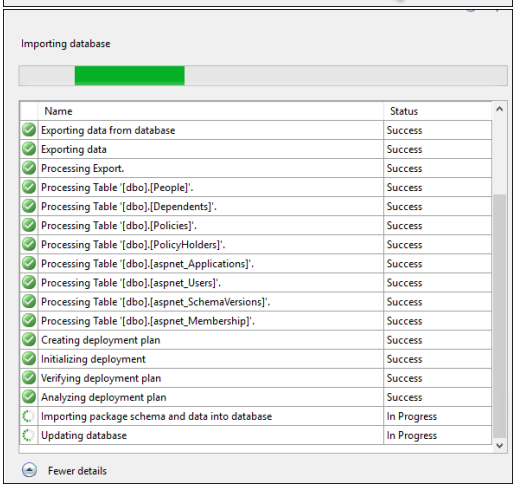


18. Update the remainder of the Deployment Settings screen using these inputs and then select **Next**:
    a. **Edition of Azure SQL database:** Standard
    b. **Maximum database size (GB):** 1
    c. **Service Objective:** S1

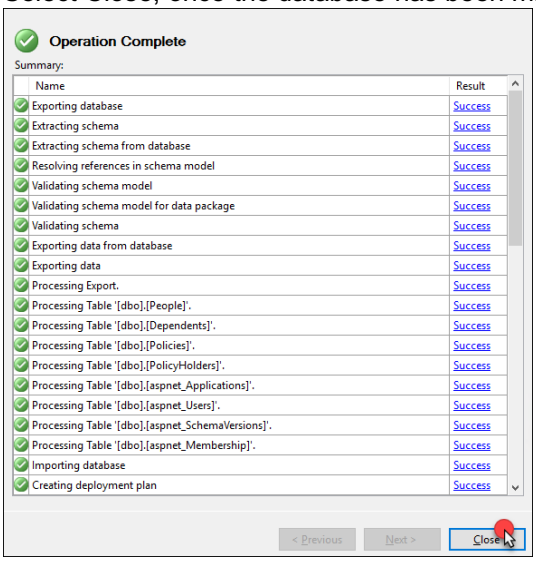


19. Select **Finish** and allow the Database to be migrated to Azure

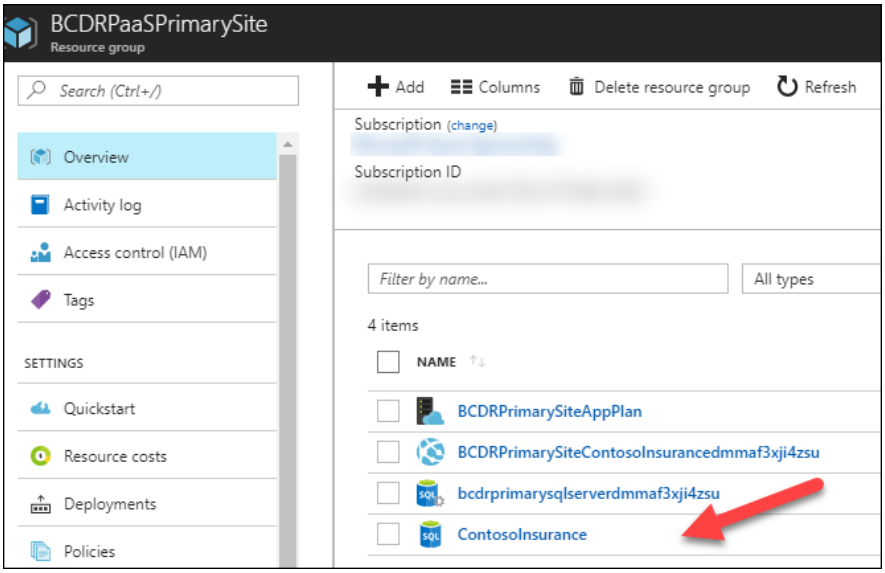




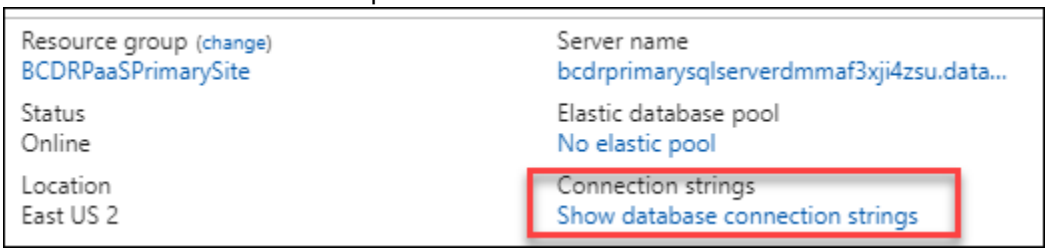
20. Select Close, once the database has been migrated to Azure SQL Database



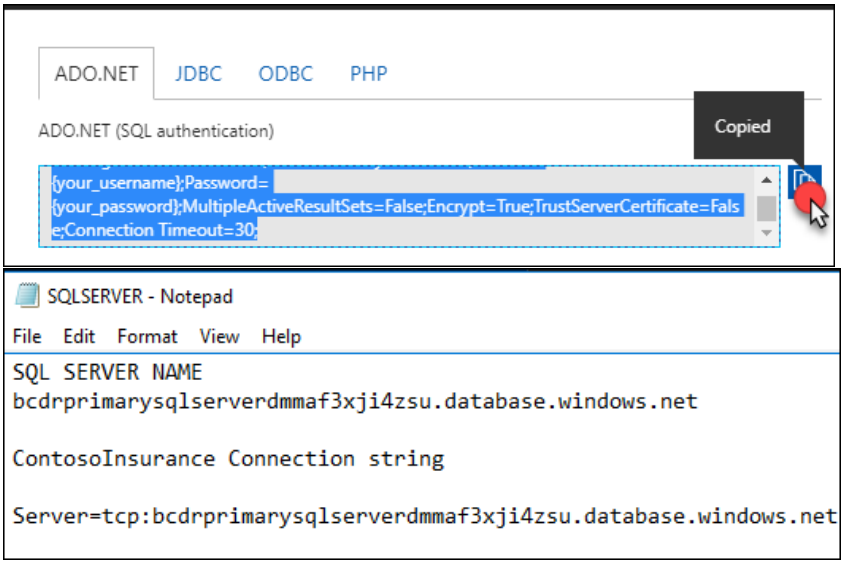
21. Move back to the Azure portal on **LABVM**. Open the **BCDRPaaSPrimarySite** resource group and notice that there is a new SQL Database called **ContosoInsurance**



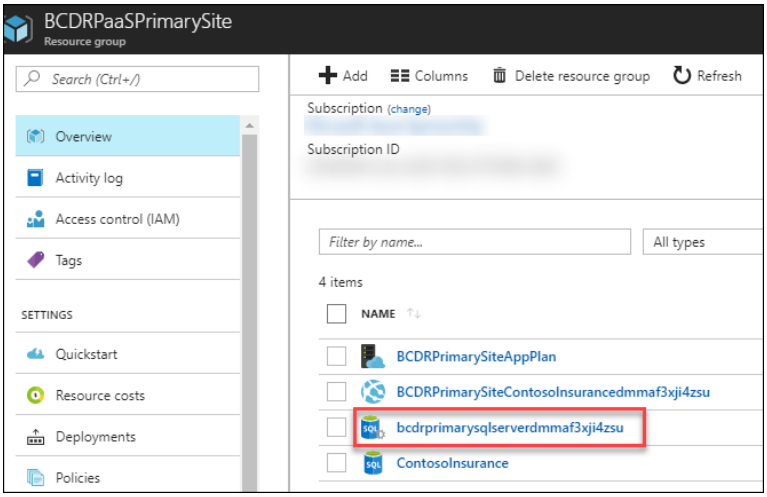
22. Select **ContosoInsurance** to open the resource then select **Show database connection strings**



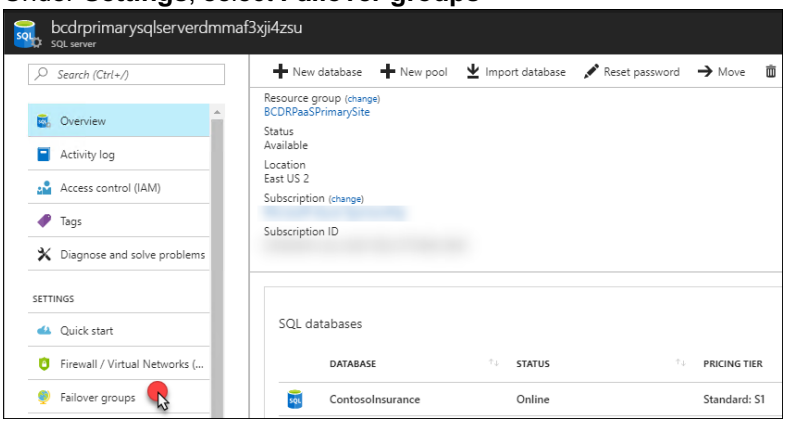
23. Select the **Copy to clipboard** link to capture the connection string and then paste it to your **SQLSERVER.TXT** file



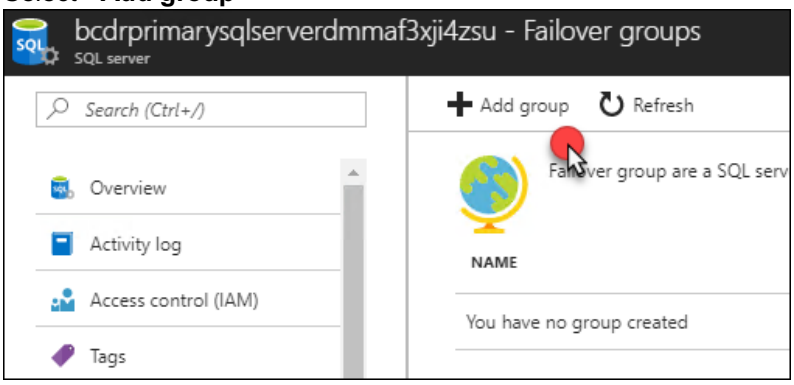
24. In the Azure portal, move back to the **BCDRPaaSPrimarySite** resource group and then select the **SQL Server** resource



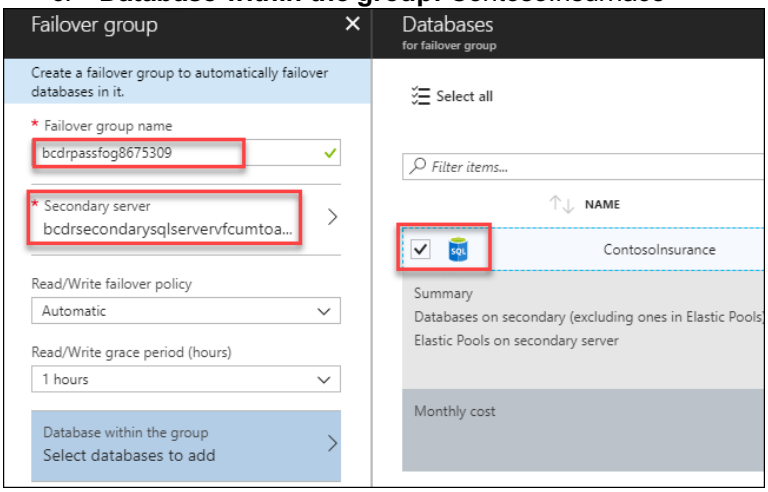
25. Under **Settings**, select **Failover groups**



26. Select **+Add group**

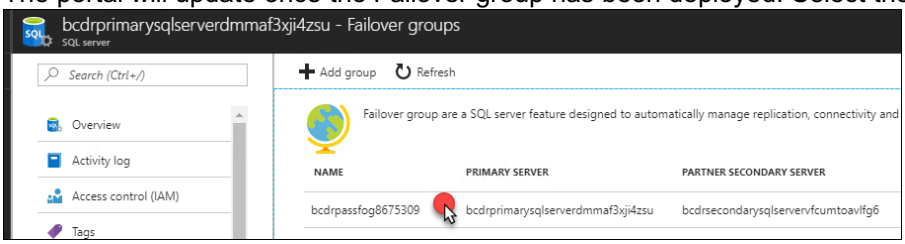


27. Complete the **Failover group** blade using these inputs and then select **Create:**
    a. **Failover group name:** enter a lowercase unique name 3-24 characters using bcdrpassfogxxx
    b. **Secondary Server:** select the secondary SQL Server from your BCDRPaaSSecondarySite
    c. **Database within the group:** ContosoInsurnace

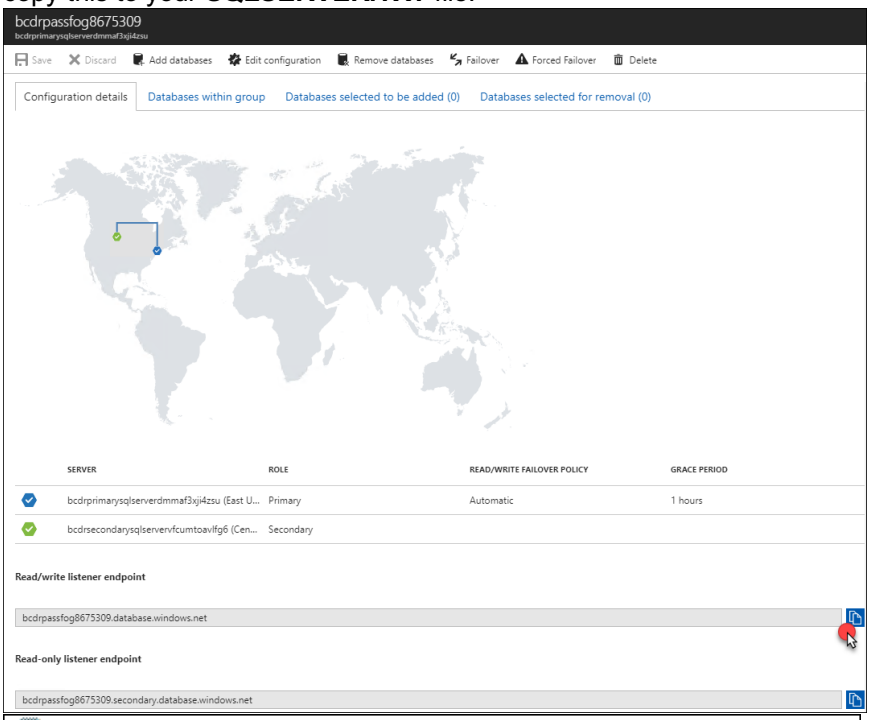


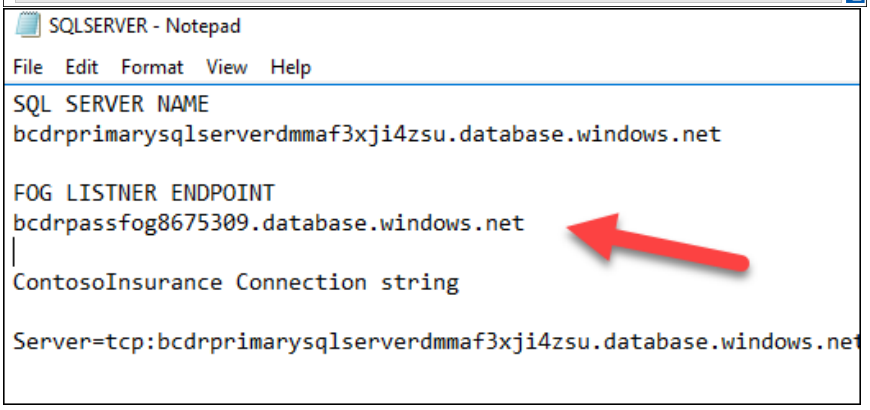
28. The portal will submit a deployment

29. The portal will update once the Failover group has been deployed. Select the **group**

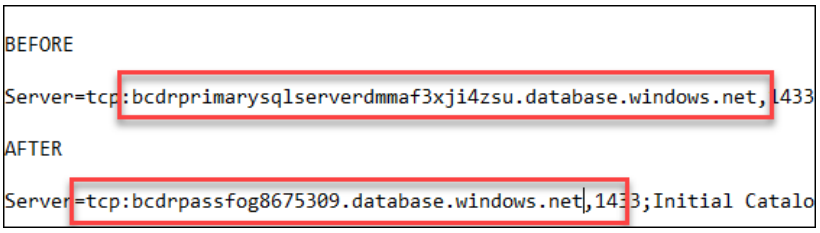


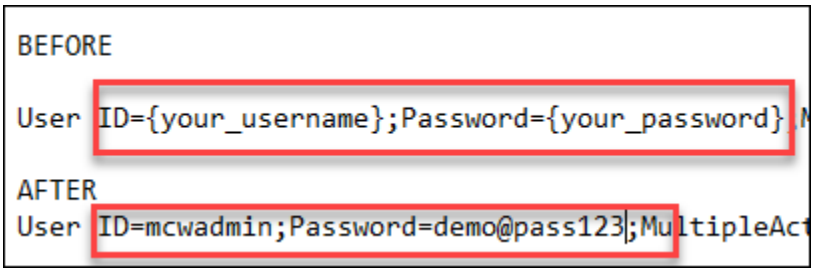
30. The Failover group (FOG) portal will appear. Copy the name of the Listener endpoint to your clipboard and then copy this to your **SQLSERVER.TXT** file.



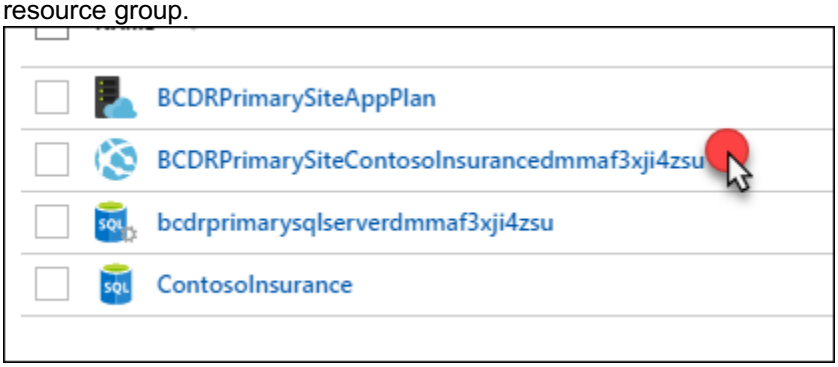


31. In the **SQLSERVER.TXT** file, **update the server name** in the connection string with the name of the FOG listener endpoint. Also, **change the user name and password** to the credentials for the SQL Server:
    a. **Username:** mcwadmin
    b. **Password:** demo@pass123

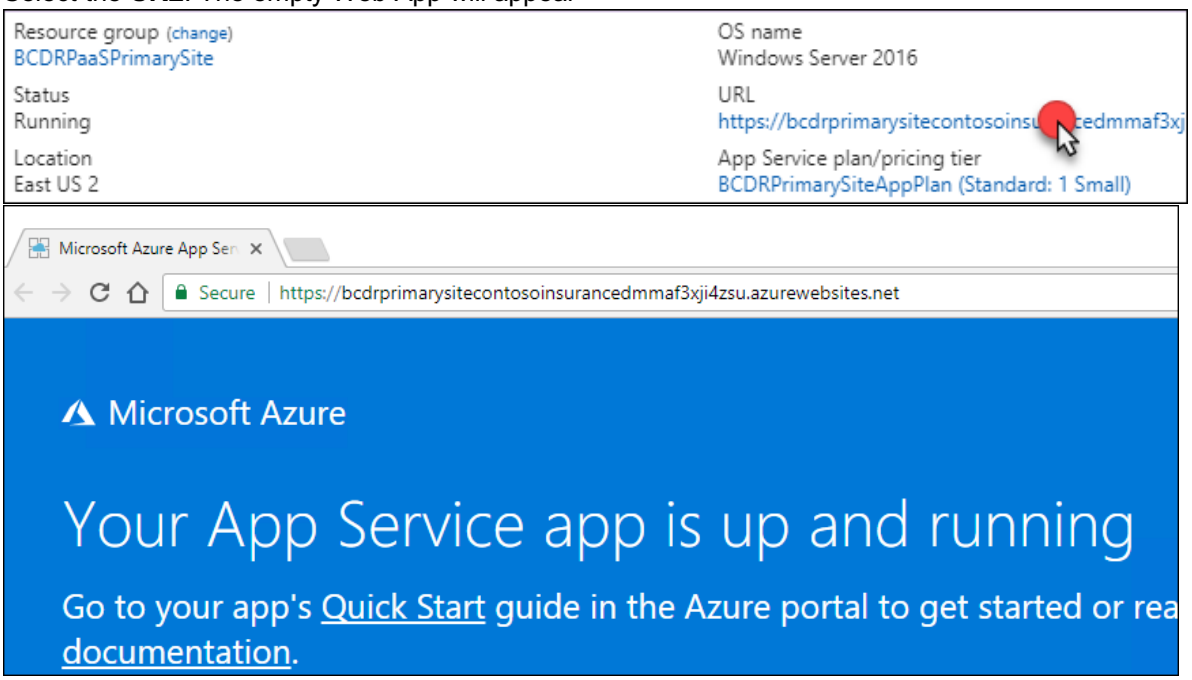




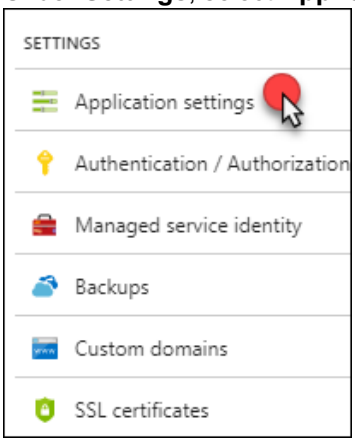
32. The new connection string will be used for the Web App. This will ensure that when the SQL Database is failed over that the server is always pointing to the Failover group. Open the Web App in the **BCDRPaaSPrimarySite** resource group.



33. Select the **URL**. The empty Web App will appear



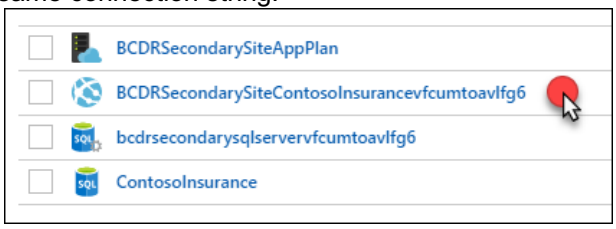
34. Under **Settings**, select **Application Settings**



35. Scroll down to the **Connection strings** settings and add a new connection string using the following inputs, then select **Save**.
    a. **Name:** PolicyConnect
    b. **Value:** Paste in the updated string you created with the Failover group name from the SQLSERVER.TXT file

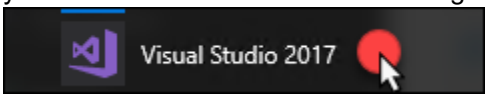
Note: You must use the Name **PolicyConnect**. This is the name that the recognized by the Application in the source code.

36. Repeat the same procedure on the Web App located in the **BCDRPaaSSecondarySite** resource group using the same connection string:

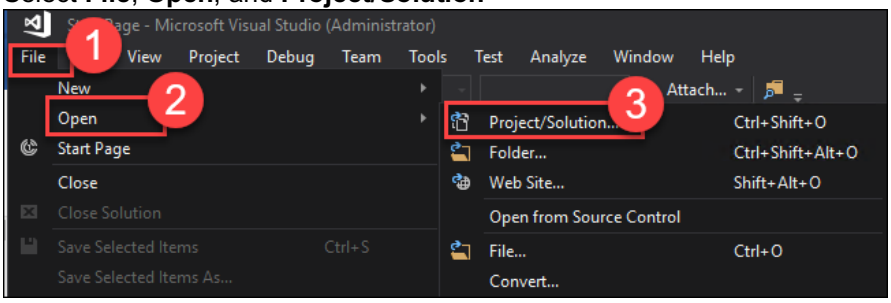


    a. **Name:** PolicyConnect
    b. **Value:** Paste in the updated string you created with the Failover group name from the SQLSERVER.TXT file.

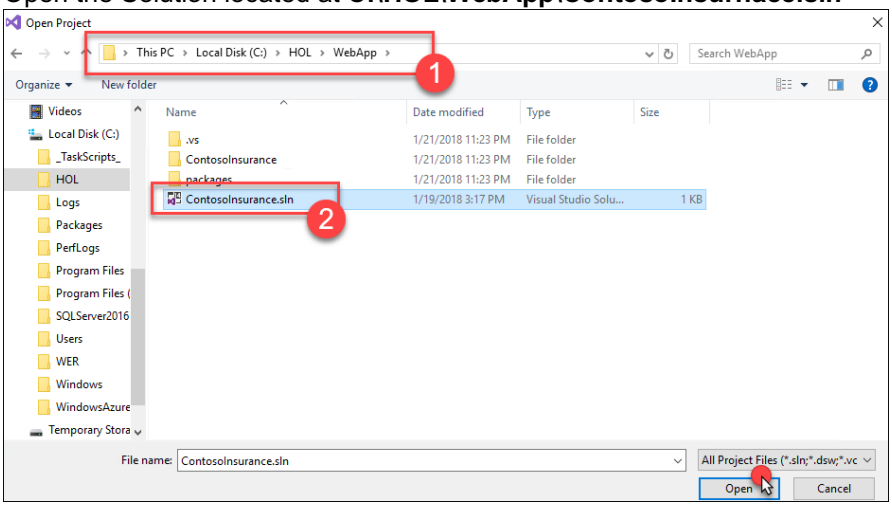
37. On the LABVM open **Visual Studio**. You will be required to login to Visual Studio. If you don't have an account you can create a free account following the prompts.



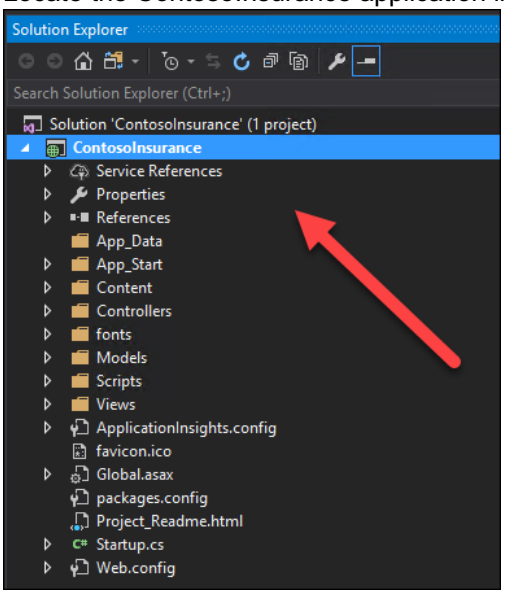
38. Select **File**, **Open**, and **Project/Solution**



39. Open the Solution located at **C:\HOL\WebApp\ContosoInsurnace.sln**

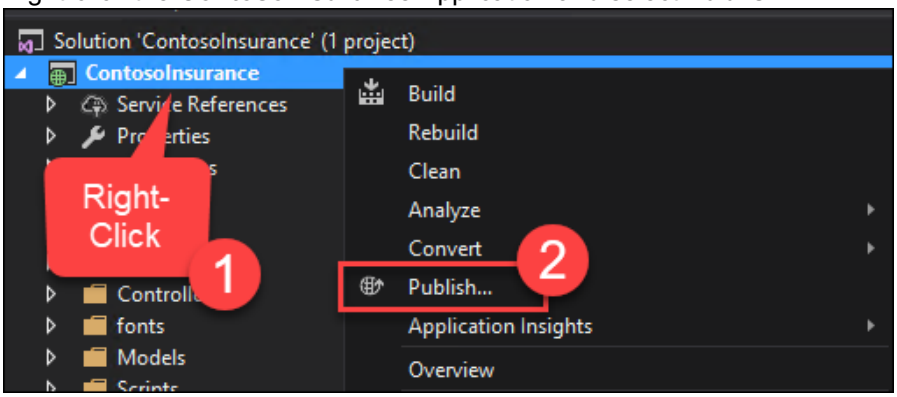


40. Locate the ContosoInsurance application in the Solution Explorer on the right-hand area of Visual Studio

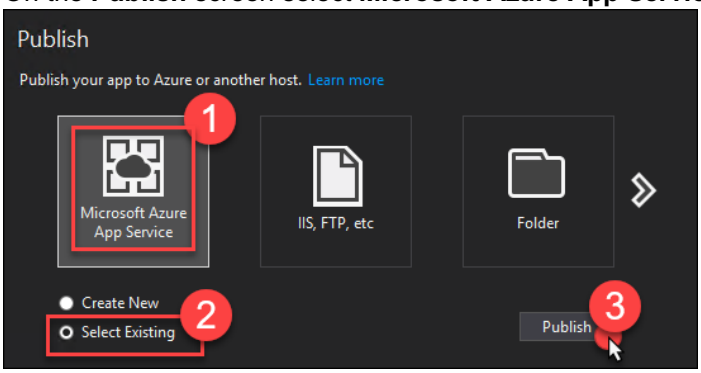


Note: If for some reason the Solution Explorer is not seen you can select **View -> Solution Explorer** on the Menu bar of Visual Studio.

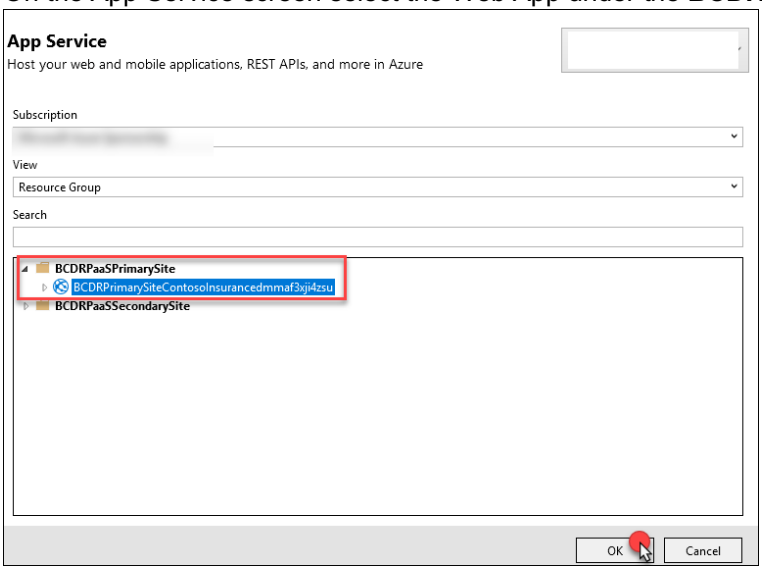
41. Right-click the **ContosoInsurance** Application and select **Publish**



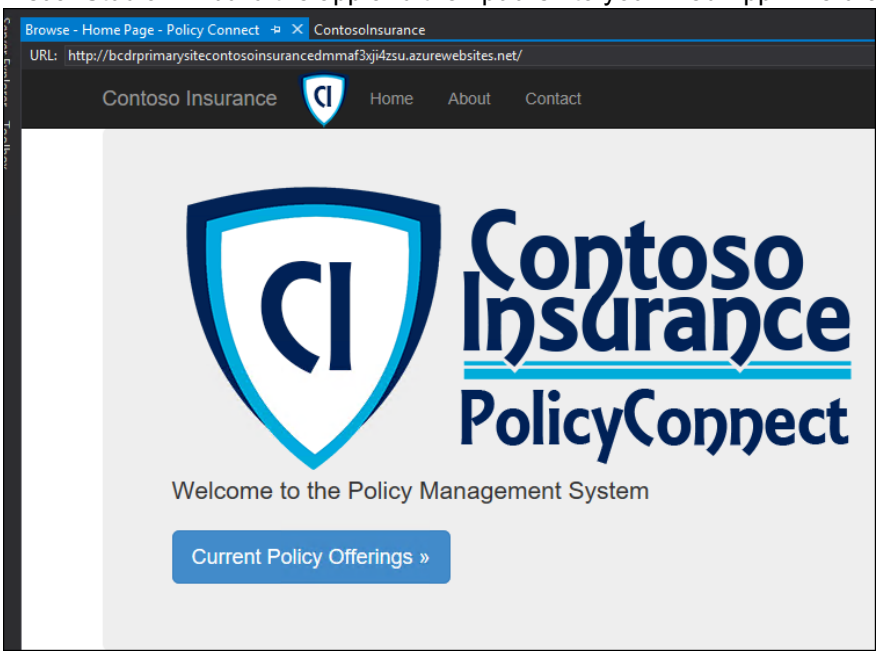
42. On the **Publish** screen select **Microsoft Azure App Service** and then **Select Existing** and finally **Publish**



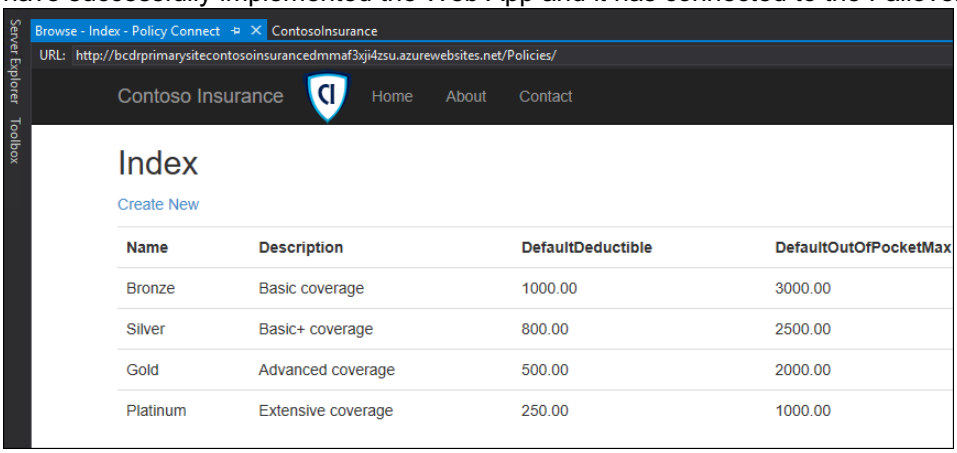
43. On the App Service screen select the Web App under the **BCDRPaaSPrimarySite**. Then select **OK**.



44. Visual Studio will build the app and then publish to your Web App. The browser should open to the application.



45. Select the **Current Policy Offerings** button, and the page should load with data showing. This means that you have successfully implemented the Web App and it has connected to the Failover Group database.

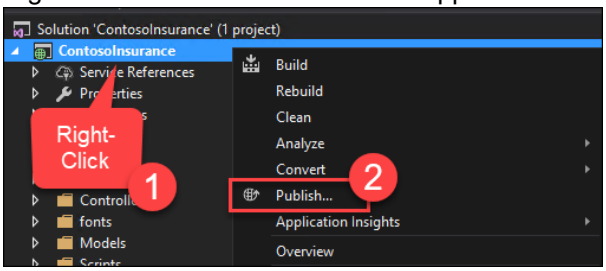


46. Close Visual Studio

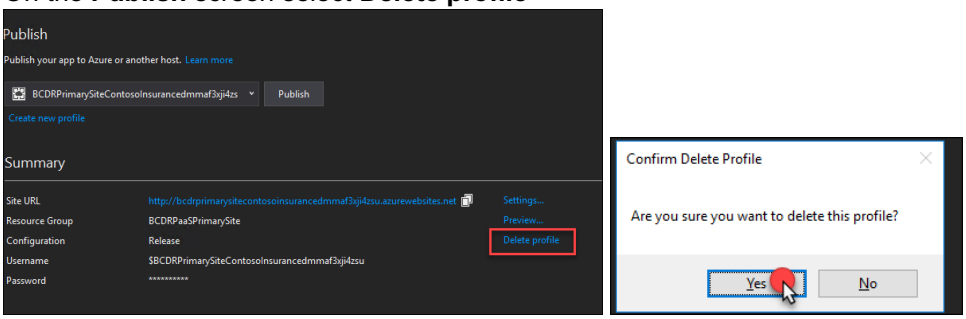
47. Start Visual Studio

48. Re-open the **ContosoInsurance.sln** Solution

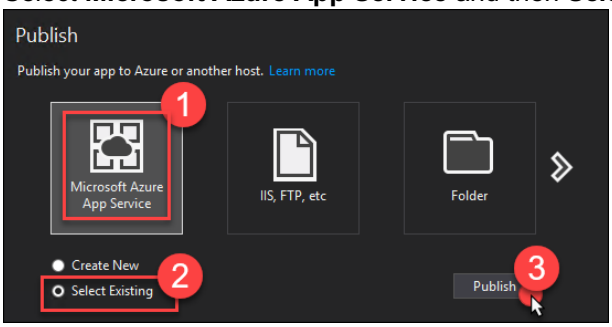
49. Right-click the **ContosoInsurance** Application and select **Publish**



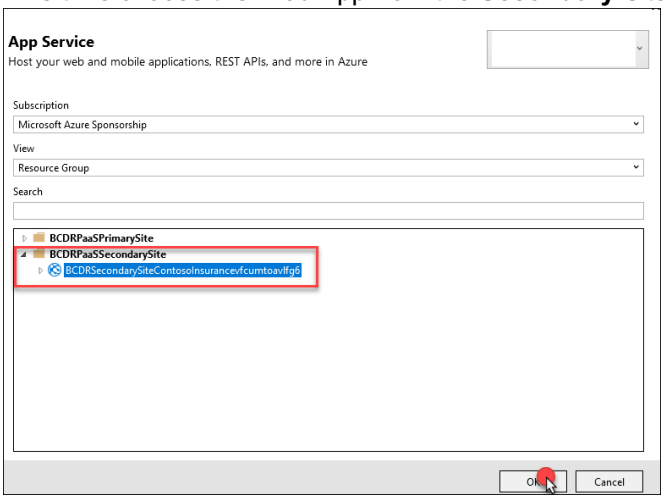
50. On the **Publish** screen select **Delete profile**



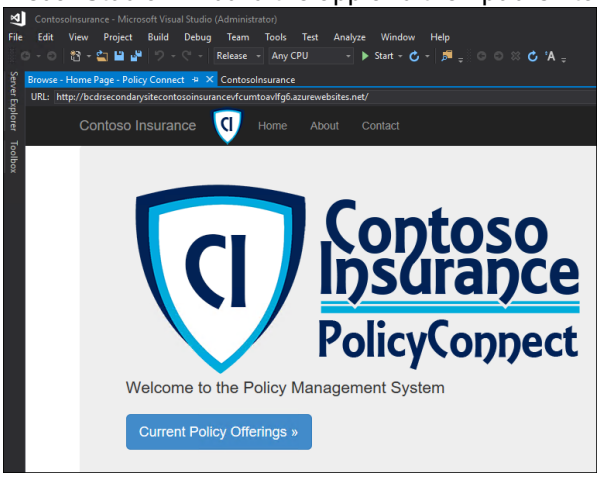
51. Select **Microsoft Azure App Service** and then **Select Existing** and finally **Publish**



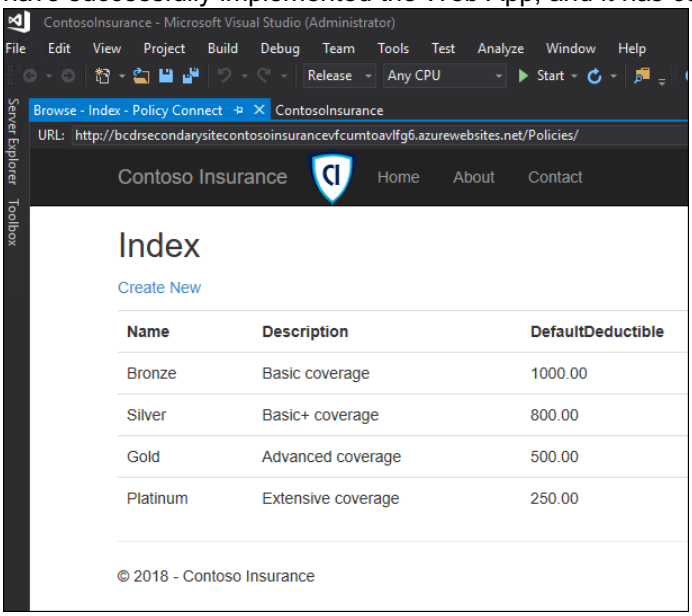
52. This time choose the Web App from the **Secondary** Site running in the **BCDRPaaSSecondarySite**. Select **OK**.



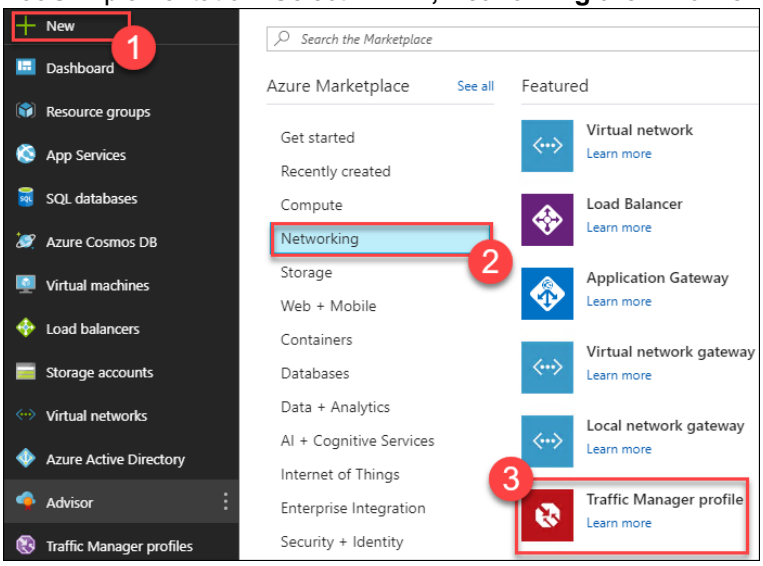
53. Visual Studio will build the app and then publish to your Web App. The browser should open to the application.



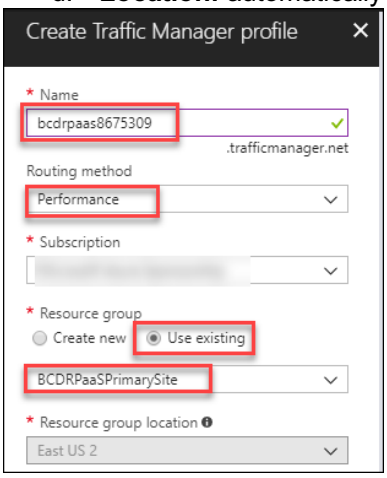
54. Select the **Current Policy Offerings** button, and the page should load with data showing. This means that you have successfully implemented the Web App, and it has connected to the Failover Group database.



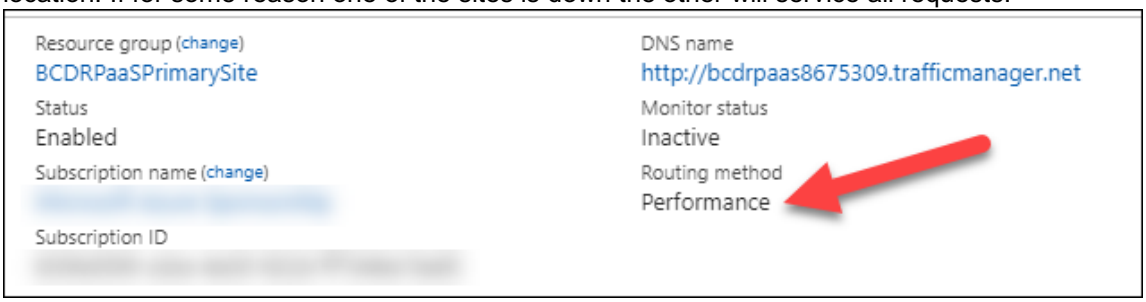
55. Close Visual Studio and move back to the Azure Portal. The next step will be to deploy a Traffic Manager for this PaaS implementation. Select **+NEW, Networking then Traffic Manager profile** in the Azure portal.



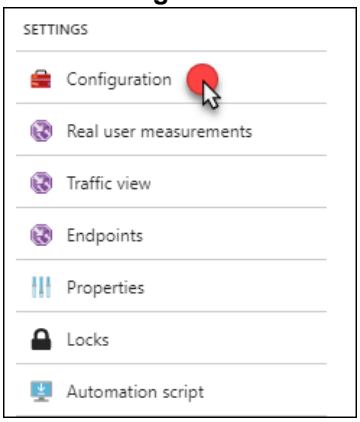
56. Complete the **Create Traffic Manager profile** using the following inputs, then select **Create**:
    a. **Name:** unique name all lowercase using bcdrpaasxxx
    b. **Routing method:** Performance
    c. **Resource group:** Use existing / BCDRPaasPrimarySite
    d. **Location:** automatically assigned based on the BCDRPaaSPrimarySite



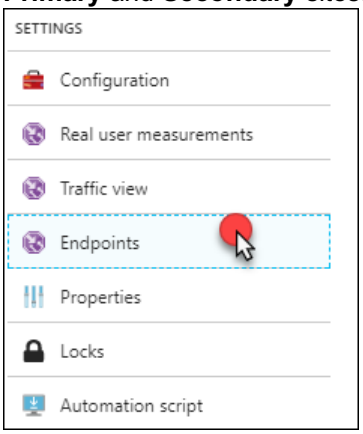
57. Once the Traffic Manager profile is created, open it in the Azure portal. Notice the DNS name. This is the URL that you will use to connect to the application. Once configured this DNS name will always respond and doesn't matter which location is responding or where if the current primary database is located. Since the **Performance** routing method was selected the closest site to the end users will be calculated and they will be sent to that location. If for some reason one of the sites is down the other will service all requests.



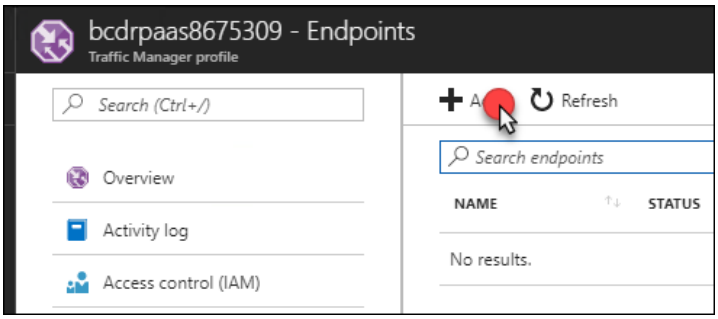
58. Select **Configuration** and review the configurations



59. Next select **Endpoints**. This is where you will configure the two external load balancers that are located your **Primary** and **Secondary** sites.

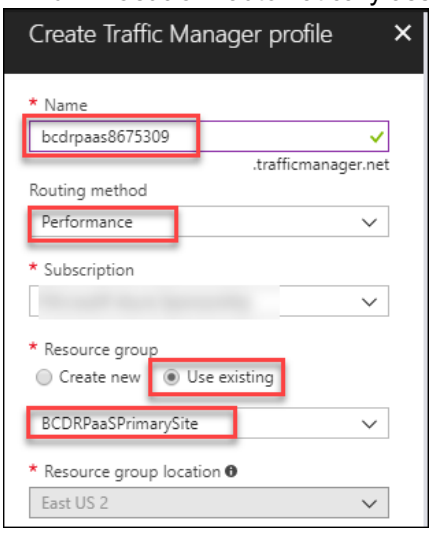


60. Select **+Add**



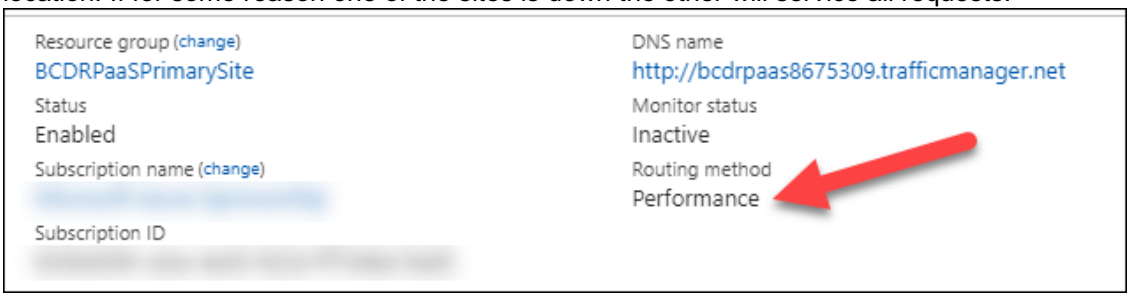
61. Complete the **Add endpoint** using the following inputs and then select **OK**

62. Complete the **Create Traffic Manager profile** using the following inputs, then select **Create**:
    a. **Name:** unique name all lowercase using bcdrpaasxxx
    b. **Routing method:** Performance
    c. **Resource group**: Use existing / BCDRPaasPrimarySite
    d. **Location:** automatically assigned based on the BCDRPaaSPrimarySite

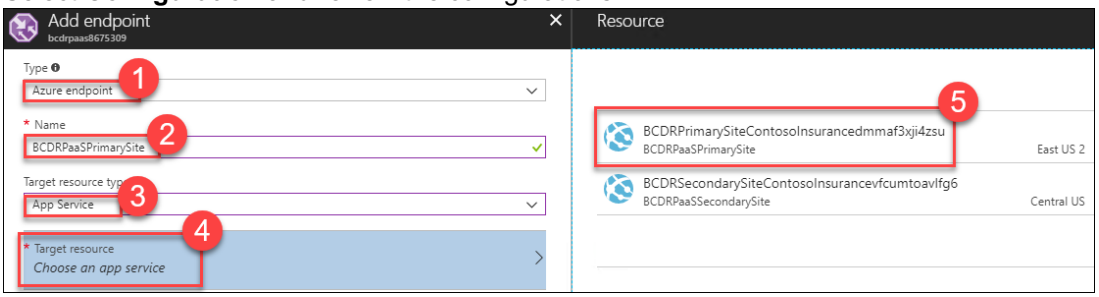


63. Once the Traffic Manager profile is created, open it in the Azure portal. Notice the DNS name. This is the URL that you will use to connect to the application. Once configured, this DNS name will always respond and doesn't matter which location is responding or where if the current primary database is located. Since the **Performance**

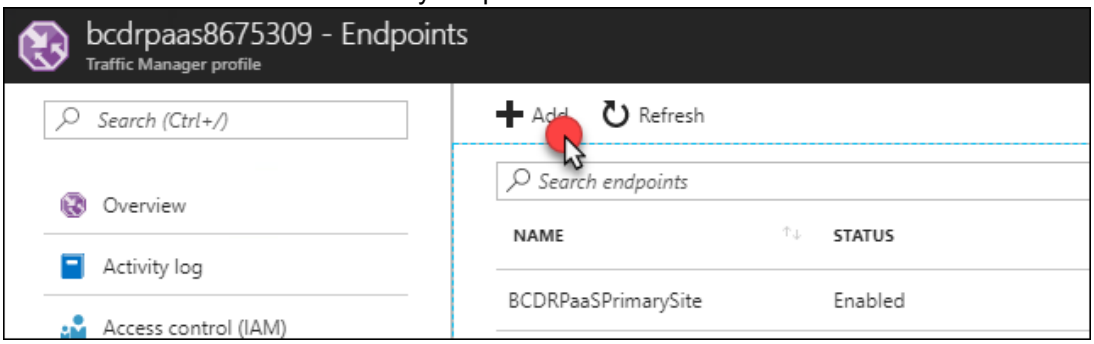
routing method was selected the closest site to the end users will be calculated, and they will be sent to that location. If for some reason one of the sites is down the other will service all requests.



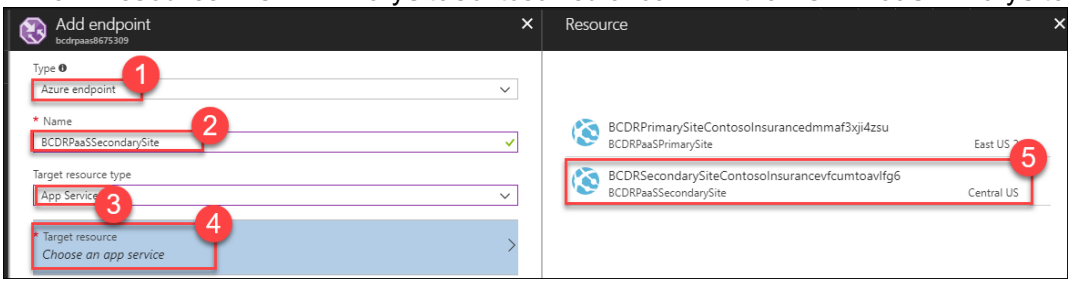
64. Select **Configuration** and review the configurations



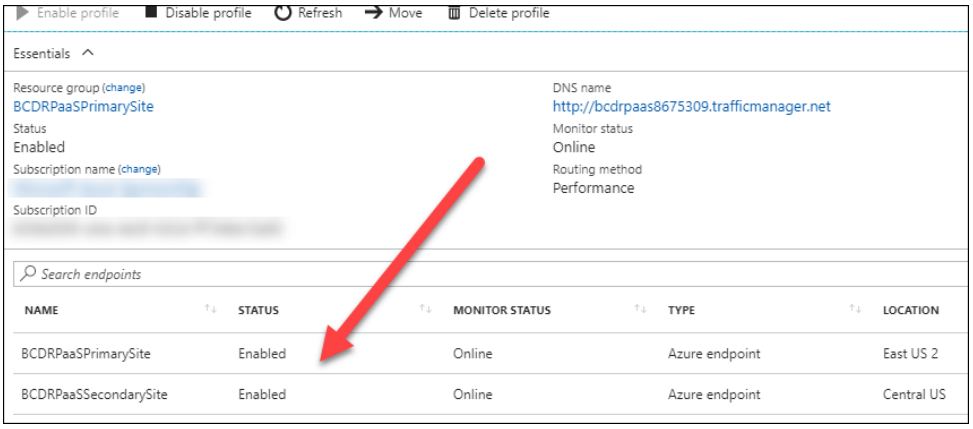
65. Select**+Add**. Notice that the Primary endpoint was created as **"Enabled"**



66. Complete the **Add endpoint** using the following inputs and then select **OK**:
    a. **Type:** Azure endpoint
    b. **Name**: BCDRPaaSSecondarySite
    c. **Target resource type**: App Service
    d. **Target resource**: Choose an app service
    e. **Resource:** BCDRPrimarySiteContosoInsurancexxx in the BCDRPaaSPrimarySite

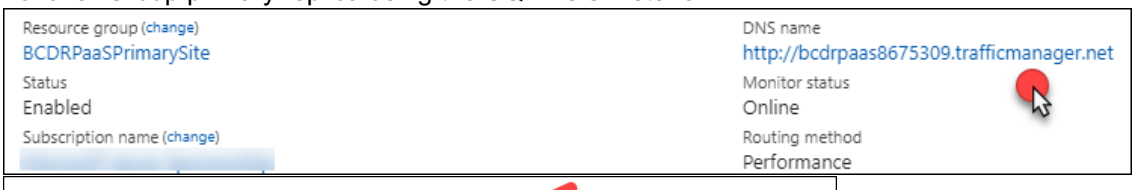


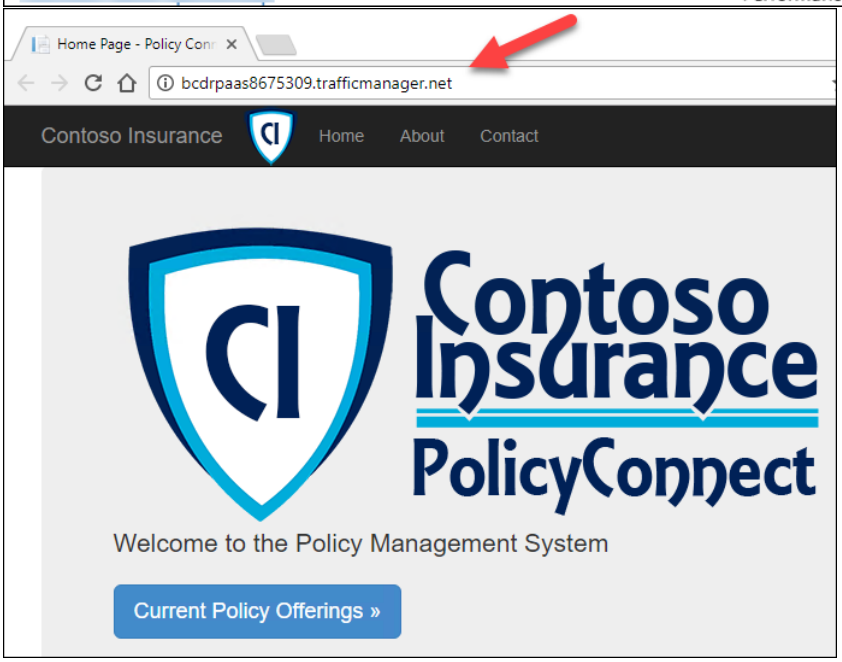
67. Once the second endpoint has been added select **Overview**. The Traffic Manager will monitor the Endpoints and if the **Primary** or **Secondary** site moves to a **Monitor Status** of **Degraded**, then the Traffic Manager will direct traffic only to the other site until the service is restored. The current **Monitor Status** shows that the **Primary** site and the **Secondary** site are **Online**. If there was an outage at the web layer for one of these sites then the site will move to **Degraded,** and the other site will service all requests.



Note: All of this is automatic and easily configured with a vanity domain by adding a C NAME record in DNS to point to the DNS name of the Traffic manager. This would allow for a site like www.contoso.com to resolve to the DNS name of the traffic manager. The users will never know that the site is failed over or failed back as long as one site is up and the database is active in the Failover group.

68. Select the DNS name of the Traffic manager the Policy Connect web application will load. This is connecting to one of the two Web Apps running in the **Primary** Site or **Secondary** Site and talking to the Azure SQL Database Failover Group primary replica using the SQL FOG Listener.



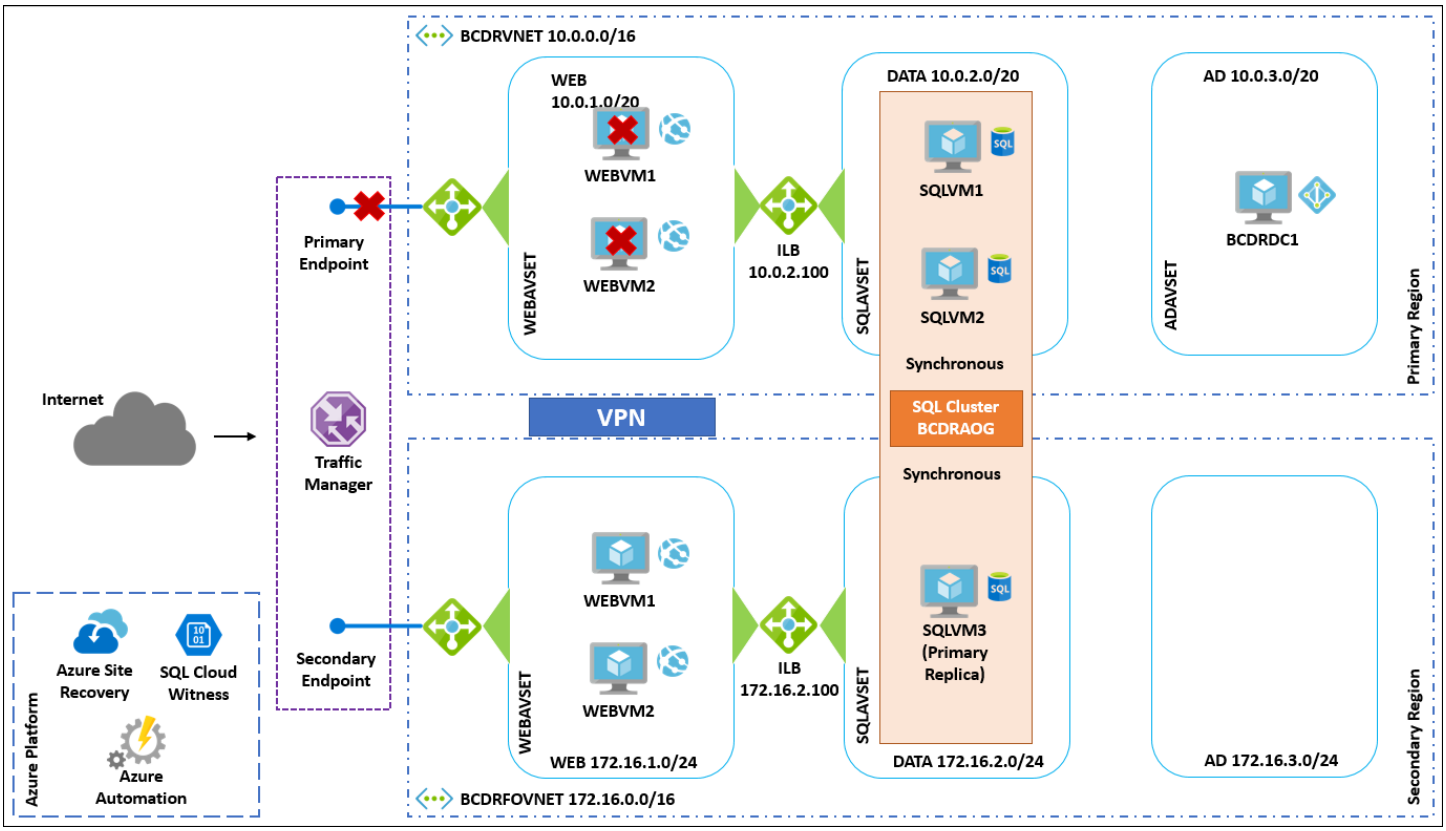


Exercise 4: Simulate failovers

Duration: 75 minutes

Now, that your applications have been made ready for high-availability and BCDR you will now simulate their capabilities. First, you will Failover the **Azure IaaS environment** from your **Primary** to **Secondary** Region. Next, you will migrate the **On-Premises** environment to Azure. The **PaaS** environment will be tested to ensure that failing over the database doesn't cause an outage to the application. Finally, you will failback the Azure IaaS environment from the **Secondary** site to the **Primary** site.

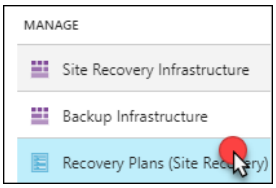
Task 1: Failover Azure IaaS region to region



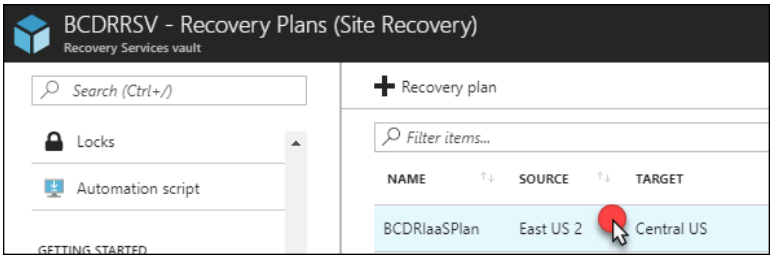
1. Using the Azure portal, open the **BCDRIaaSPrimarySite** resource group. Locate the Traffic Manager DNS URL and select it to ensure that the application is up and running form the Primary Site. Pin the Traffic manager to your dashboard for easy access to this URL or make a favorite in your browser.

2. From the Azure portal, open the **BCDRRSV** Recovery Services Vault located in the **BCDRAzureSiteRecovery** resource group

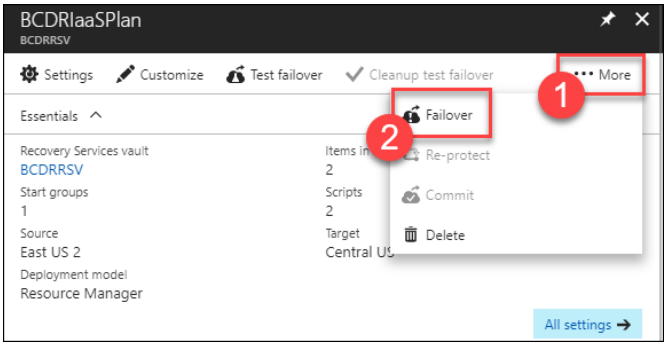
3. Select **Recovery Plans (Site Recovery)** in the **Manage** area



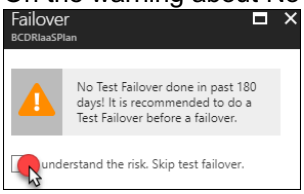
4. Select **BCDRIaaSPlan**



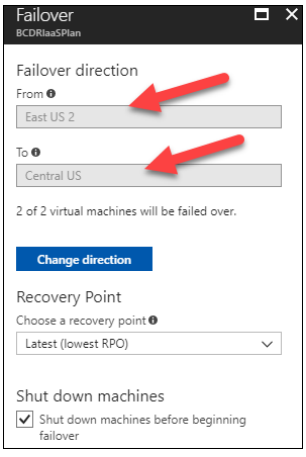
5. Select **More** and then **Failover**



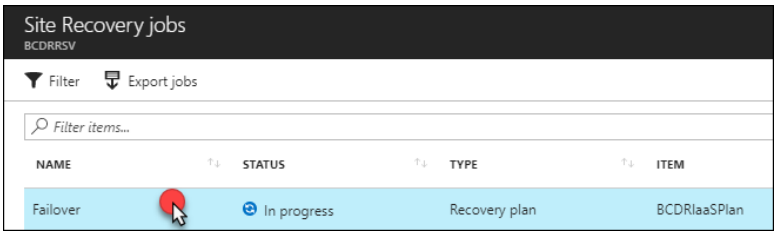
6. On the warning about No Test Failover, select **I understand the risk, Skip test failover**



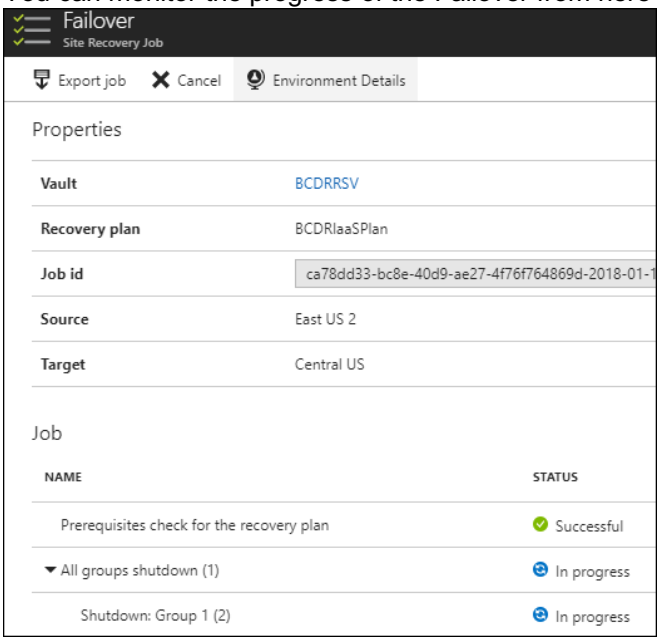
7. Review the Failover direction. Notice that **From** is the **Primary** site, and **To** is the **Secondary** site. Select **OK**.



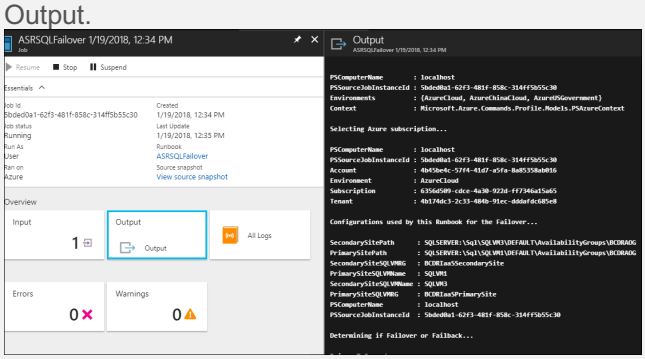
8. After the Failover is initiated close the blade and move to **Jobs**, then **Site Recovery Jobs**. Select the **Failover** job to monitor the progress.



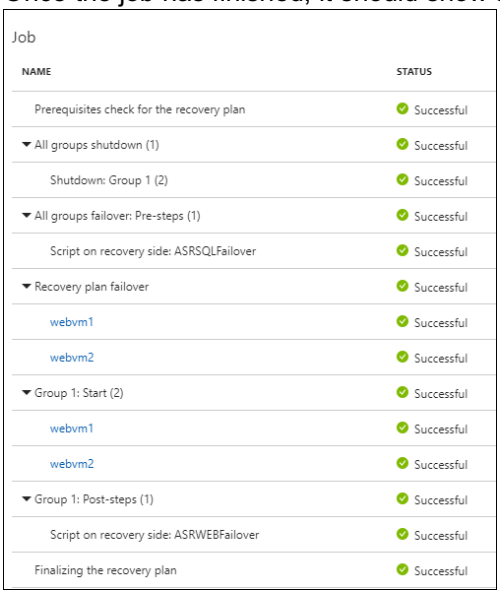
9. You can monitor the progress of the Failover from here



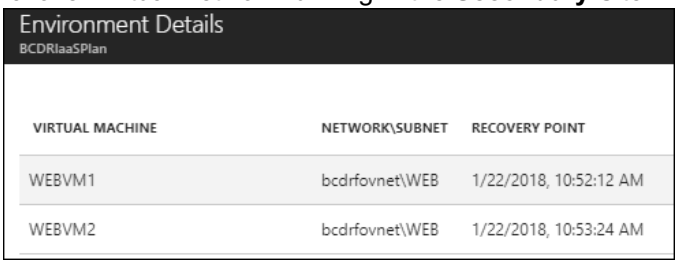
Note: Do not make any changes to your VMs in the Azure portal during this process. Allow ASR to take the actions and wait for the failover notification prior to moving on to the next step. You can open another portal view in a new browser tab and review the output of the Azure Automation Jobs, by opening the jobs and selecting Output.



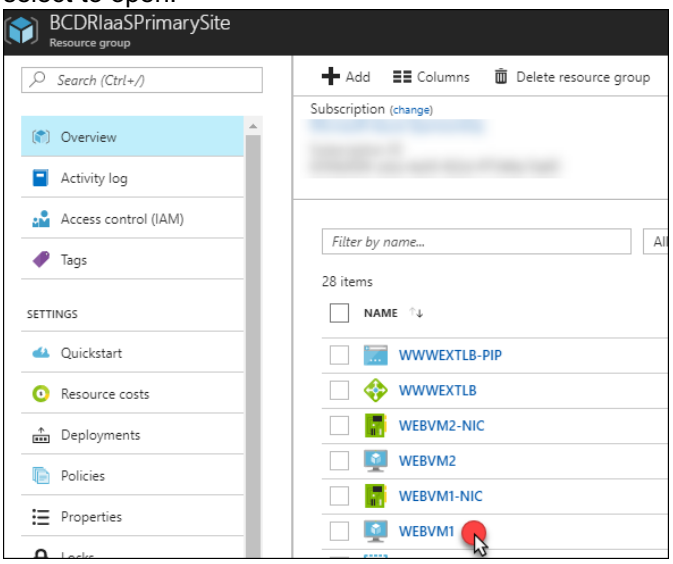
10. Once the job has finished, it should show as successful for all tasks



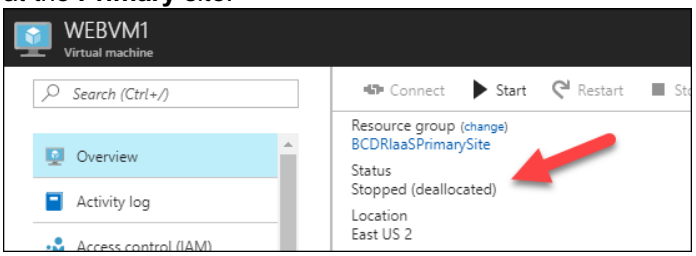
11. There is also details of the failover VMs shown. Notice how they are running the **BCDRFOVNET**. This is the failover Virtual Network running in the **Secondary** Site.



12. Select **Resource groups** and select **BCDRIaaSPrimarySite**. Locate the **WEBVM1** in the resource group and select to open.

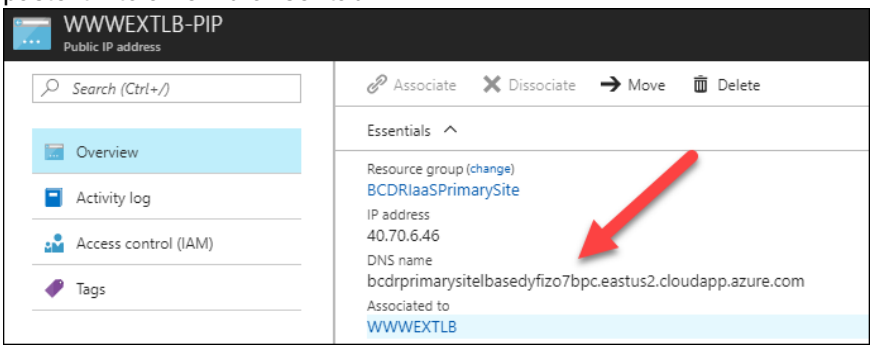


13. Notice that it currently shows as **Status: Stopped (deallocated).** This shows that failover has stopped the VMs at the **Primary** site.



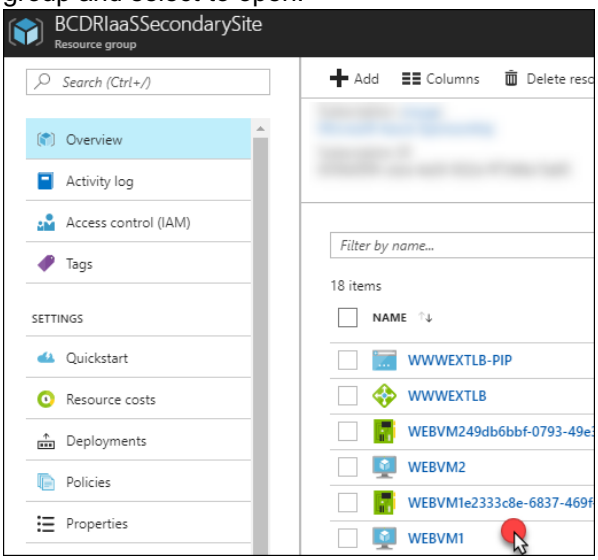
Note: Do not select Start! This is a task only for ASR.

14. Move back to the Resource group and select the **WWWEXTLB-PIP** Public IP address. Copy the DNS name and paste it into a new browser tab.

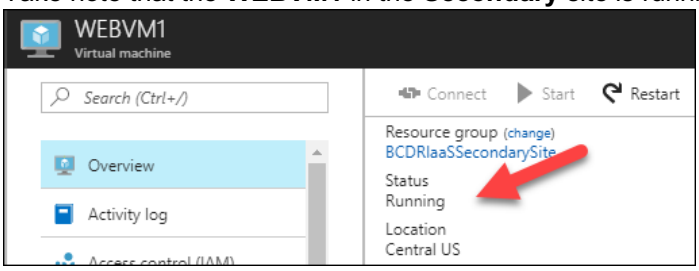


15. The site will be unreachable at the Primary location

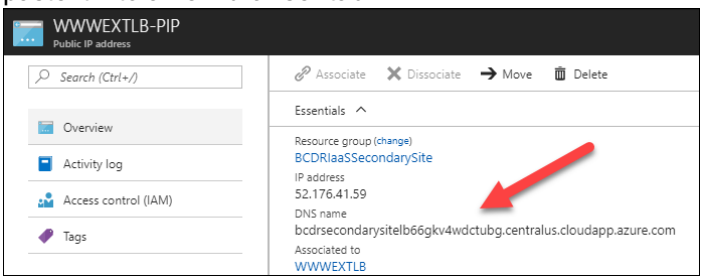
16. In the Azure portal, move to the **BCDRIaaSSecondarySite** resource group. Locate the **WEBVM1** in the resource group and select to open.



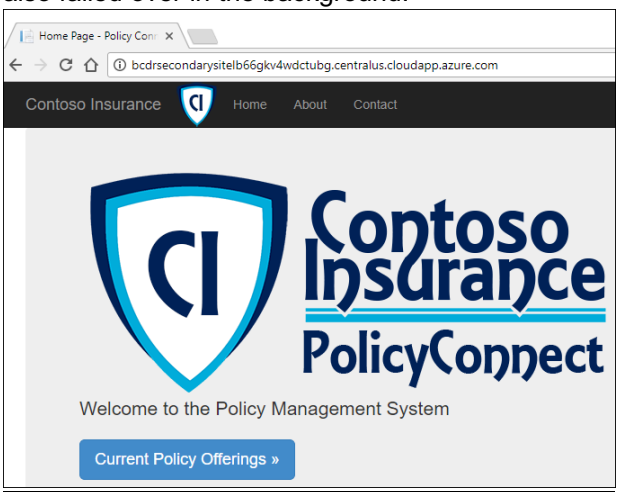
17. Take note that the **WEBVM1** in the **Secondary** site is running

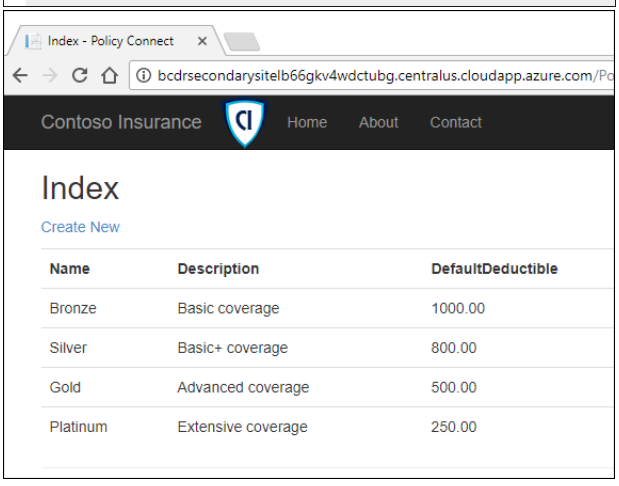


18. Move back to the Resource group and select the **WWWEXTLB-PIP** Public IP address. Copy the DNS name and paste it into a new browser tab.

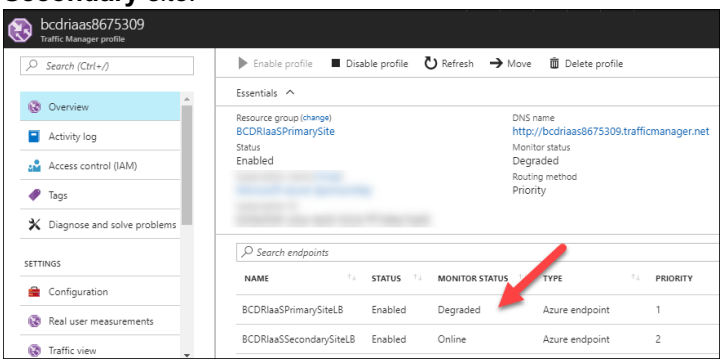


19. The Application running on the WEBVM1 and WEBVM2 is now responding from the **Secondary** Site. Make sure to select the current Policy Offerings to ensure that there is connectivity to the SQL Always On group that was also failed over in the background.

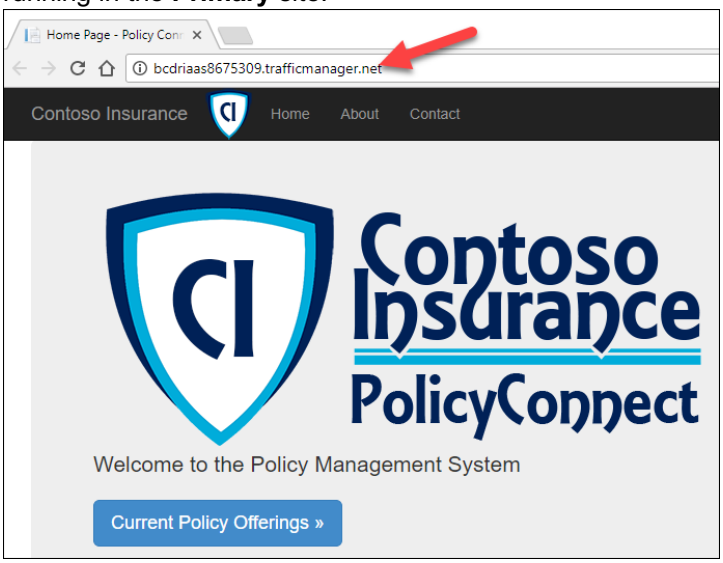




20. Using the Azure portal to locate the **BCDRIaaS Traffic Manager** profile in the **BCDRIaaSPrimarySite** resource group. Notice that the Monitor Status has moved to **Degraded** for the **Primary** Site and moved to **Online** for the **Secondary** site.

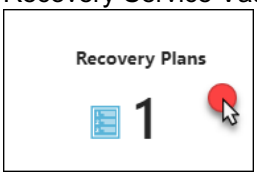


21. Select **DNS Name URL**. The site load immediately and is failed over. The users will always be using this DNS URL, so they there is no change in how they access the site even though it is failed over. There **will** be downtime as the failover happens, but once the site is back online the experience for them will be no different than when it is running in the **Primary** site.

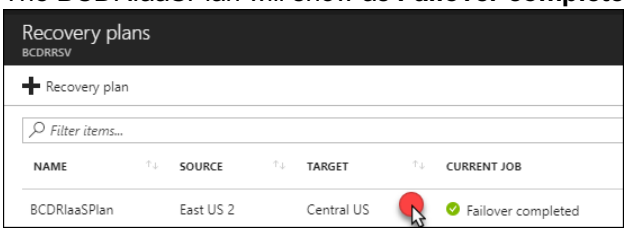


Option task: If you wish you can RDP to **SQLVM3** and open the SQL Management Studio to review the Failed over **BCDRAOG**. You will see that **SQLVM3** which is running the **Secondary** site is now the Primary Replica.

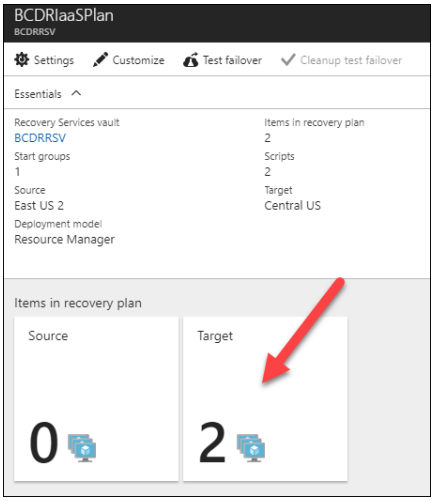
22. Now, that you have successfully failed over you need to prep ASR for the Failback. Move back to the **BCDRSRV** Recovery Service Vault using the Azure portal. Select Recovery Plans on the ASR dashboard.



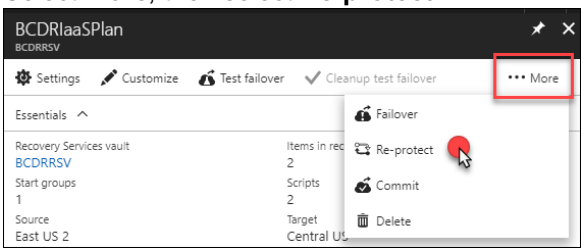
23. The BCDRIaaSPlan will show as **Failover completed.** Select the Plan



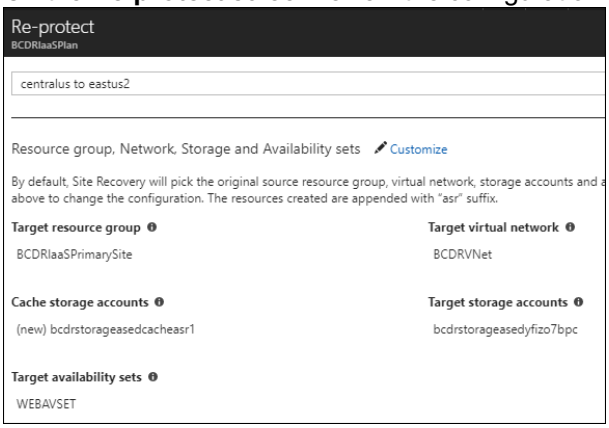
24. Notice that now 2 VMs are show in the **Target**



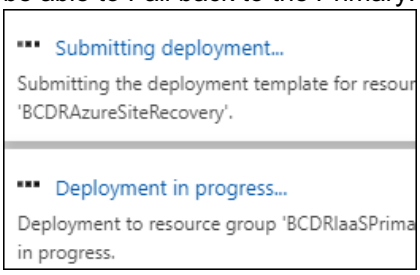
25. Select **More**, then select **Re-protect**



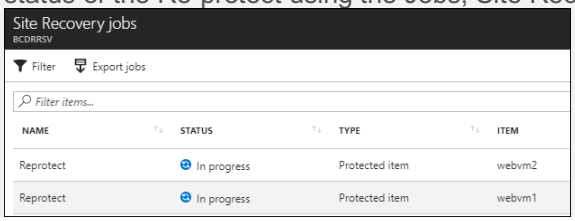
26. On the **Re-protect** screen review the configuration and then select **OK**



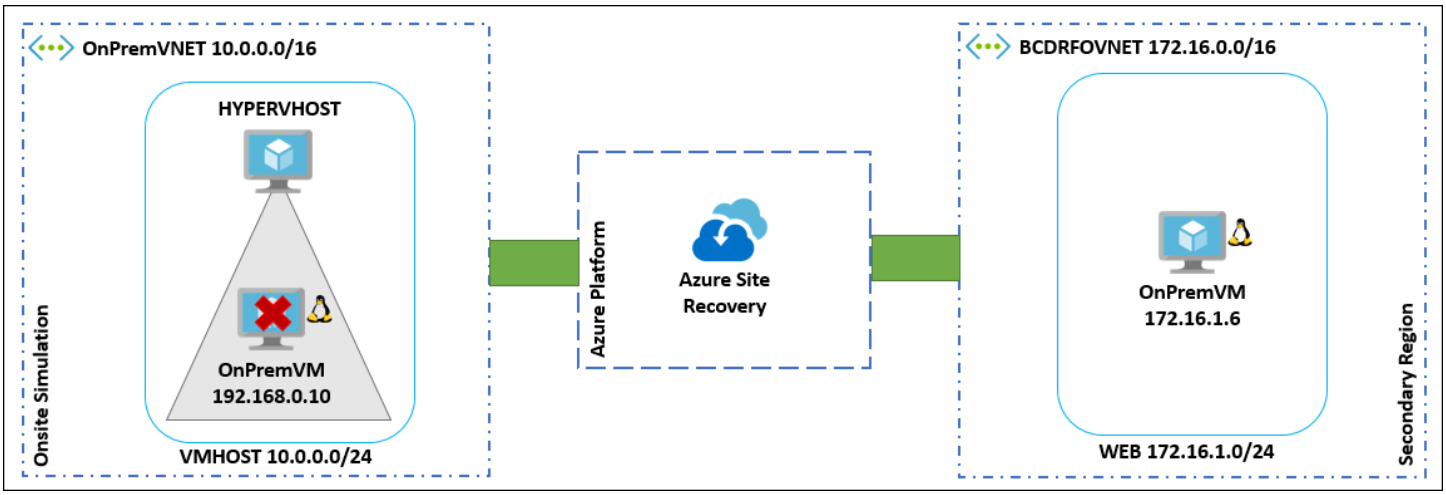
27. The portal will submit a deployment. This process will take some time, by first committing the failover and then synchronizing the WEBVM1 and WEBVM2 back to the Primary Site. Once this process is complete, then you will be able to Fail back to the Primary.



Note: You will perform the Failback later in the HOL, so it is safe to move on to the next task. You can check the status of the Re-protect using the Jobs, Site Recovery Jobs area of the BCDRSRV.

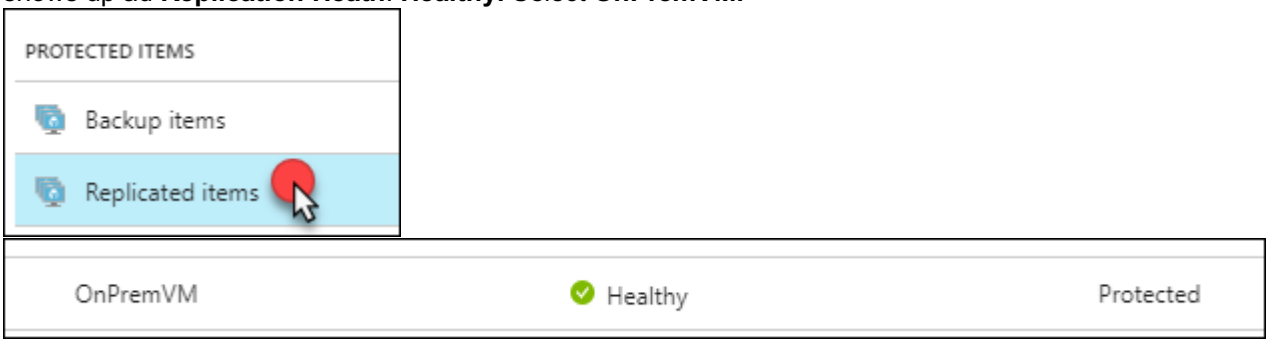


Task 2: Migrate the on-premises VM to Azure IaaS

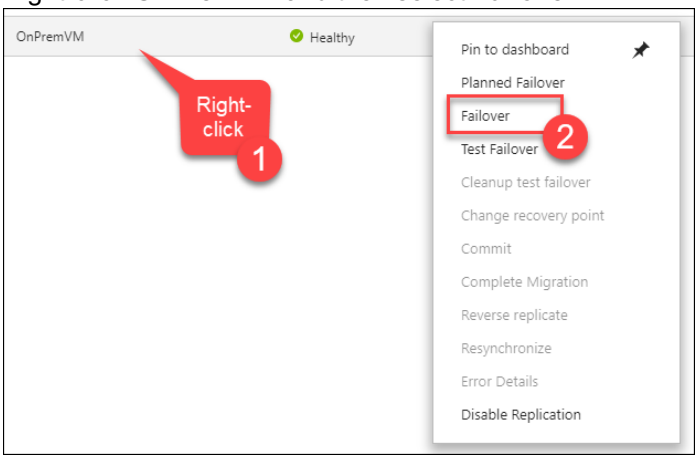


1. From the Azure portal, open the **BCDRRSV** Recovery Services Vault located in the **BCDRAzureSiteRecovery** resource group

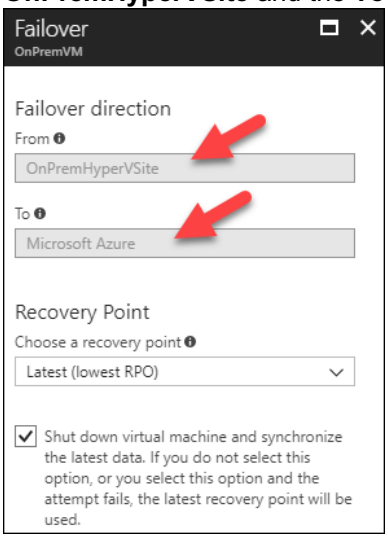
2. Open the **BCDRSRV** and select **Replicated Items** under the **Protected Items** area. Make sure that **OnPremVM** shows up ad **Replication Heath**: **Healthy.** Select **OnPremVM.**



3. Right-click **OnPremVM** and then select **Failover**

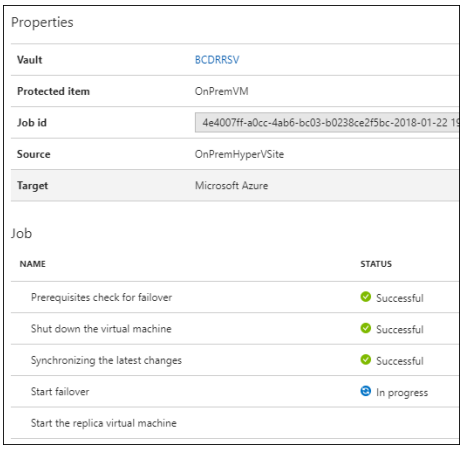


4. Select you understand the risk without a Test Failover and then on the Failover blade notice that the **From** is **OnPremHyperVSite** and the **To** is **Microsoft Azure**. Select **OK**.

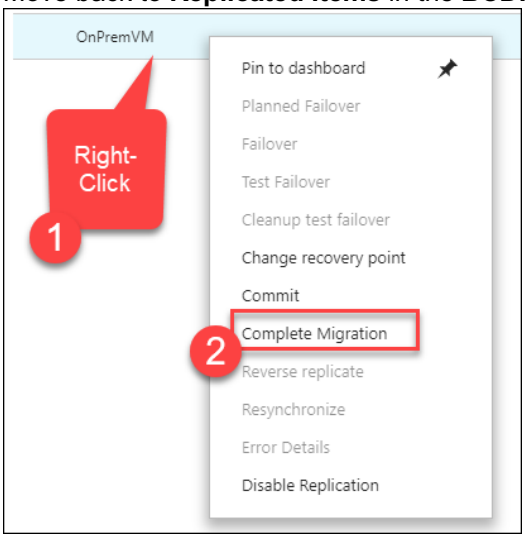


5. The Azure portal will provide a notification that the Failover is starting

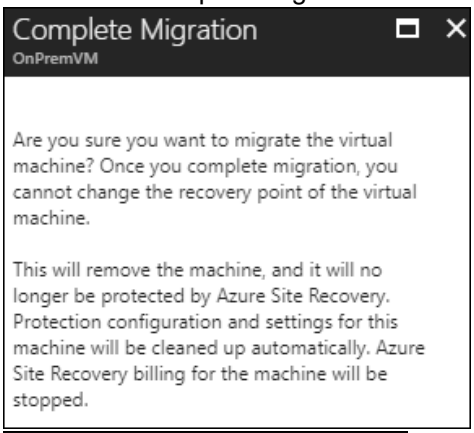
6. Using Jobs, Site Recovery Jobs, you can watch the progress of the failover

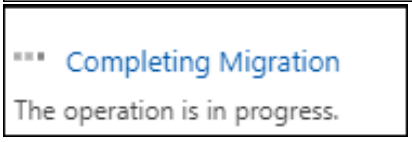


7. Move back to **Replicated Items** in the **BCDRSRV** and right-click **OnPremVM**. Select **Complete Migration**.

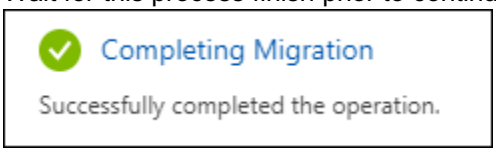


8. Review the Complete Migration blade and then select **OK**

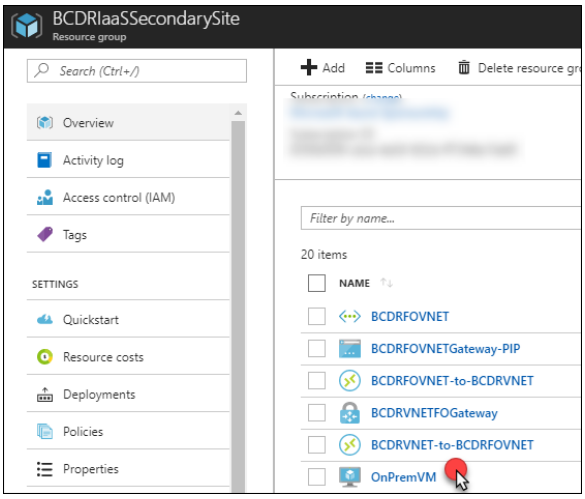




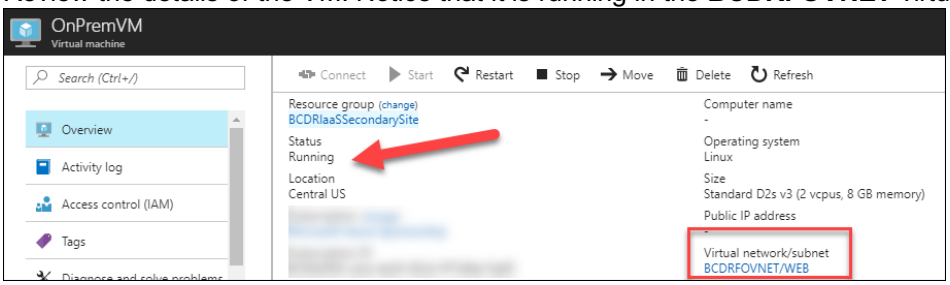
9. Wait for this process finish prior to continuing



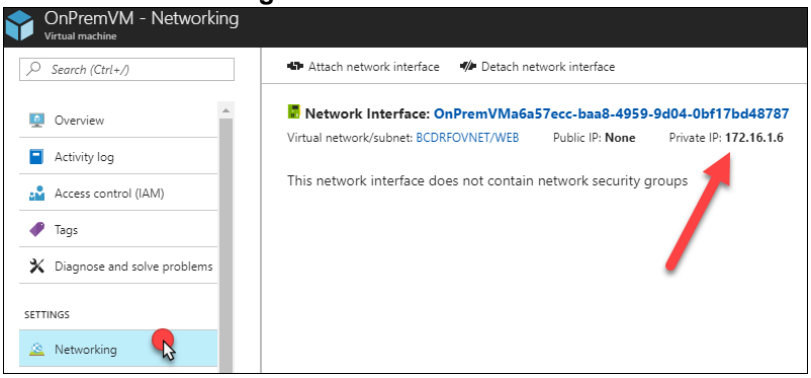
10. Once the Migration is completed, you can move over to the **BCDRIaaSecondarySite** Resource group and locate and select the **OnPremVM**.



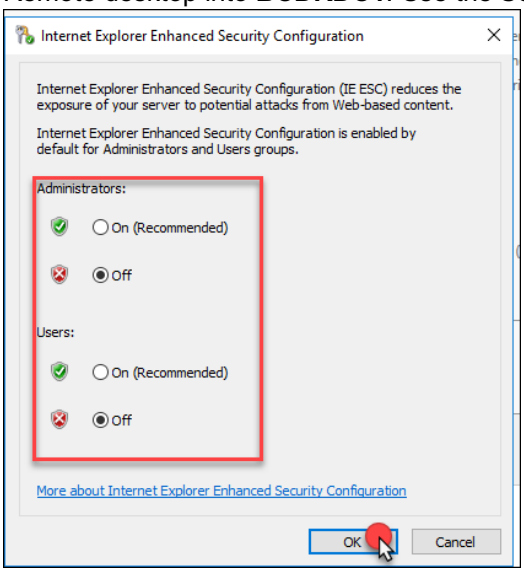
11. Review the details of the VM. Notice that it is running in the **BCDRFOVNET** virtual network in the **WEB** Subnet



12. Select the **Networking** link. Notice the IP address of the VM

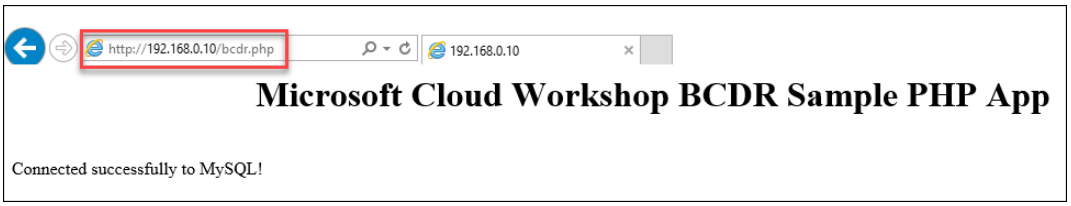


13. Remote desktop into **BCDRDC1**. Use the Server Manager to disable the **IE Enhanced Security Configuration**.



14. Point the browser of BCDRDC1 at the IP address of the **OnPremVM**. And it should load the sample web application

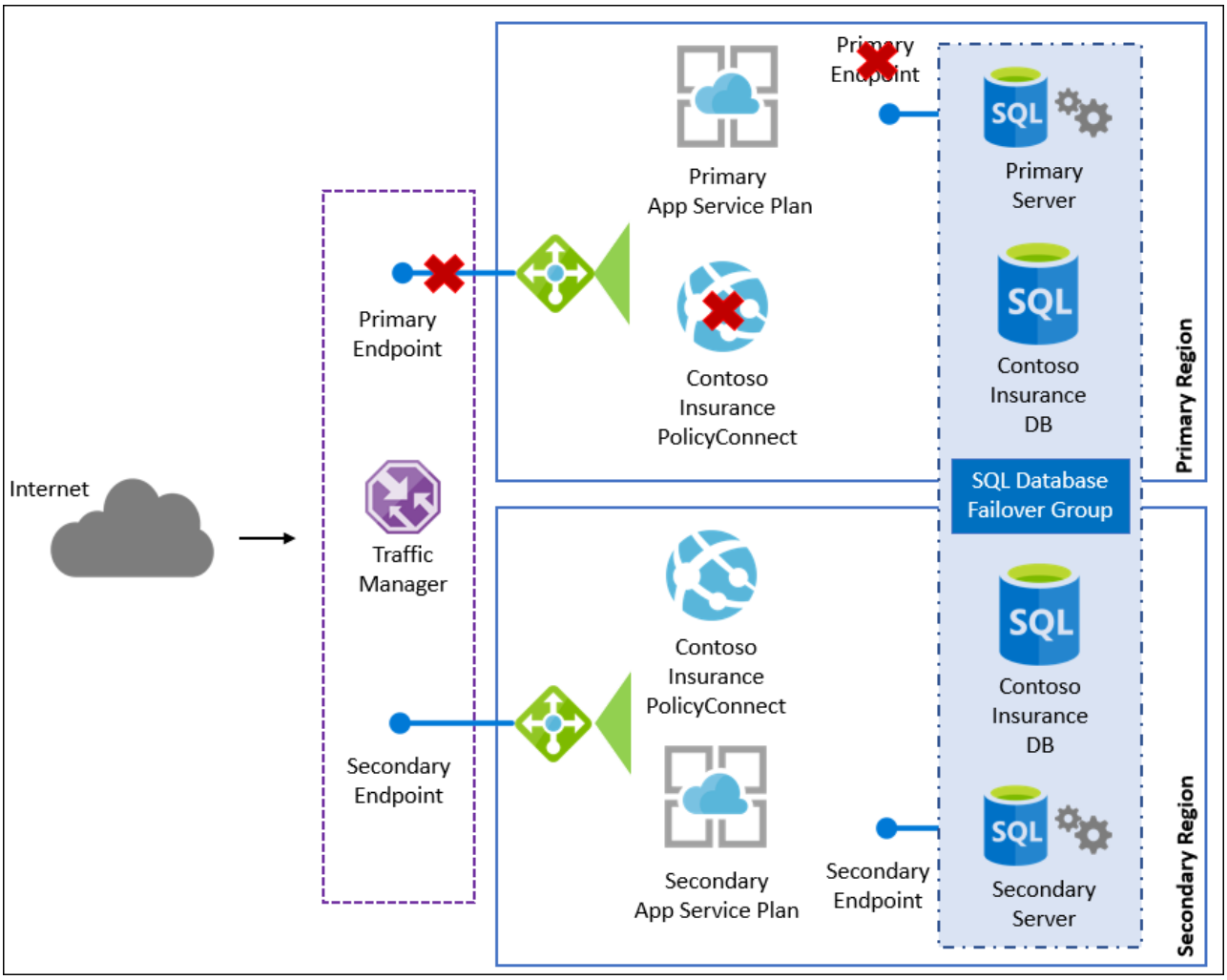
```
http://172.16.1.?/bcdr.php
```



15. Your OnPremVM has been successfully migrated to Azure

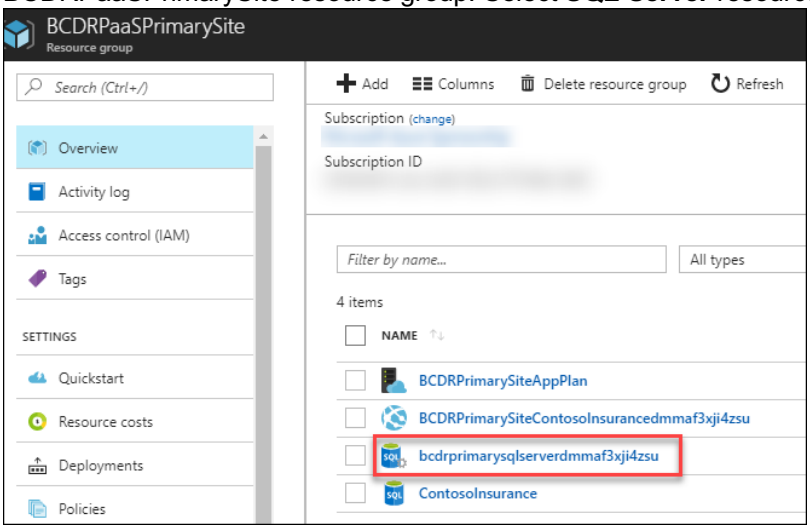
Option Task: If you wish you can Remote Desktop back to the HYPERVHOST, and you will see that the migrated VM has shutdown.

Task 3: Failover and failback Azure PaaS

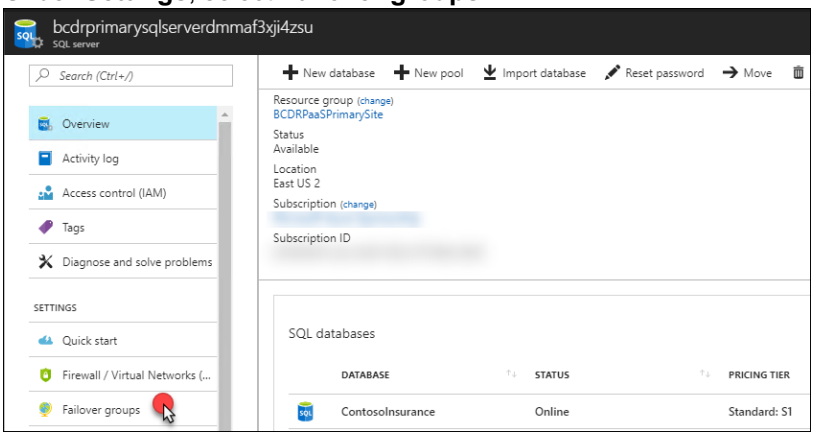


1. Using the Azure portal, open the **BCDRPaaSPrimarySite** resource group. Locate the Traffic Manager profile and then click the URL to ensure that the application is running.

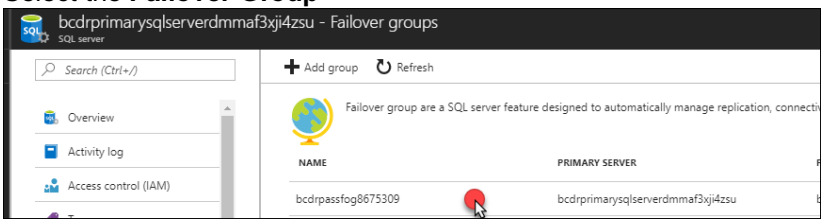
2. Once you are sure that the website is active and connecting to the database, move back to the BCDRPaaSPrimarySite resource group. Select **SQL Server** resource.



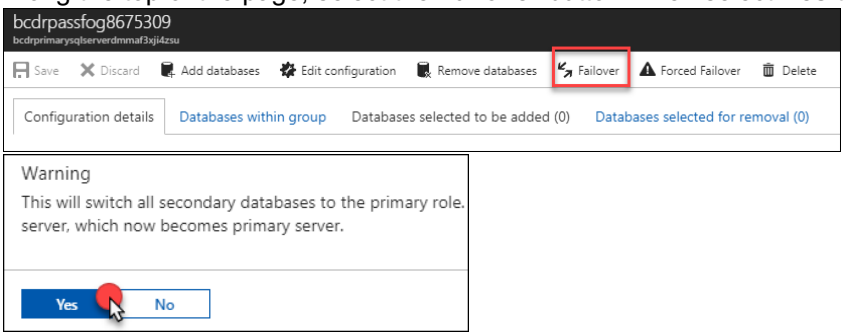
3. Under **Settings**, select **Failover groups**



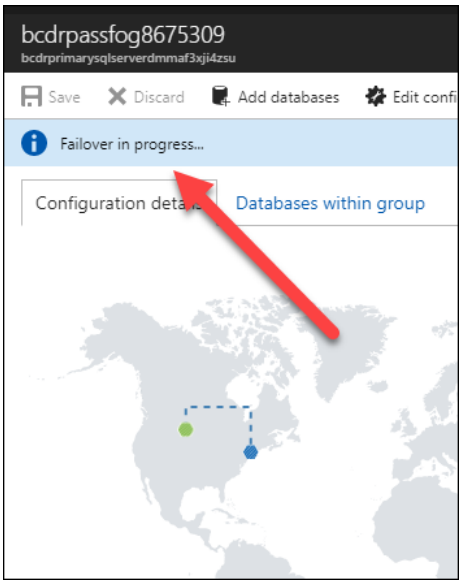
4. Select the **Failover Group**



5. Along the top of the page, select the **Failover** button. Then select **Yes** to confirm.

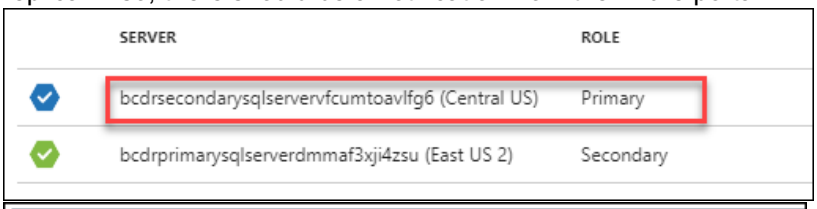


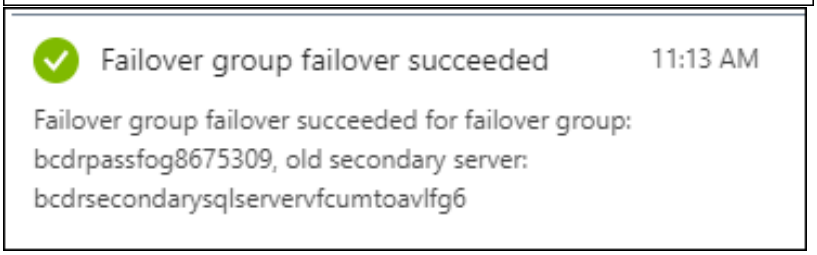
6. The portal will notify you of the **Failover in progress**



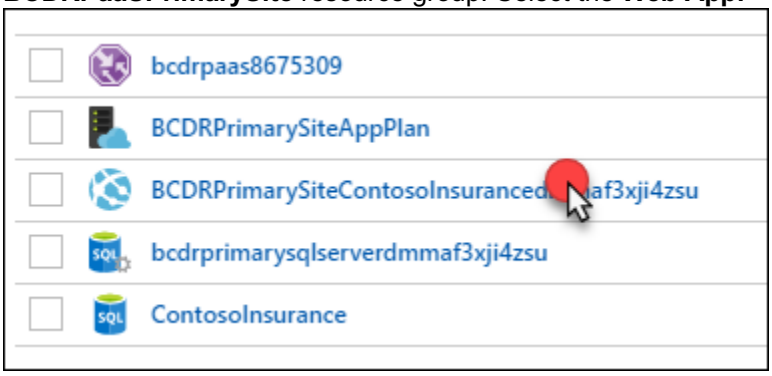
7. While, this is happening move back to your browser tab with the Contoso Insurance application is running on using the PaaS Traffic Manager link. Attempt to use the application. You should see no difference during the failover, but there could be some slowdown in the responses from the web pages that access the database.

8. After a few minutes, move back to the Azure portal to the page where you performed the Failover. You should see that the Failover has completed and the Server running in the Secondary site will now show as the Primary replica. Also, there should be a notification from the Azure portal.

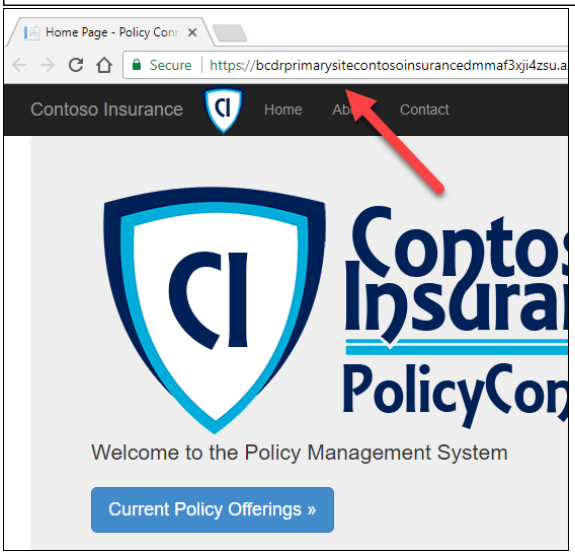




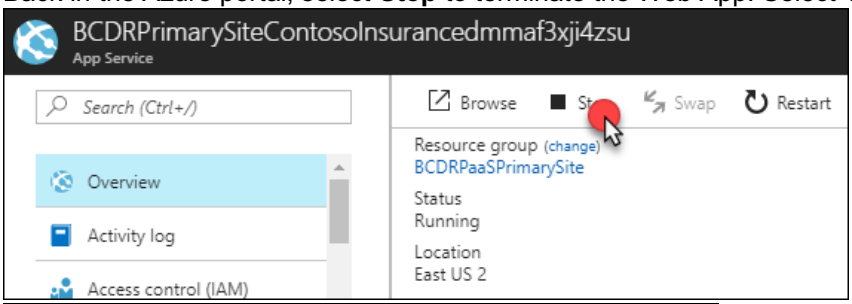
9. Next, to simulate a failover of one of the Azure regions you will stop the Primary Web App. Move to the **BCDRPaaSPrimarySite** resource group. Select the **Web App.**



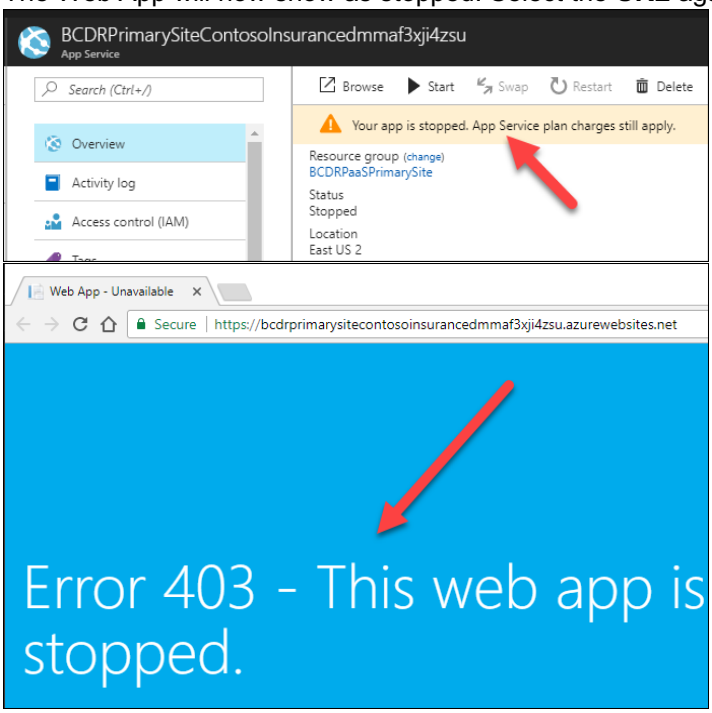
10. Select the URL of the **Web App**. The browser should open to the direct link to the Application.



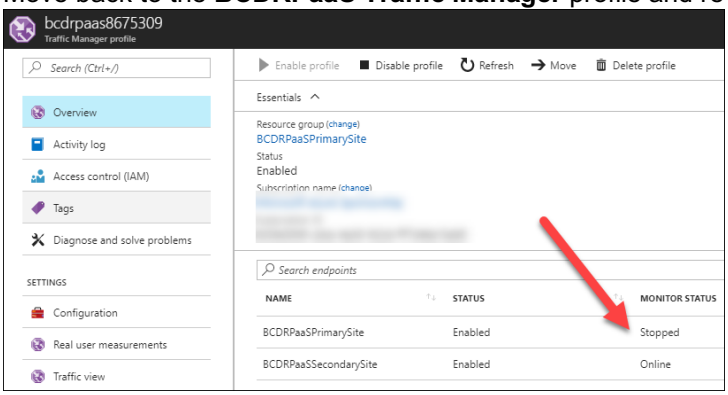
11. Back in the Azure portal, select **Stop** to terminate the Web App. Select **Yes** to confirm.



12. The Web App will now show as stopped. Select the **URL** again, and notice that the Web App shows as stopped.

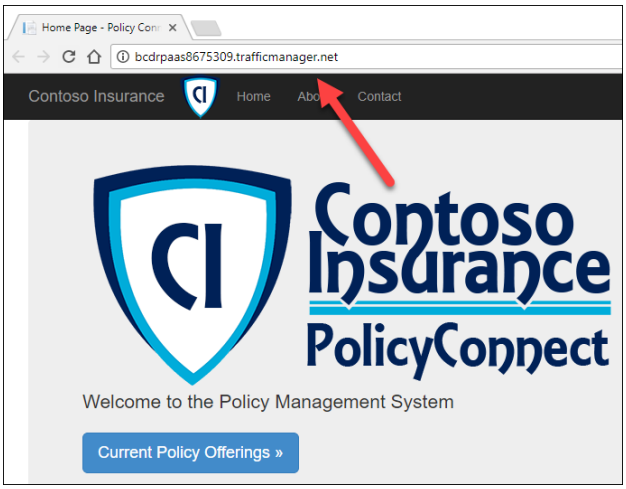


13. Move back to the **BCDRPaaS Traffic Manager** profile and review the **Monitor Status**



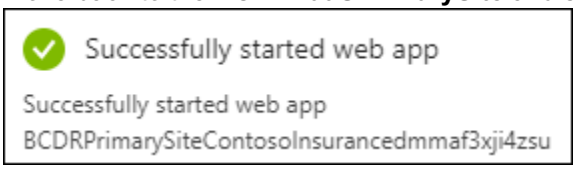
14. Select the **DNS Name** URL

15. Notice that the website loads as normal. You can select refresh or F5, and you will always get back to the site.

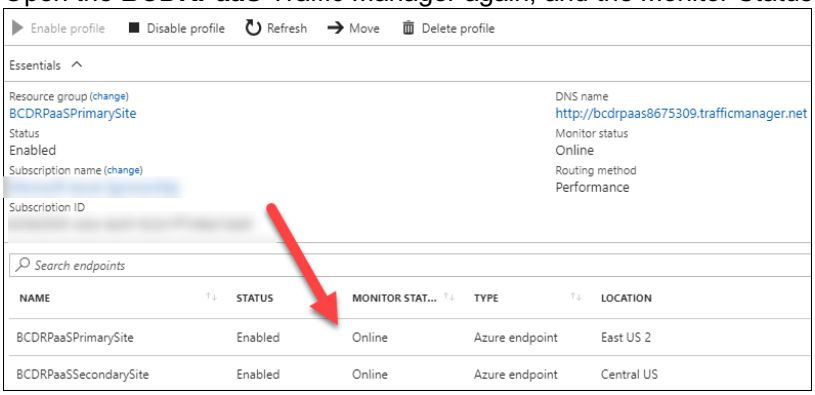


16. In the current configuration, you are completely failed over to the **Secondary** site. There were no configurations for you to do and this was completely transparent to the user of the application.

17. Move back to the **BCDRPaaSPrimarySite** and start the Web App

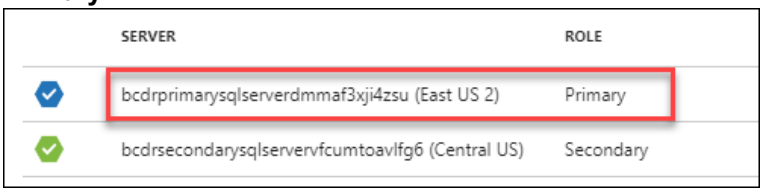


18. Open the **BCDRPaaS** Traffic Manager again, and the Monitor Status should show both as Online again



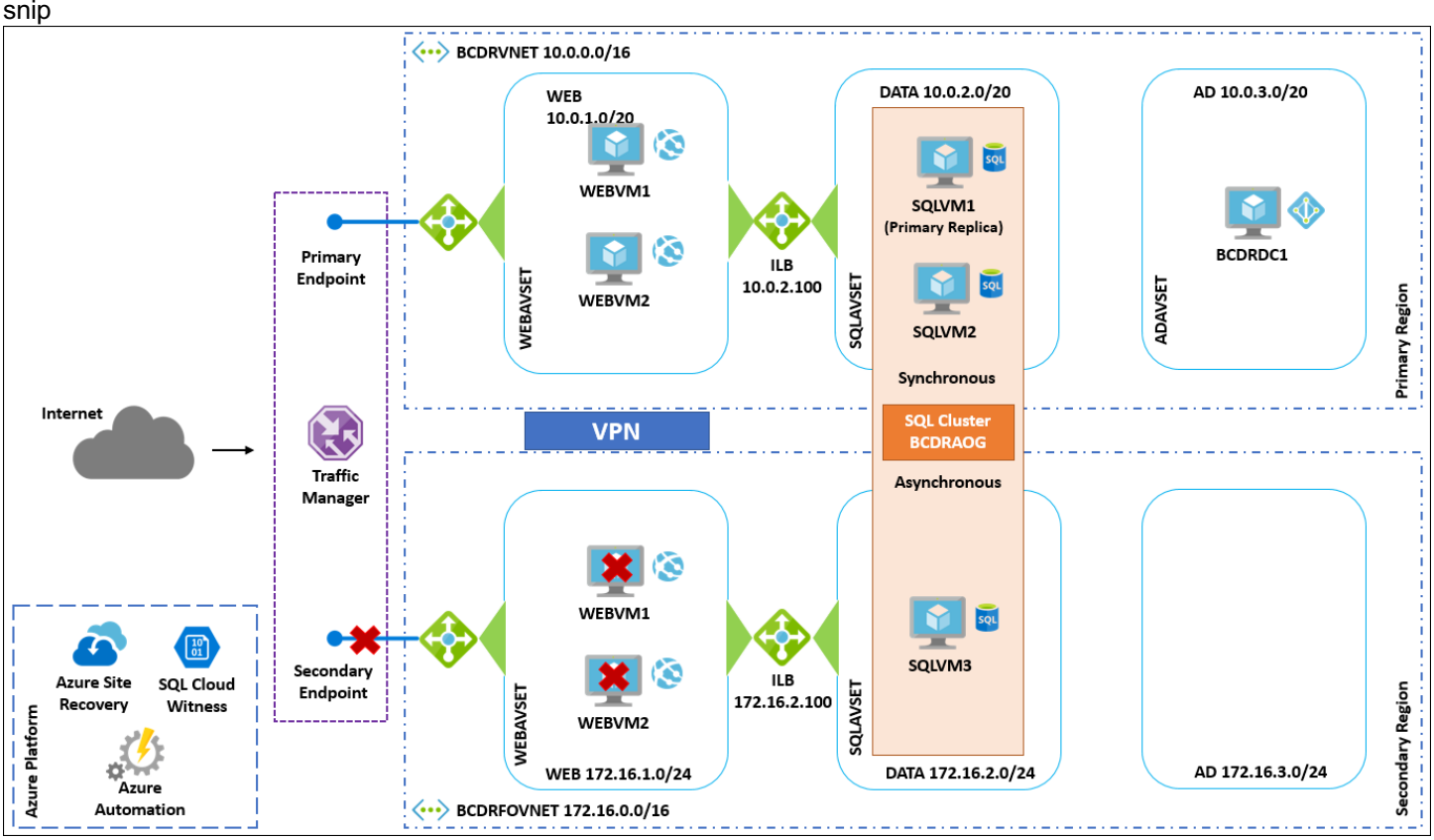
19. Move back to your **BCDRPaaSPrimarySite** resource group and select through to your **SQL Server**. Select the **Failover group**. Select **Failover** and **Confirm**.

20. Once the Failover back to the **Primary** site is completed, the **SQL Server** in the **Primary** site will show as the **Primary**

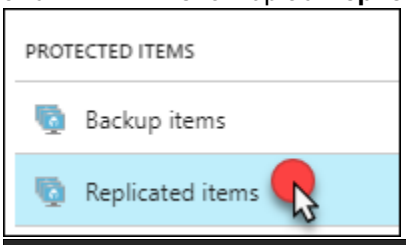


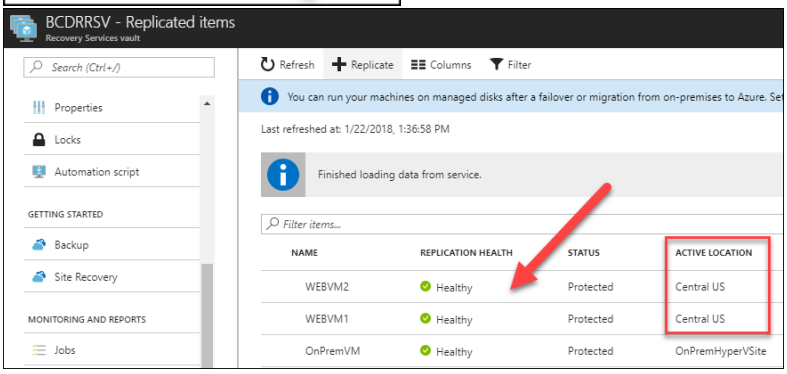
Task 4: Failback Azure IaaS region to region

snip

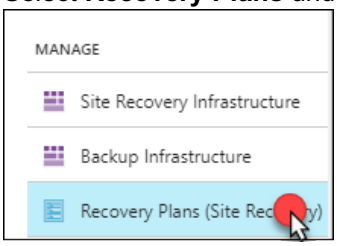


1. Open the **BCDRSRV** and select **Replicated Items** under the **Protected Items** area. Make sure that **WEBVM1** and **WEBVM2** show up ad **Replication Heath**: **Healthy.**



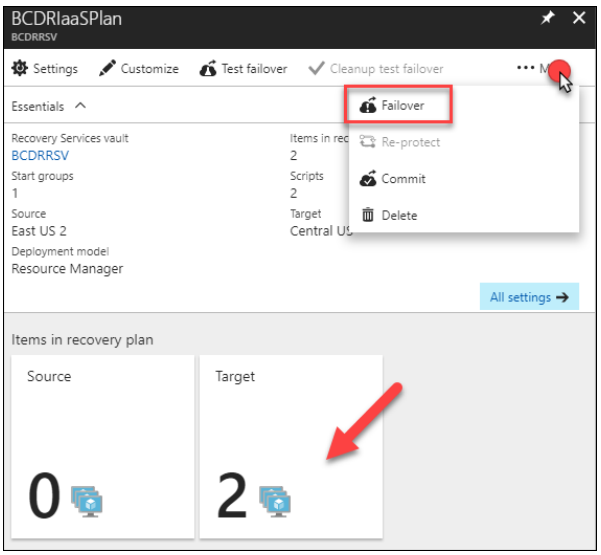


2. Select **Recovery Plans** under the **Manage** area

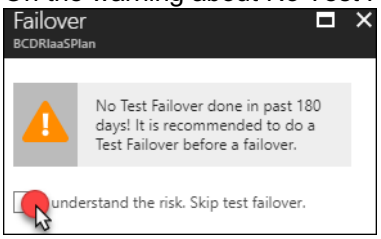


3. Select the **BCDRIaaSPlan**

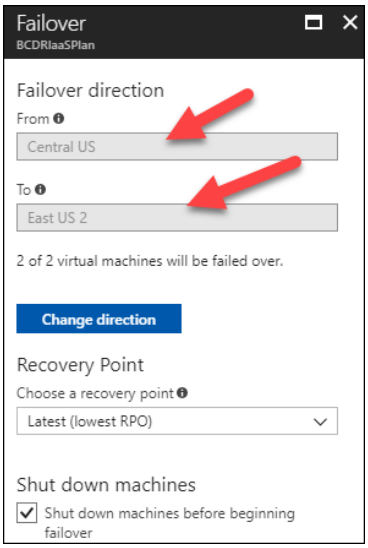
4. Notice that the VMs are still at the Target, since they are Failed over to the Secondary Site. Select **More**, **Failover.**



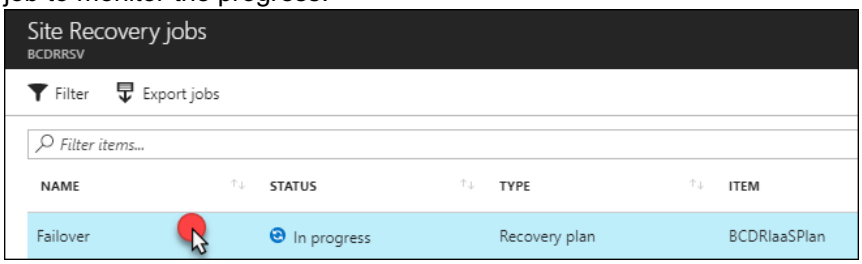
5. On the warning about No Test Failover, select **I understand the risk, Skip test failover.**



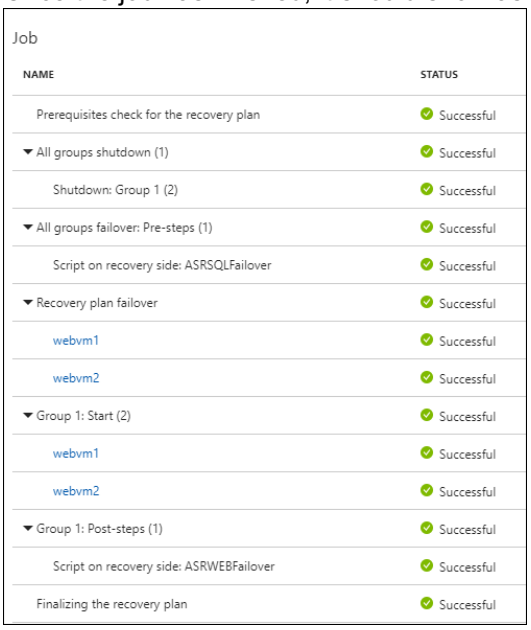
6. Review the Failover direction. Notice that **From** is the **Secondary** site and **To** is the **Primary** site. Select **OK**.



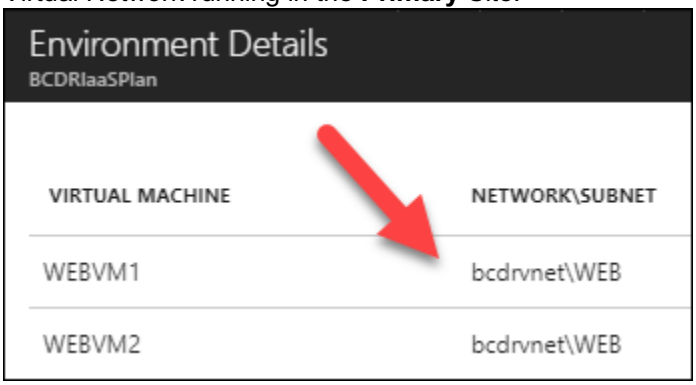
7. After the Failover is initiated close the blade and move to **Jobs**, then **Site Recovery Jobs**. Select the **Failover** job to monitor the progress.



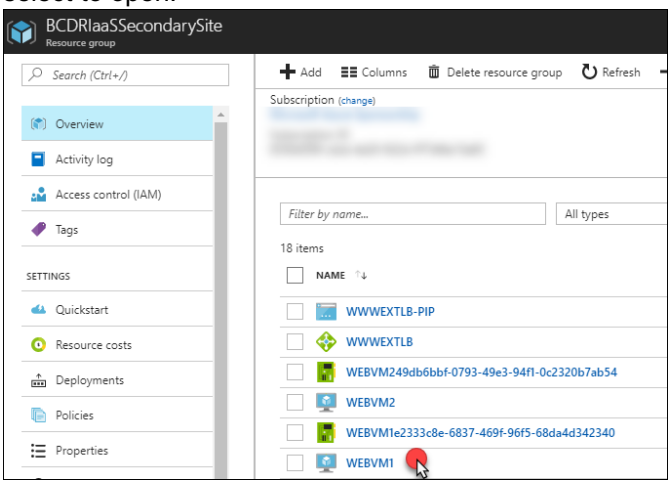
8. Once the job has finished, it should show as successful for all tasks



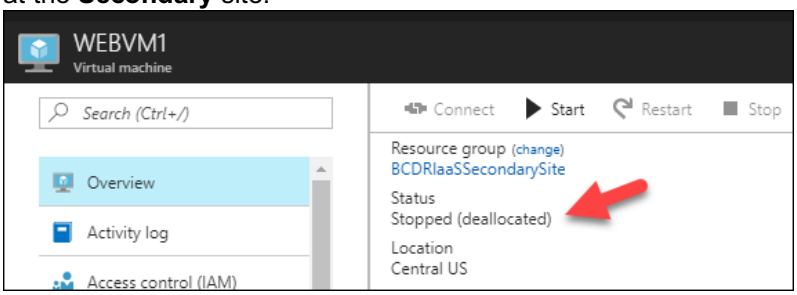
9. There is also details of the failover VMs shown. Notice how they are running the **BCDRVNET**. This is the primary Virtual Network running in the **Primary** Site.



10. Select Resource groups and select **BCDRIaaSSecondarySite**. Locate the **WEBVM1** in the resource group and select to open.

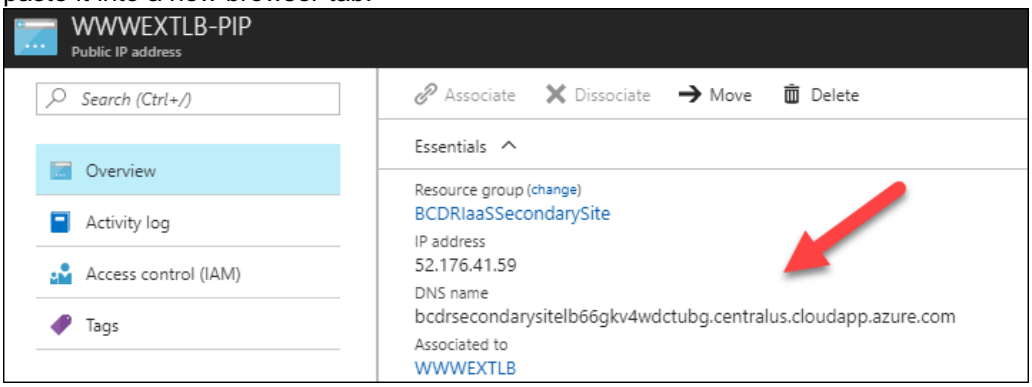


11. Notice that is currently shows as **Status: Stopped (deallocated).** This shows that failover has stopped the VMs at the **Secondary** site.



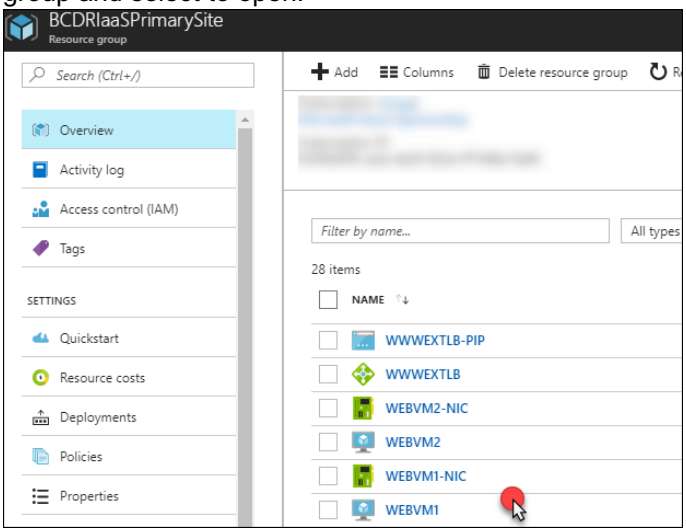
Note: Do not select Start! This is a task only for ASR.

12. Move back to the Resource group and select the **WWWEXTLB-PIP** Public IP address. Copy the DNS name and paste it into a new browser tab.

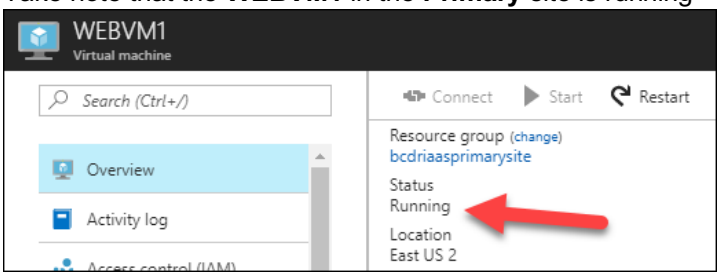


13. The site will be unreachable at the **Secondary** location

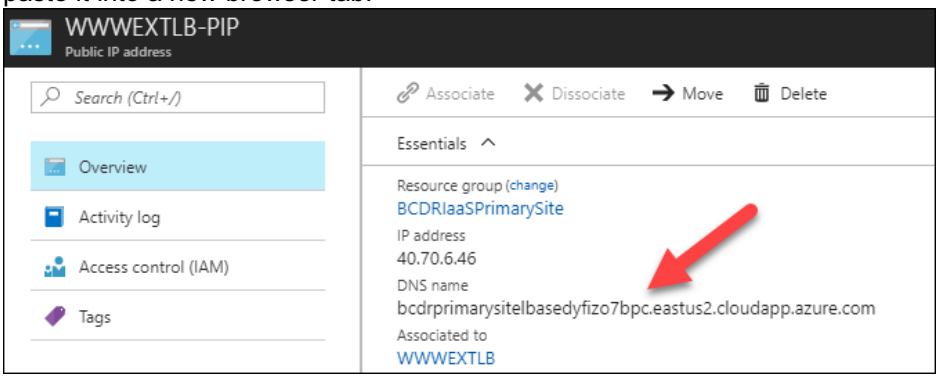
14. In the Azure portal, move to the **BCDRIaaSPrimarySite** resource group. Locate the **WEBVM1** in the resource group and select to open.



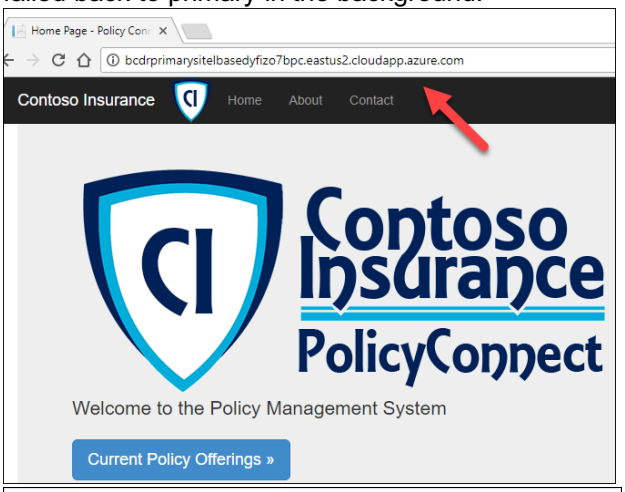
15. Take note that the **WEBVM1** in the **Primary** site is running

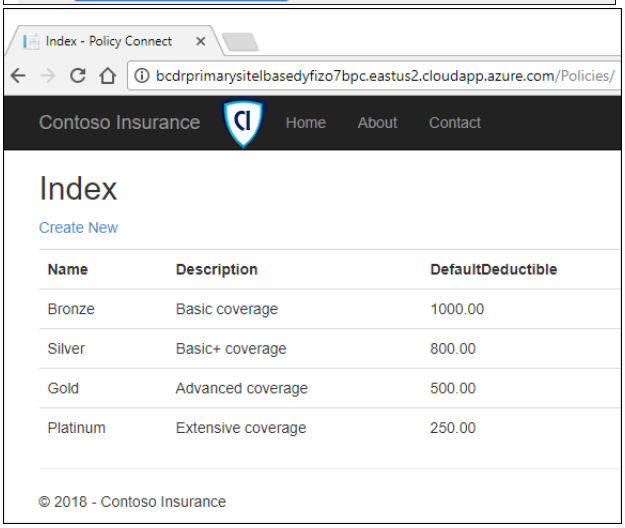


16. Move back to the Resource group and select the **WWWEXTLB-PIP** Public IP address. Copy the DNS name and paste it into a new browser tab.

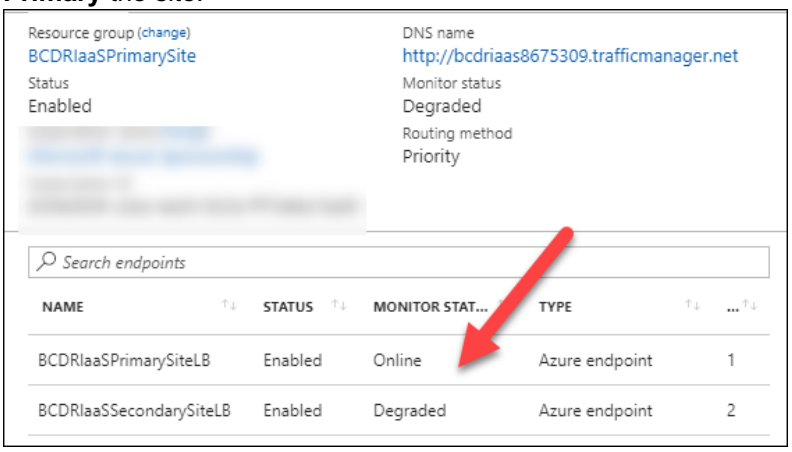


17. The Application running on the **WEBVM1** and **WEBVM2** is now responding from the **Primary** Site. Make sure to select the current **Policy Offerings** to ensure that there is connectivity to the SQL Always On group that was also failed back to primary in the background.

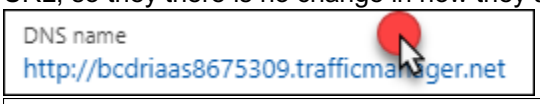


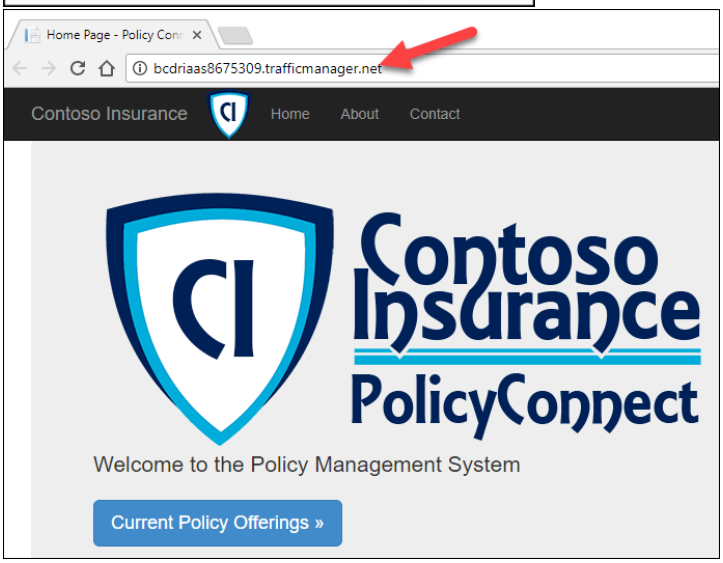


18. Using the Azure portal, locate the **BCDRIaaS Traffic Manager** profile in the **BCDRIaaSPrimarySite** resource group. Notice that the Monitor Status has moved to **Degraded** for the **Secondary** Site and moved to **Online** for **Primary** the site.



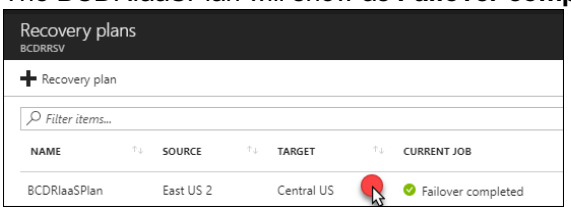
19. Select the **DNS Name URL**. The site load immediately and is failed over. The users will always be using this DNS URL, so they there is no change in how they access the site even though it is failed over.





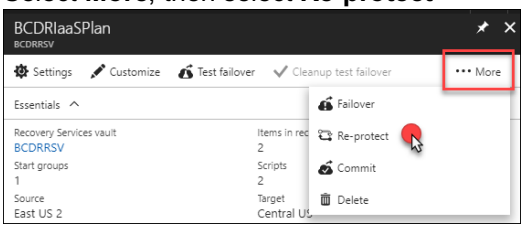
20. Now, that you have successfully failed back you need to prep ASR for the Failover again. Move back to the **BCDRSRV** Recovery Service Vault using the Azure portal. Select Recovery Plans on the ASR dashboard.

21. The BCDRIaaSPlan will show as **Failover completed.** Select the Plan.

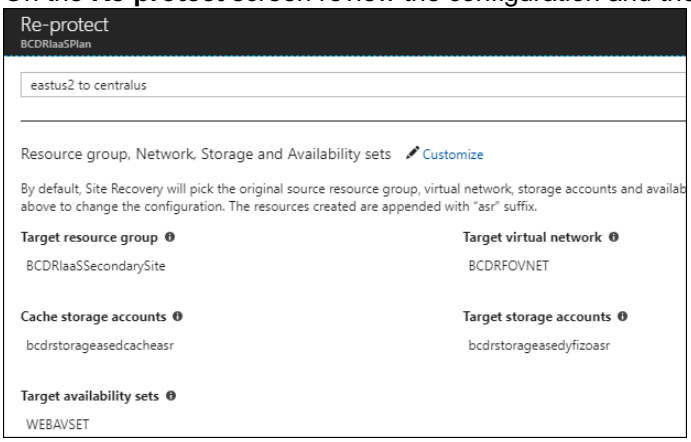


22. Notice that now 2 VMs are show in the **Source**

23. Select **More**, then select **Re-protect**

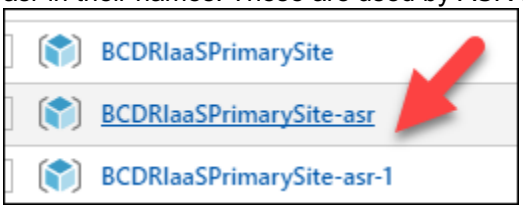


24. On the **Re-protect** screen review the configuration and then select **OK**



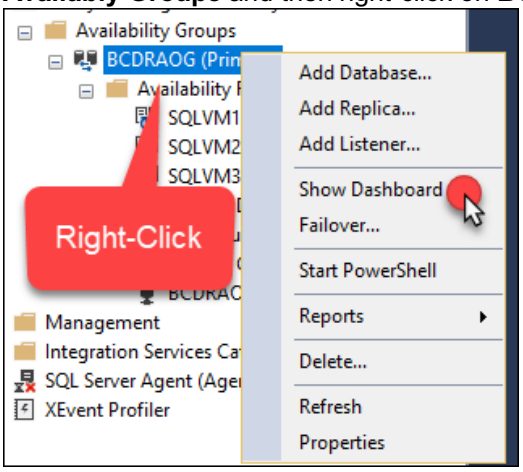
25. The portal will submit a deployment. This process will take some time, by first committing the failover and then synchronizing the **WEBVM1** and **WEBVM2** back to the **Primary** Site. Once this process is complete, then you will be able to Fail over again from the **Primary** to **Secondary** site from the perspective of ASR.

26. Select **Resource groups** in the Azure portal and notice that two new Resource groups have been created with -asr in their names. These are used by ASR for the Failover and Failback, so they should be left in place.

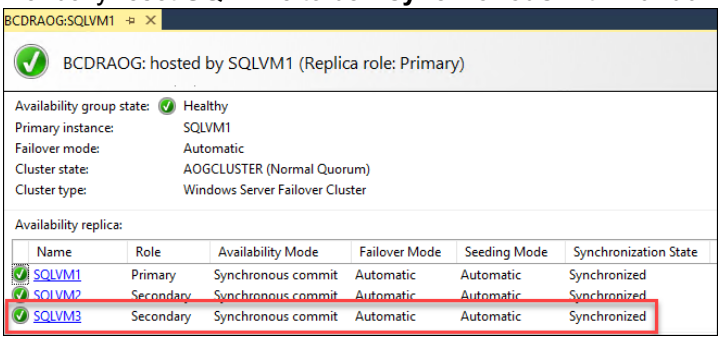


27. Next, you need to reset the SQL AOG environment to ensure a proper failover. To do this open Remote Desktop to **BCDRDC1** and then jump to **SQLVM1** using Remote desktop.

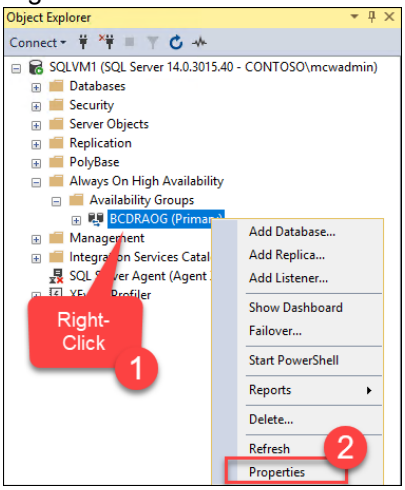
28. Once connected to **SQLVM1** open SQL Management Studio and Connect to **SQLVM1**. Expand the **Always On Availably Group**s and then right-click on **BCDRAOG** and then select **Show Dashboard**.



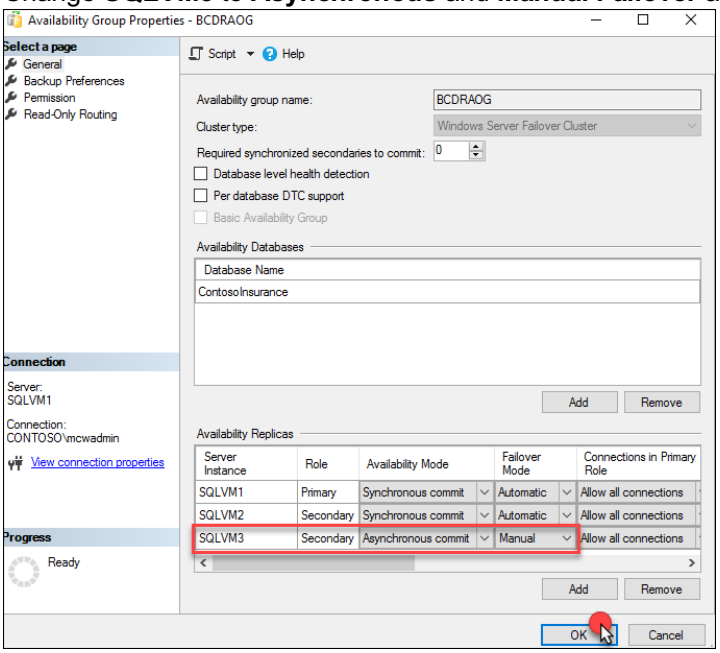
29. Notice that all the Replica partners are now Synchronous Commit with Automatic Failover Mode. You need to manually reset **SQLVM3** to be **Asynchronous** with **Manual Failover**.



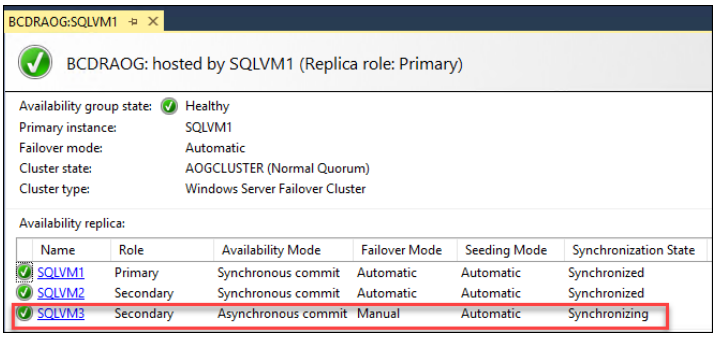
30. Right-click the **BCDRAOG** and select **Properties**



31. Change **SQLVM3** to **Asynchronous** and **Manual Failover** and select **OK**



32. Show the Availability Group Dashboard again. Notice that they change has been made and that the AOG is now reset.



Note: This task could have been done using the Azure Automation script during Failback, but more DBAs would prefer a good clean failback and then do this manually once they are comfortable with the failback.

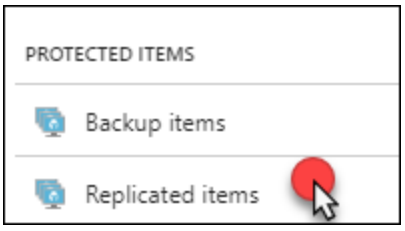
After the hands-on lab

Duration: 15 minutes

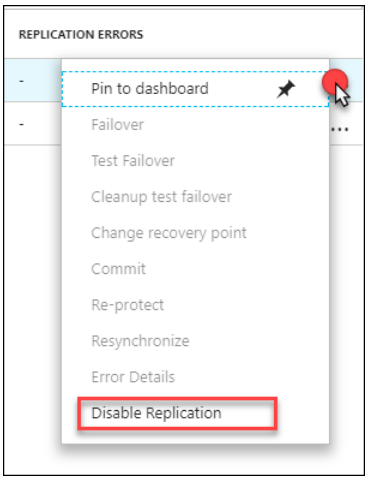
There are many items that were created as a part of this lab, and they should be deleted once you no longer desire to retain the environments.

Task 1: Disable replication in the recovery services vault

1. To clean up the environment, you must first disable the replication of the **WEBVM1, WEBVM2** and **OnPremVM** in the **BCDRSRV** Recovery Service Vault. Open **BCDRSRV** in the Azure portal and select **Replicated Items** in the **Protected Items** area.



2. Select the **Ellipse** next to each replicated item and then select **Disable Replication**. On the following screen select **OK**



3. After this process is completed you can move on to the next task

Task 2: Delete all BCDR resource groups

1. Using the Azure Portal delete each of the BCDR Resource Groups that you created

Resource Group Name	Location
BCDRAzureAutomation	Your Location
BCDRAzureSiteRecovery	Secondary
BCDRIaaSPrimarySite	Primary
BCDRIaaSSecondarySite	Secondary
BCDRLabRG	Your Location
BCDROnPremPrimarySite	Primary
BCDROnPremPrimarySite-asr	Primary
BCDROnPremPrimarySite-asr-1	Primary
BCDRPaaSPrimarySite	Primary
BCDRPaaSSecondarySite	Secondary

You should follow all steps provided *after* attending the Hands-on lab.