Your fastest way to learn. Why wait?

FIREBRAND

Microsoft

Implementing Microsoft Azure Infrastructure Solutions

Labs

Courseware Version 3.0

KIT CODE: K-400-02

Your fastest way to learn. Why wait?

FIREBRAND

Table of Contents

Your fastest way to learn. Why wait?

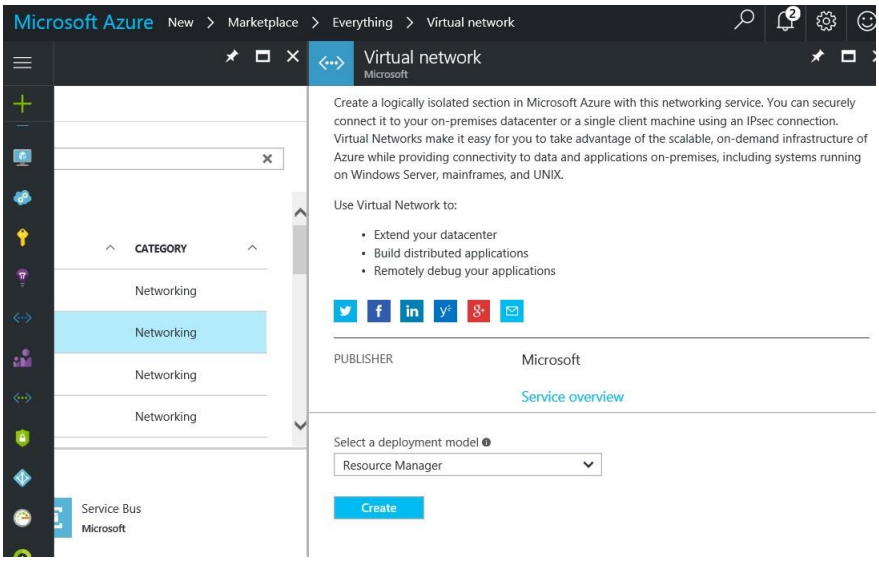
FIREBRAND

Lab 1 Azure Networking

In this lab, you will create 2 Azure VNets. One classic VNET using the Classic Portal and one ARM VNet using the ARM Portal. You will then use VNET Peering to connect the two VNets together. Finally, you will create a new Virtual Network Gateway and use it to create a Point-to-Site connection.

Exercise 1 Creating an ARM VNet

1. Using the Http://Portal.Azure.Com login to the ARM Portal.
2. Use the  icon to search for and create a new Virtual Network, making sure you choose to create a Resource Manager Virtual Network



3. Create a virtual network with the following settings:

Name: ARMVnet

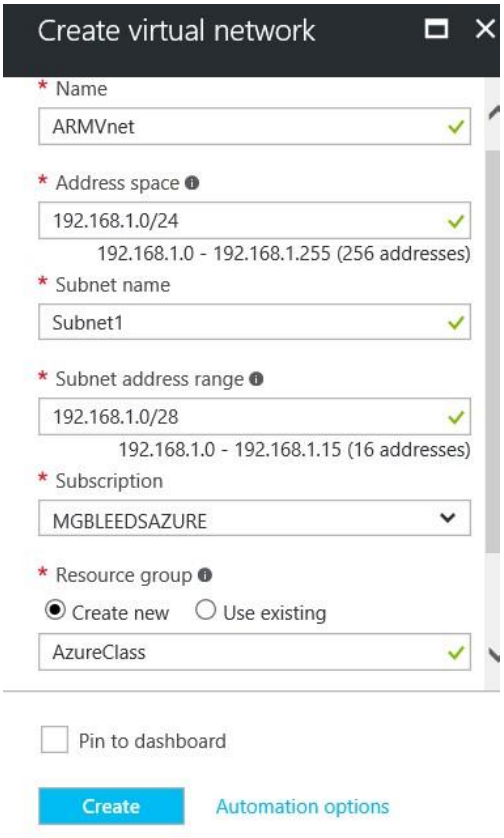
Address Space: 192.168.X.0/24 (Where X is your student number provided by your instructor)

Subnet Name: Subnet1

Subnet Address Range: 192.168.X.0/28 (Where X is your student number provided by your instructor)

Resource Group: Create a new Resource group called AzureClass

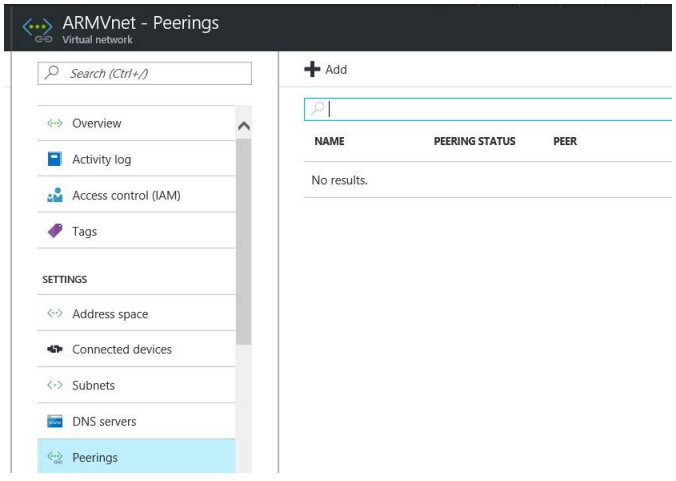
Region: North Europe



Exercise 2 Creating a Peering between your two VNets

1. Go to the properties of your ARM VNet and select the 

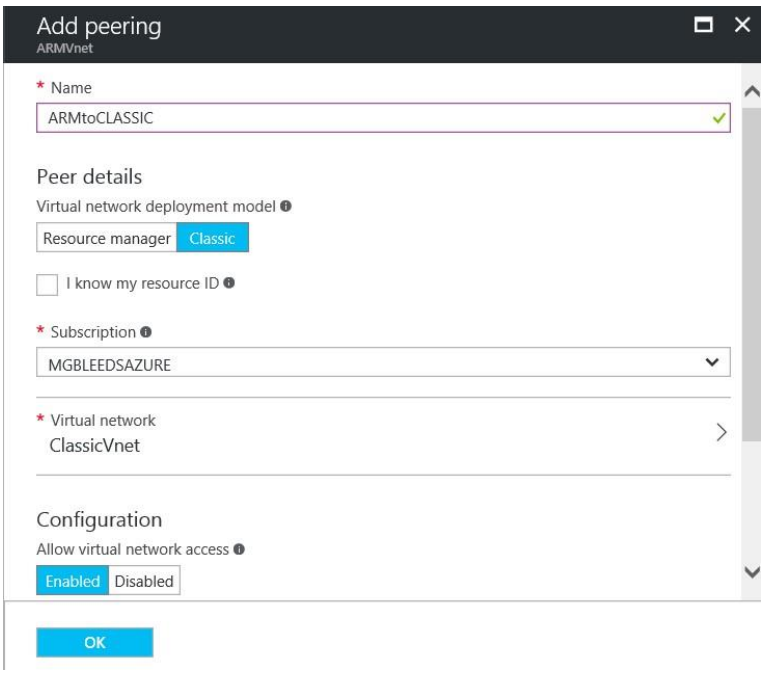
section



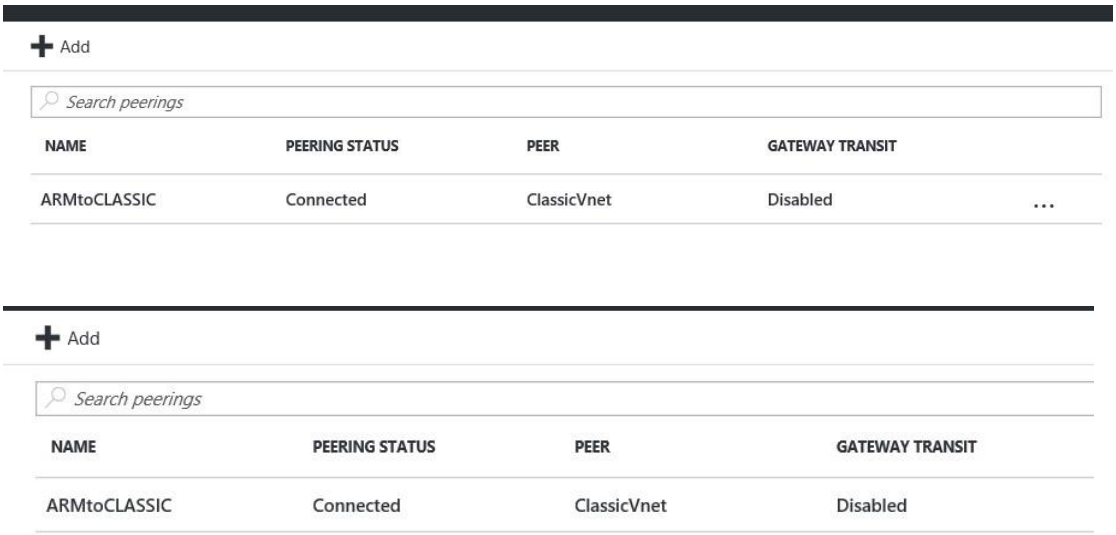
2. Use the 

to create a new Peering with the following settings:

Name: ARMtoCLASSIC
Peer Details: Classic
Virtual network: ClassicVnet

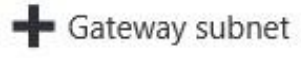


After a few minutes the Peering connection should show Status as Created and then shortly after it should show as connected.

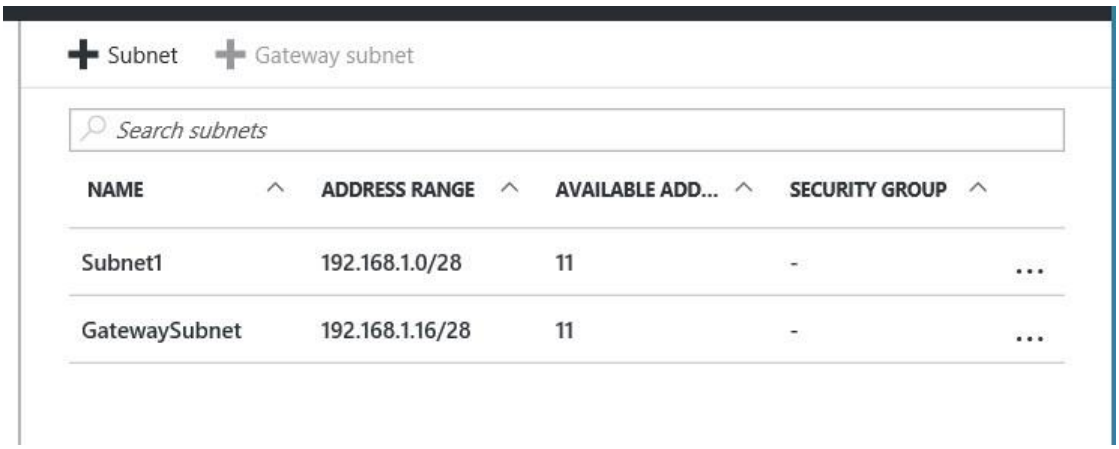


In a later exercise, you will deploy VMs to each of these VNets and they should be able to connect to each other.

Exercise 3 Creating a Virtual Network Gateway

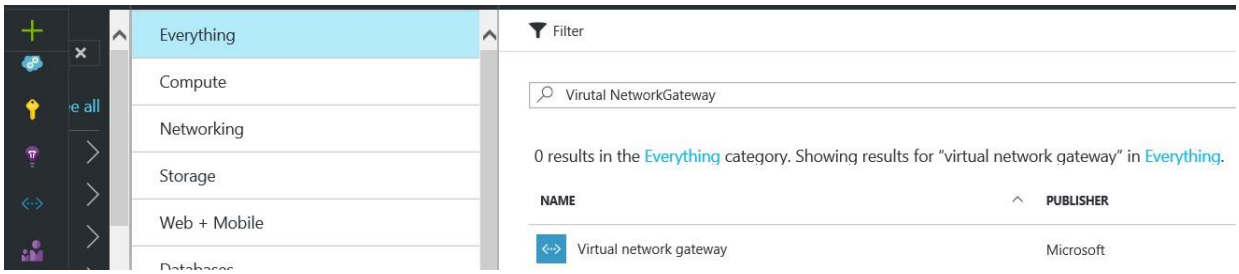
1. Go to the properties of your ARM VNet and go to the Subnets section then using the 

option create a new Gateway Subnet with the default settings.



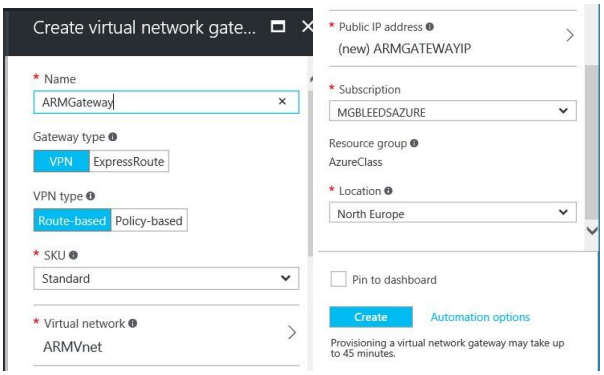
You will be deploying your new Virtual Network Gateway to this GatewaySubnet

2. Using the  create a new Virtual Network Gateway



3. Use the following settings to create your new Virtual Network Gateway
   Name: ARMGateway
   Gateway Type: VPN
   VPN Type: Route-based
   SKU: Standard
   Virtual Network: ARMVnet
   Public IP Address: Create a new public IP Address with a name of ARMGATEWAYIP

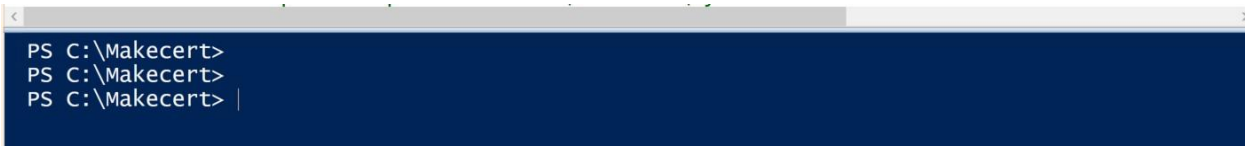
Leave everything else at their defaults.



Use the **Create** button to create your gateway, this process can take up t 45mins to complete. While your ARM Gateway is being created you can generate the certificates that will be needed to authenticate your Point-to-Site connection. You will be using

Makecert.exe to generate a root and client certificate locally, then once your Gateway is created you can upload the root certificate to Azure. Mkaecert.exe should have been installed in c:\makecert folders in the local host server in the classroom. If it is not installed there ask your instructor where it is.

4. Open the ISE as Administrator and the in the console pane navigate to the c:\makecert folder.

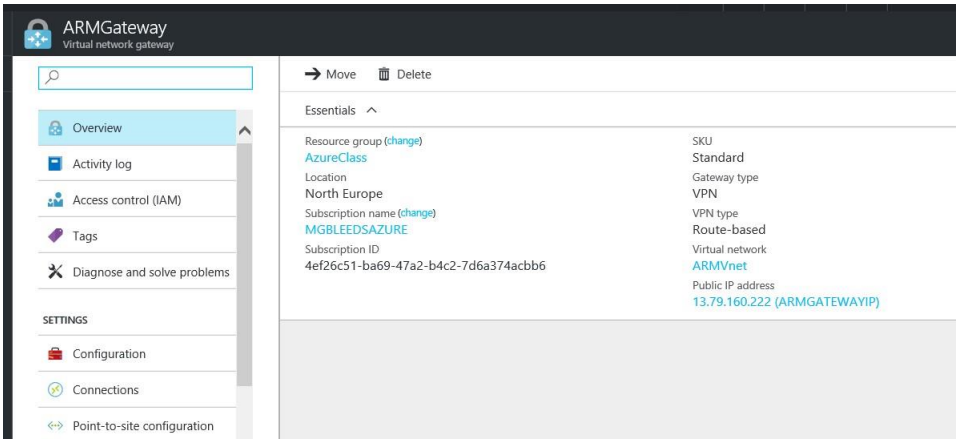


5. Use the commands below to create the Root and client certificate that will be used in this lab (the line number in the screen shot below are irrelevant also be aware that there are 2 commands below, they are separated on to multiple lines using the ` tick.)

```
2   #Make RootCert and Client Cert
3   .\makecert.exe -sky exchange -r -n "CN=ARMP2SRootCert" -pe -a sha1 `
4               -len 2048 -ss My "ARMP2SRootCert.cer" -b 01/02/2017
5
6   .\makecert.exe -n "CN=ARMP2SClientCert" -pe -sky exchange -m 96 -ss My `
7               -in "ARMP2SRootCert" -is my -a sha1 -b 01/02/2017
8
```

```
PS C:\Makecert> .\makecert.exe -sky exchange -r -n "CN=ARMP2SRootCert" -pe -a sha1 `
            -len 2048 -ss My "ARMP2SRootCert.cer" -b 01/02/2017
Succeeded

PS C:\Makecert> .\makecert.exe -n "CN=ARMP2SClientCert" -pe -sky exchange -m 96 -ss My `
            -in "ARMP2SRootCert" -is my -a sha1 -b 01/02/2017
Succeeded

PS C:\Makecert>
```

Now that your certificates are created you will need to wait for the gateway to be created before you continue. Your gateway is only fully created once it has been given a Public IP Address.



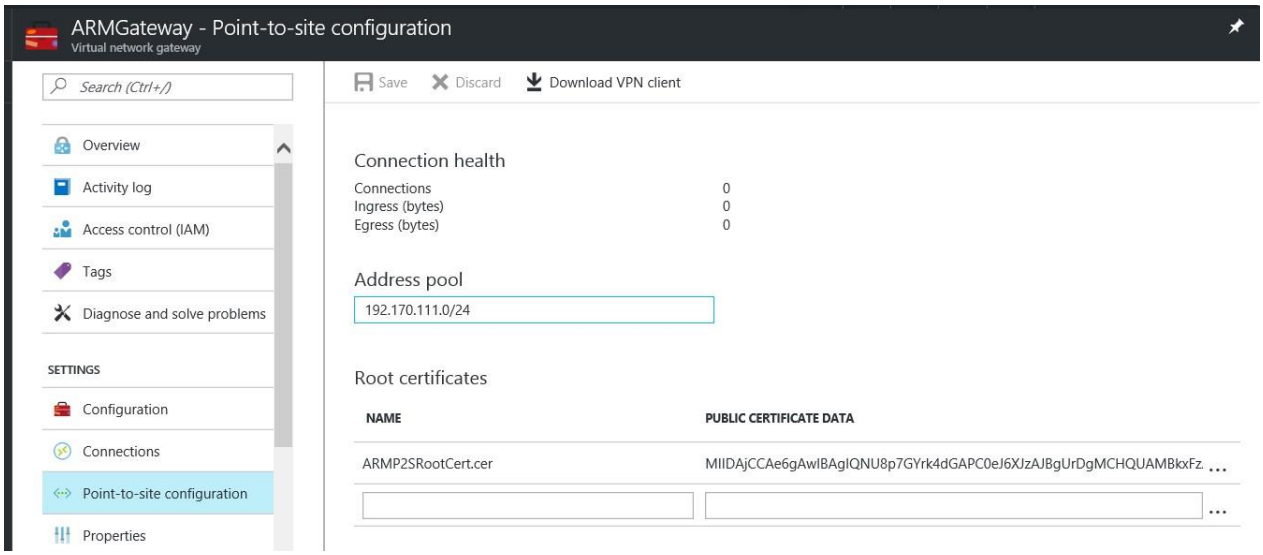
6. From the ISE use the Add-AzureRMAccount command to login to your account (If you have multiple subscriptions you might have to use the SelectAzureRMSubscription cmdlet to choose the correct subscription)
7. Once your gateway has been created you can use the commands below to complete your Point-to-Site configuration.

```
#Convert Root Cert

$RootCert = "ARMP2SRootCert.cer"
$filePathForCert = "C:\Makecert\ARMP2SRootCert.cer"
$cert = new-object System.Security.Cryptography.X509Certificates.X509Certificate2($filePathForCert)
$CertBase64 = [system.convert]::ToBase64String($cert.RawData)
$p2srootcert = New-AzureRmVpnClientRootCertificate -Name $RootCert -PublicCertData $CertBase64
```

```
#Get Gateway
$name = "ARMGateway"
$res = "AzureClass"
$Gateway = Get-AzureRmVirtualNetworkGateway -Name $name -ResourceGroupName $res

#Set Client addresspool and RootCert
Set-AzureRmVirtualNetworkGateway -VirtualNetworkGateway $Gateway `
 -VpnClientAddressPool "192.170.111.0/24" -VpnClientRootCertificates $p2srootcert
```

The commands above are in three sections. Firstly, you will convert the root cert into a format that can be used by Azure. Secondly you will use variables to store information about the gateway you have created. Thirdly you will create the Point-to-Site configuration by assigning the certificate to the gateway and creating an Addresspool. NOTE. That the command Set-AzureRMVirtualNetworkGateway command is a single command on one line.

Once configured if you go back to your gateway on the portal you will see a configuration like the one below:

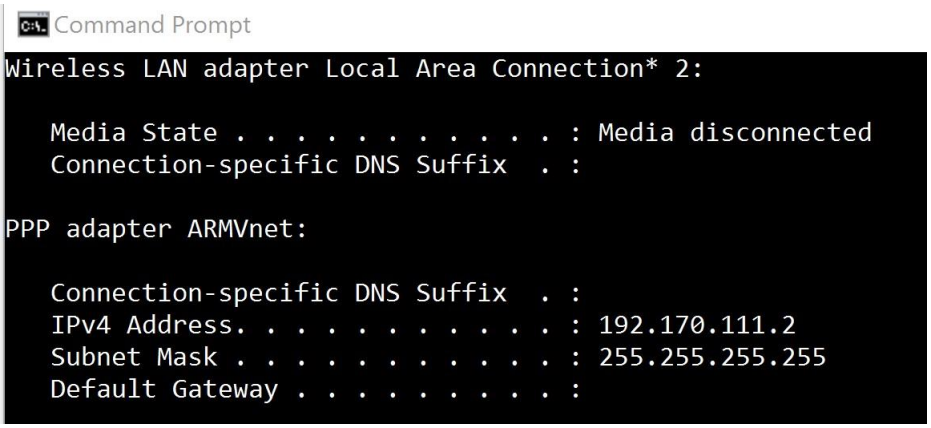


8. Use the 

option to download and install the x64 VPN Client (The client can take a few minutes to be made available)

9. Once the VPN client is installed use it to connect to you Azure VNet.#

If the connection is established, you should see a PPP connection if you run Ipconfig from the cmd prompt

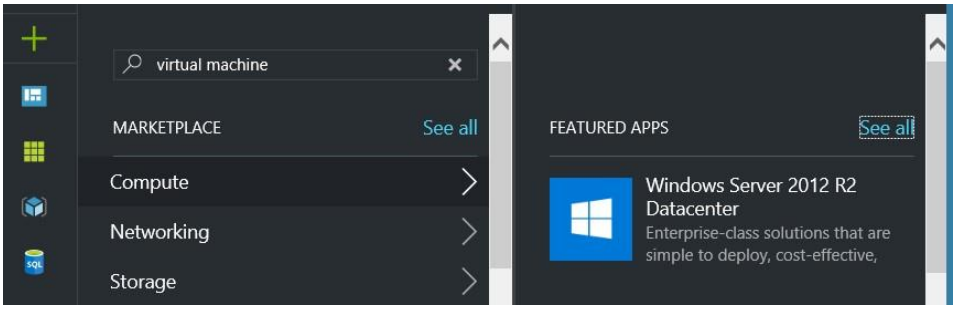


Lab 2 Deploying Azure Virtual Machines

In this Lab, you will deploy virtual machines to your newly created VNets.

Exercise 1 Creating an ARM and Classic Virtual Machines

1. From the portal use the + to create a new Virtual Machine selecting Compute and then choose to create a Windows Server 2012 R2 Datacentre Virtual Machine. Make sure you select to create an Resource Manager virtual machine.



2. Create a virtual machine with the following settings:

Name: VM1

VM Disk Type: HDD

Username: *YourName*

Password: *Use a complex Password*

Resource group: Use Existing named AzureClass

Location: NorthEurope

Size: D1_V2

Storage: Yes Use Managed Disks

Network: ARMVnet

Subnet: Subnet1

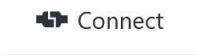
Leave all other settings at their default and click OK on the Summary screen.

3. Repeat step 2 this time choose to create a **Classic Virtual Machine** named **VM2** and place it on the classic VNet you created earlier. Choose **Standard** as the **Disk type** and accept the defaults for the storage account and cloud services but make sure you choose Virtual Network **ClassicVnet** and subnet **Subnet1** for the virtual networking section. Finally click OK on the summary screen to start the deployment of VM2.

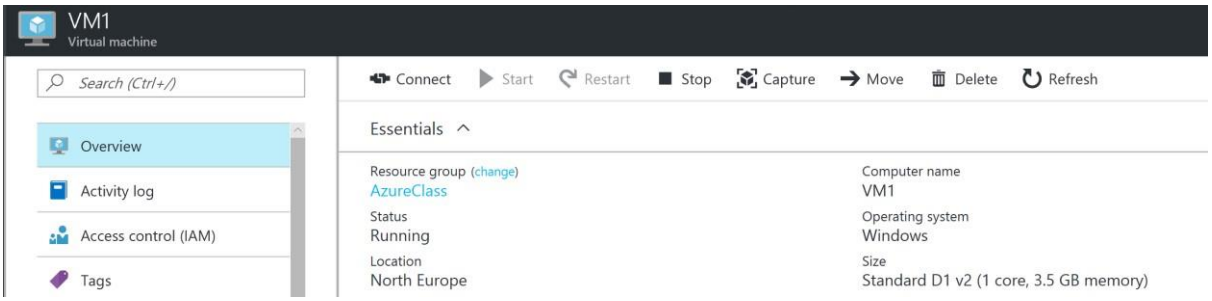
Exercise 2 Customising your newly deployed Virtual Machines

In this exercise, you will connect to both virtual machines and disable the host based firewall and install IIS on the virtual machine.

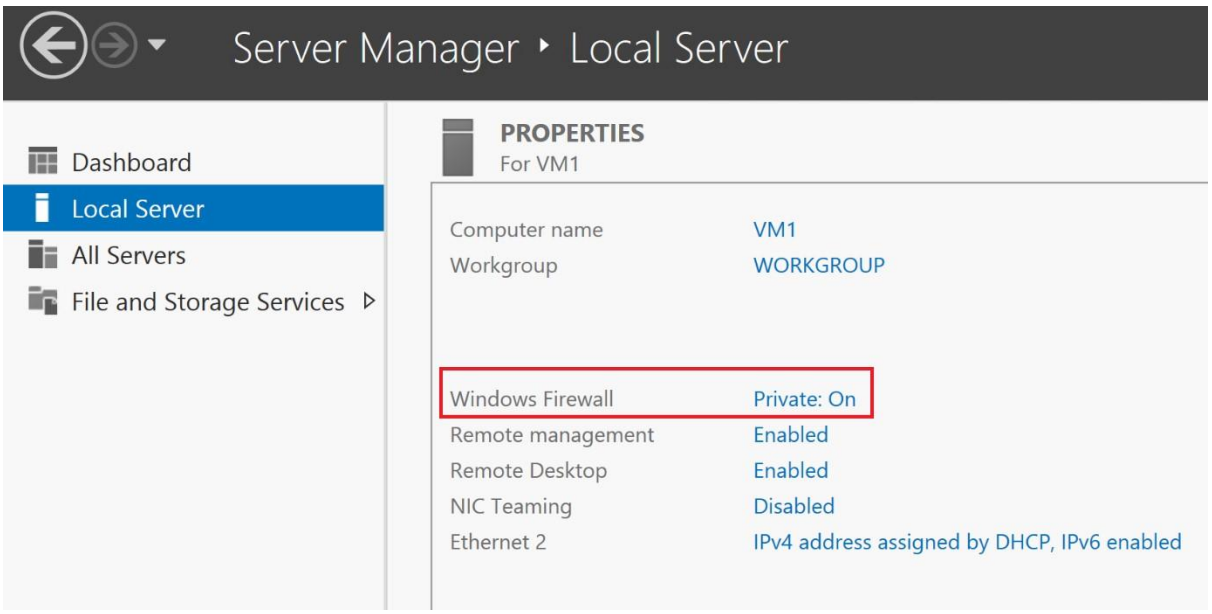
1. Using the Portal navigate to the properties of VM1 and use the Connect button



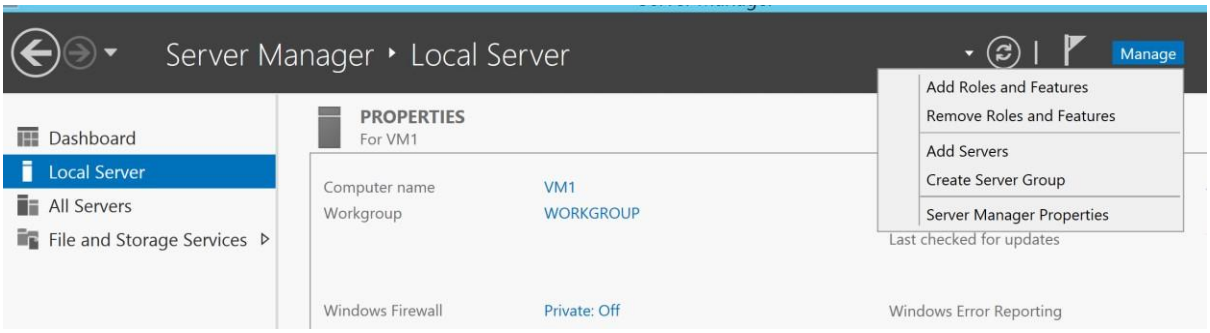
to connect to the Virtual Machine using the username and password you entered when you created the Virtual Machine.



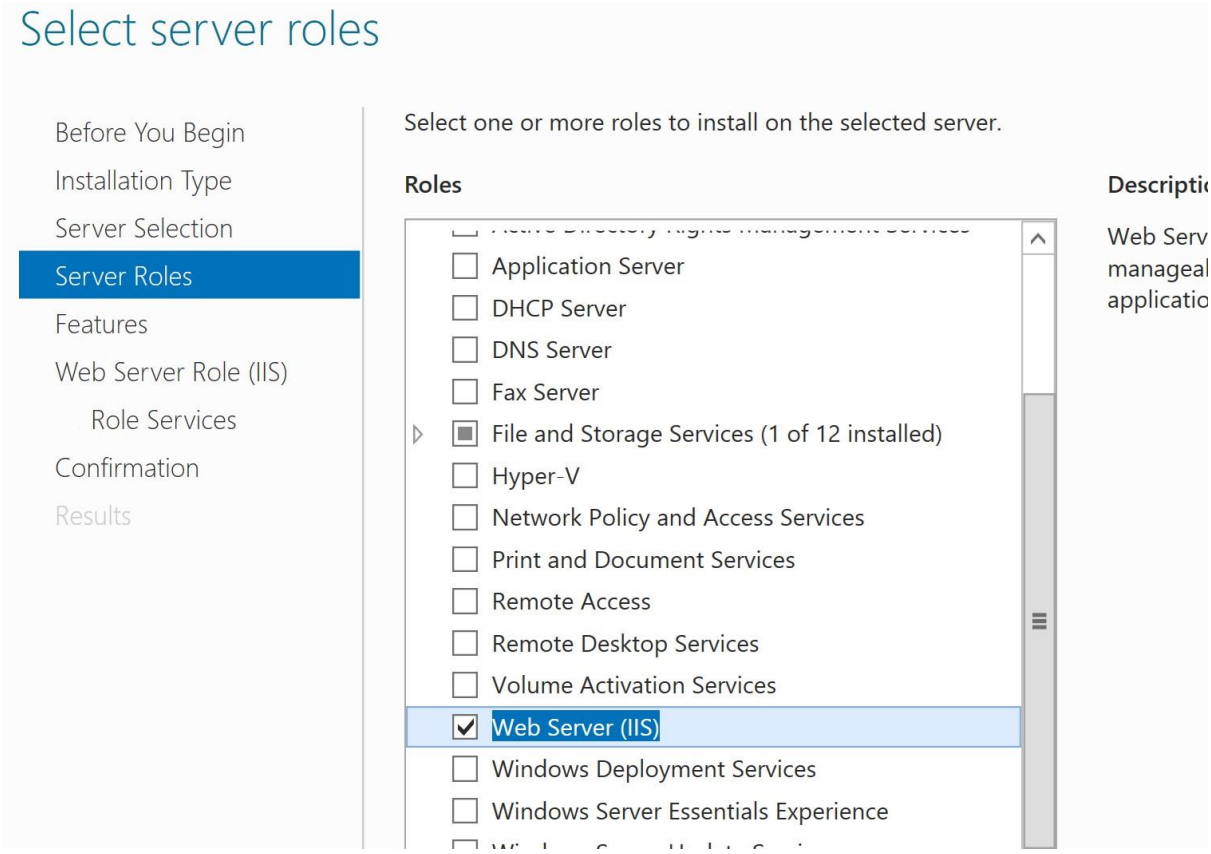
2. Once connected use Server Manager to disable the Windows Firewall



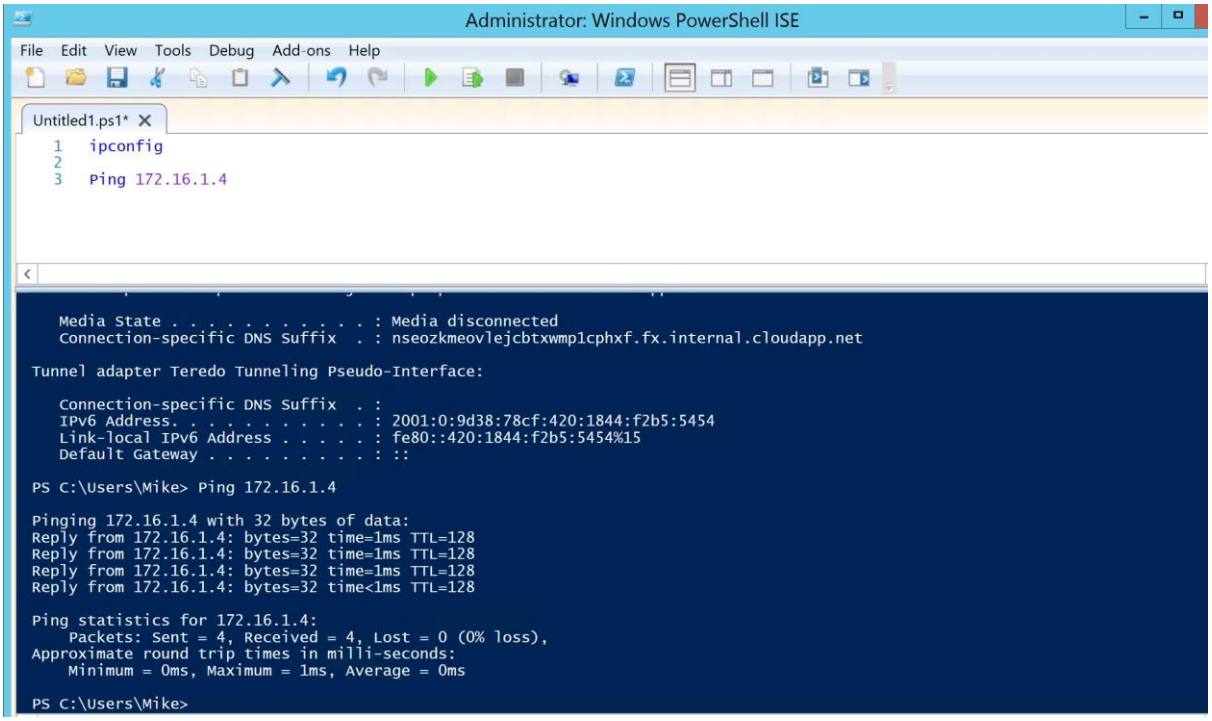
3. Using Server Manager select Manage and then Add Roles and Features



4. Follow the wizard through and add the Web Server (IIS) Role



5. Repeat steps 1 – 4 but using VM2
6. Once you have disabled the Firewall on both VM1 and VM2 you should be able to use the Command Prompt or PowerShell ISE to ping each other. Ping VM2's IP Address from VM1



Both VMs are on different Virtual Networks but because we have configured VNet Peering both VMs can contact each other.

Lab 3 Deploying Web Apps

Exercise 1 Creating a Web App

In this exercise, you will create a web app

1. From the Portal navigate t your AzureClass resource group and then click + Add 2. Using the filter search for Web App, then select Web App and click Create.

3. Create a Web App with the following settings

Name: YourName

Resource Group: AzureClass

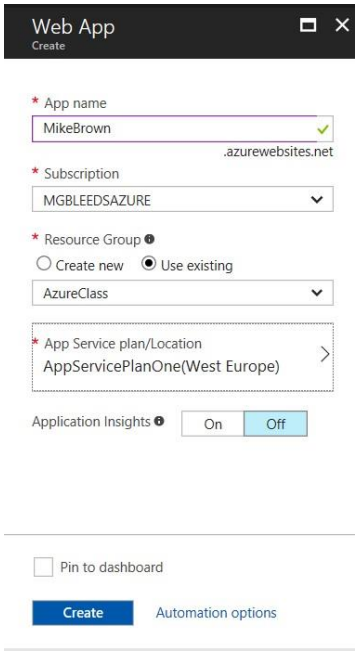
App Service Plan/Location: Select it then select Create New

App Service Plan Name: AppServicePlanOne

Location: West Europe

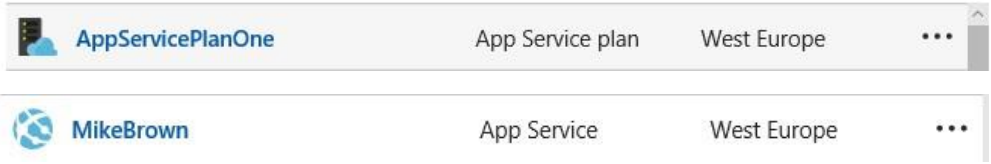
Pricing Tier: S1 Standard

Application Insights: Off

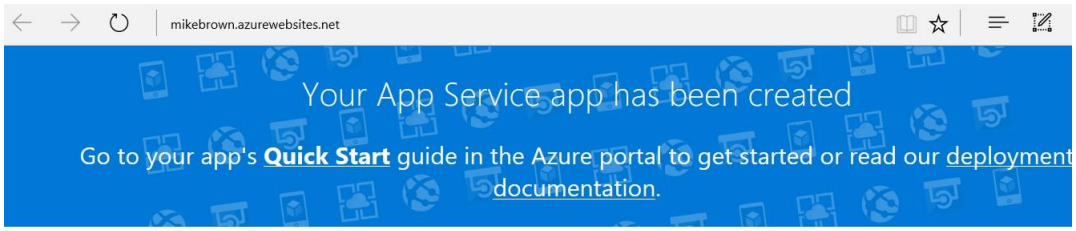


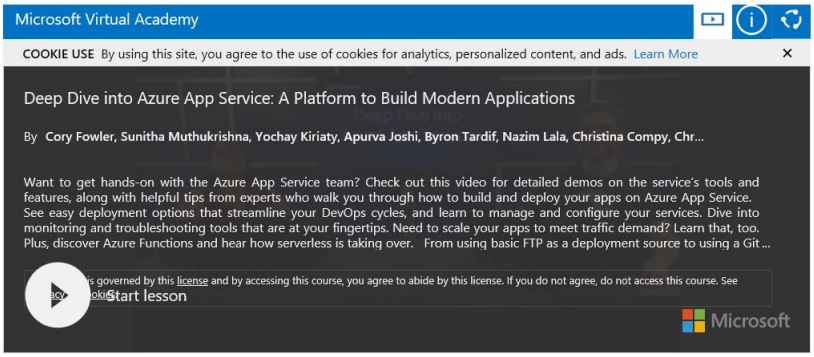
4. Click Create to Create your new Web APP

It will take a few minutes to create your new Web App and its associated App Service Plan. Once created both will appear in your AzureClass resource group.



5. Click on your Web App to open it properties. The Overview page would be displayed. Here you should be able to find the URL assigned to your web app. If you double click it, it should open the holding page for your Web App.

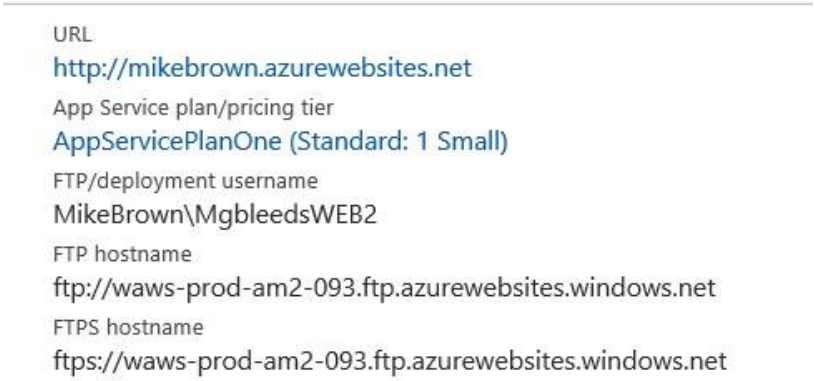




Exercise 2 Populating your web App

In this exercise, you will populate your web app with custom content

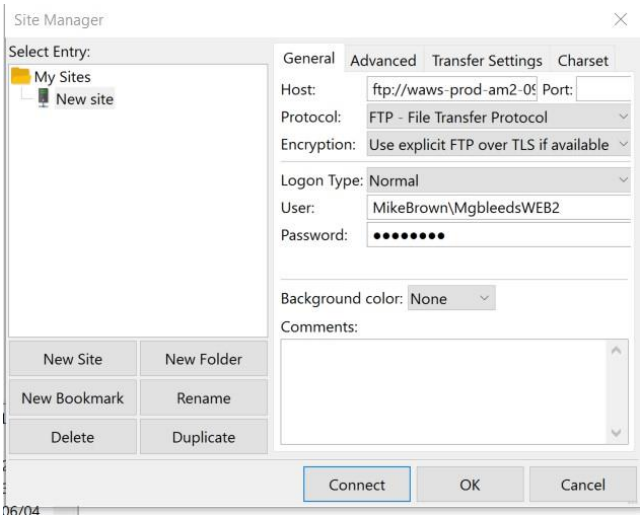
1. From the properties of you web app select the Deployment credentials section and type in new username and a new password for this deployment. Click save to save your new password.
2. From the Overview page of your web app make a note of the FTP/deployment Username and the FTP Hostname



We will new use FileZilla to upload new content for your Web App.

3. From your host Server, open FileZilla, from the File menu Select Site Manager and then New Site.
4. In the host name Section type (Copy / Paste) the FTP hostname recorded from the overview page of your Web App
5. In the Logon Type box select Normal
6. In the Username box type the FTP/Deployment username recorded from the overview page of your Web App
7. In the Password box type the password you create in step 1.

You should see a box like the one below:

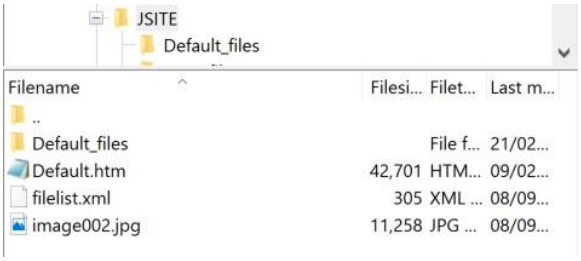


8. Click Connect to connect to you Web App. If you are shown a server certificate screen click OK.

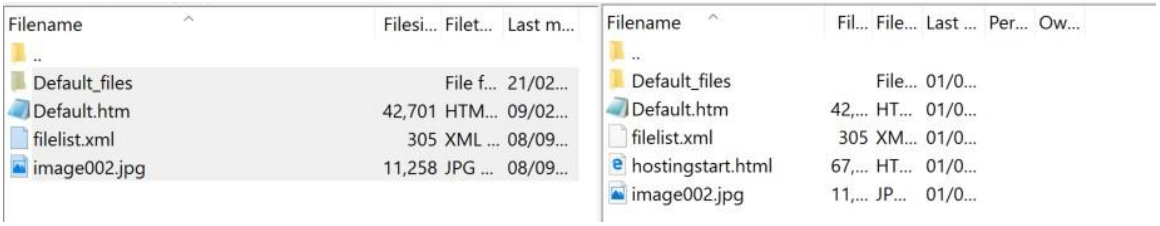
You should be connect to your web app.

9. On the right hand side of the screen in the FileName section navigate to the Site > wwwroot folder

10. On the left hand side in the filename section navigate to the c:\jsite folder



11. Copy the content of the c:\jsite folder to the /site/wwwroot folder of your web app.



Now you have copied your new web site files to your web app use a browser to navigate to the URL of your Web APP.

Lab 4 Working with Azure Storage

Exercise 1 Working with BLOB storage

In this exercise, you will be working with Azure Blob Storage and Storage explorer.

1. Login to the Azure console
2. Navigate to your Resource Group called AzureClass and then click 

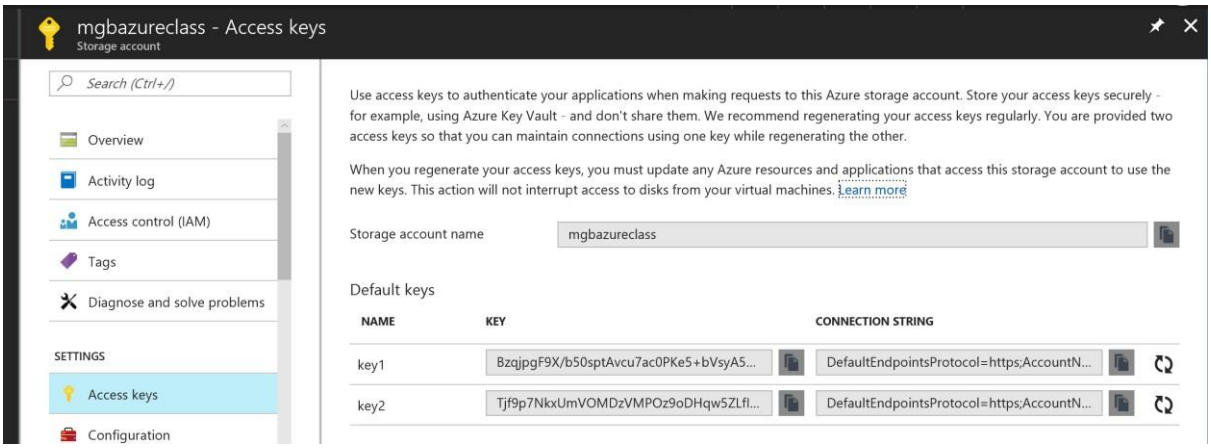
3. In the search box type Storage Account and select Storage Account to create a new storage account:

4. In the Crate Storage Account blade create the storage account with the following details:
   Name: **xxxazureclass** (Where xxx are your initials)
   Deployment Model: **Resource Manager**
   Account Kind: **General Purpose**
   Performance: **Standard**
   Replication: **Locally redundant Storage** (LRS)
   Storage Service Encryption: **Disabled**

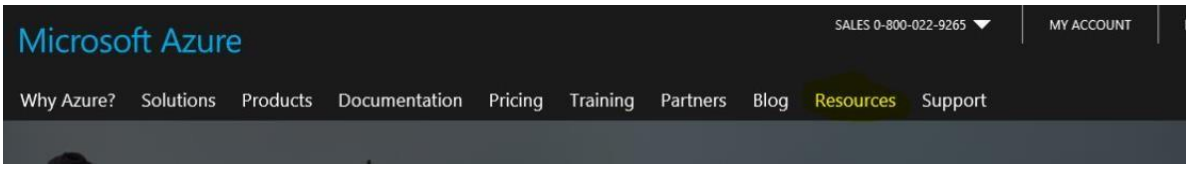
Secure Transfer Required: **Disabled**
Resource Group: **AzureClass**
Location: **North Europe**
Then click **Create**

It will take just a couple of minutes to create your new storage account.

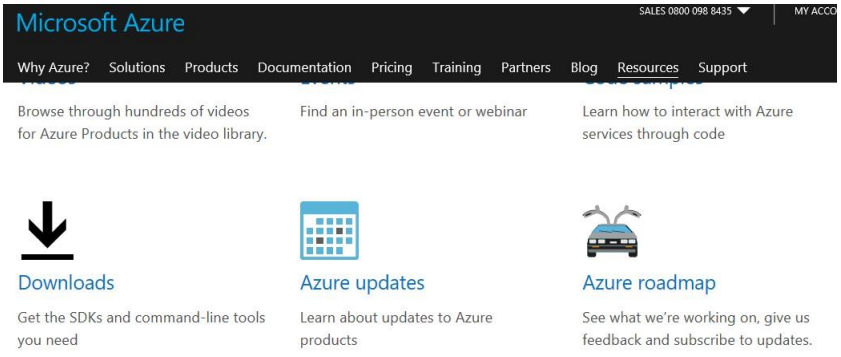
5. Once your storage account has been created navigate to its properties and the Access Keys section. Notice the two Default Keys and copy Key1



6. Go to the Azure Website Azure.Microsoft.com and click Resources



7. From the resources page scroll towards the bottom of the page and click on the downloads section



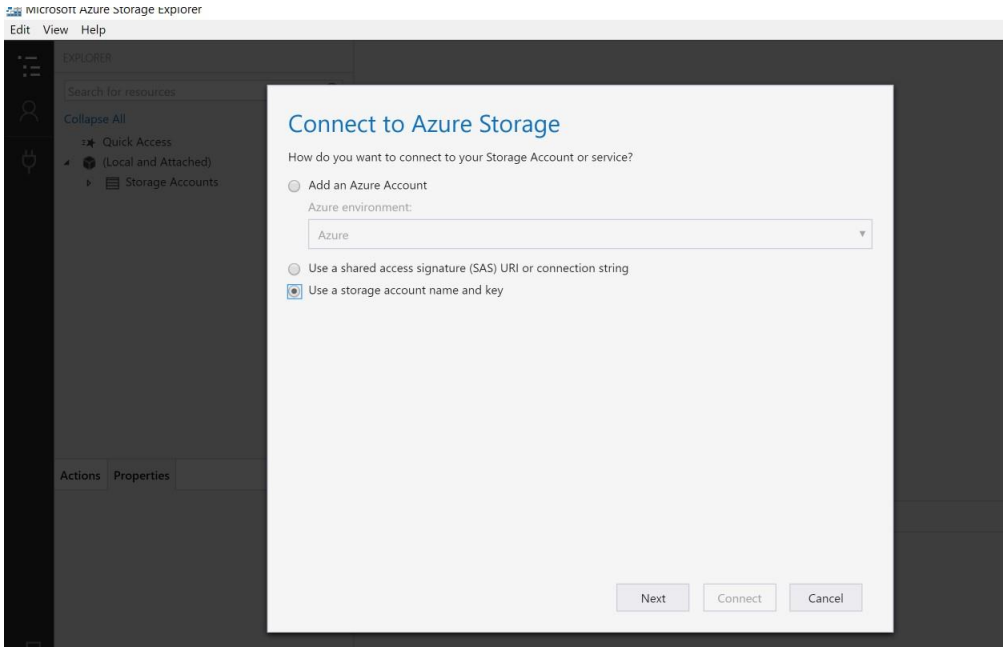
8. From the download page scroll down and find Azure Storage Emulator, click Install and download and install Azure Storage Emulator

Azure Storage Emulator

Install

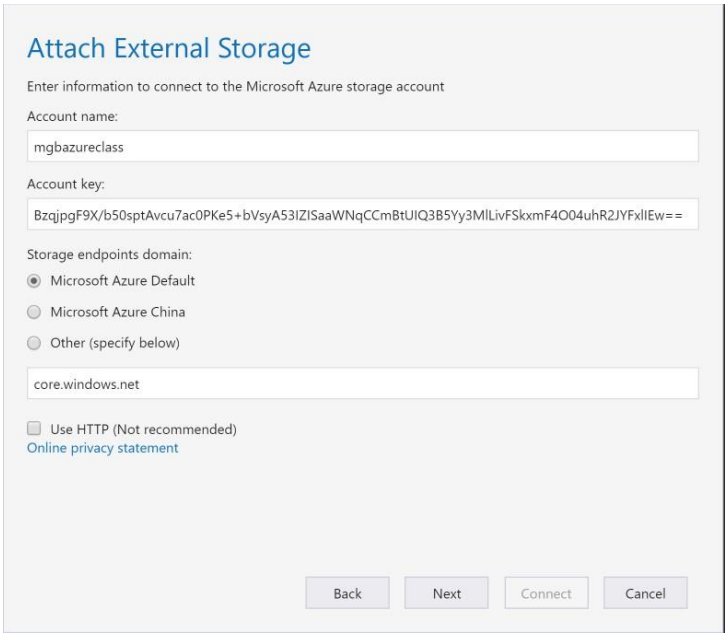
Documentation

9. Once Storage Emulator has installed in should launch, if it doesn't launch storage emulator and you should see a screen similar to the one below asking you to connect to Azure storage.



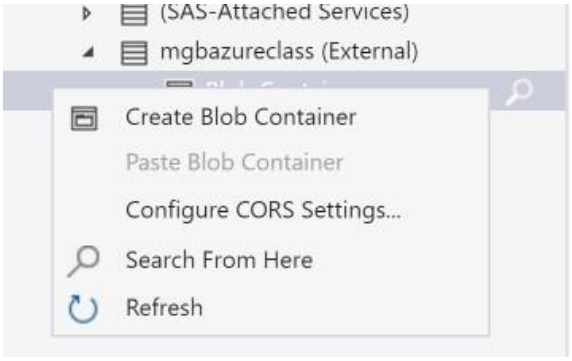
If the screen above doesn't appear, click on the icon that looks like a plug.

10. Choose the Use a Storage Account Name and Key radio button and then click Next
11. On the Attach, External Storage screen, in the Account Name type the name of the storage account you created earlier and in the Account Key section paste Key1 that you copied earlier. Then click Next and then Connect on the next screen.



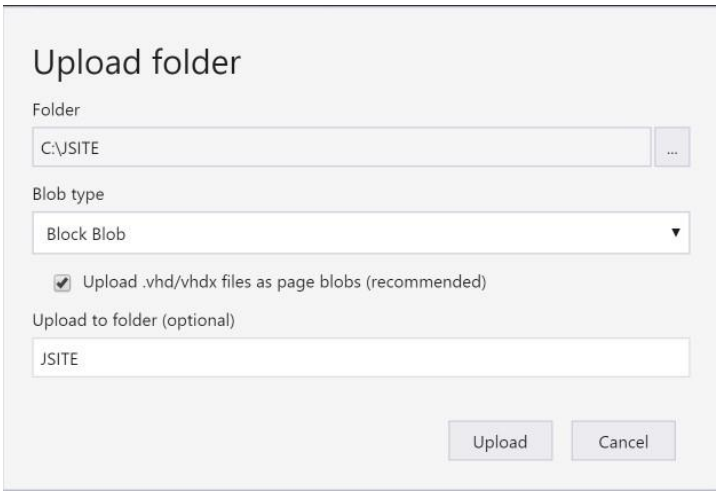
You should now be connected to your newly created Storage Account.

12. In Azure Storage Explorer navigate to your storage account and then click Blob Containers.
13. Right Click Blob Containers and the select Create Blob Container

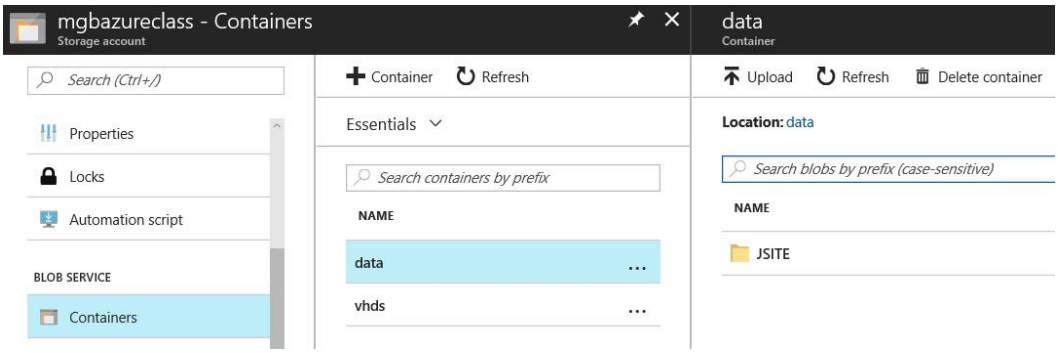


14. Create a new Blob Container called data and a new Blob Container called vhds

15. Select the newly created data container and select upload and select to upload folder. Navigate to drive c:\ and select the jsite folder and select Upload



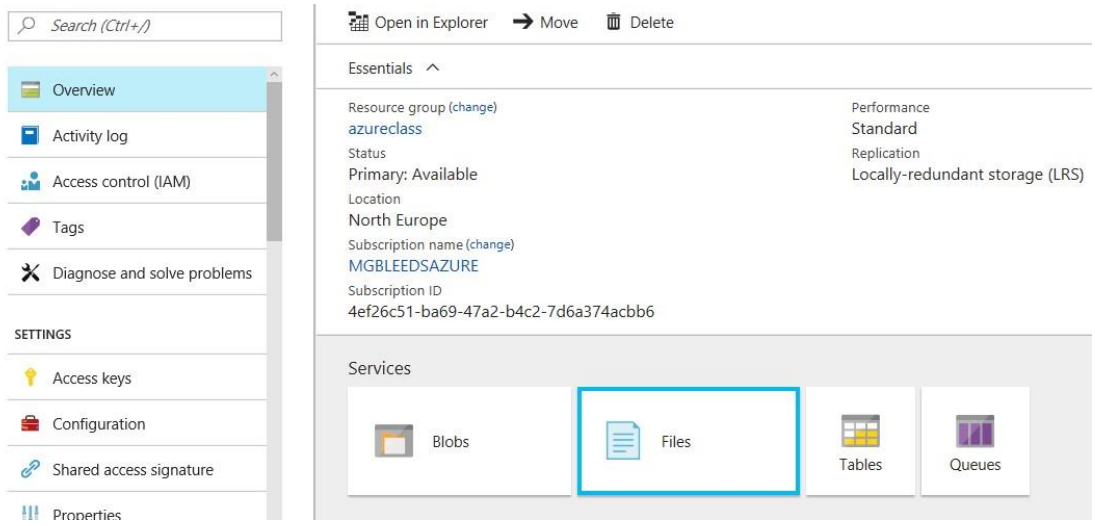
This will upload the jsite folder to your data container. If you navigate back to the Azure portal you should be able to see your newly created container and the jsite folder in the data container.



Exercise 2 Working with File Storage

In this exercise you will work with Azure file storage to create a file share and connect to a file share from the local PC.

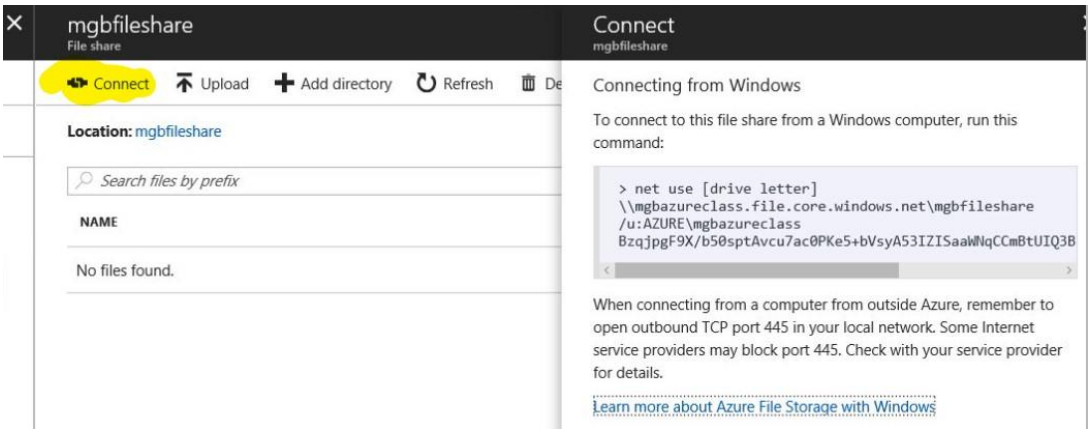
1. Connect to your Azure portal and navigate to your storage account and Files storage



2. From Files select 

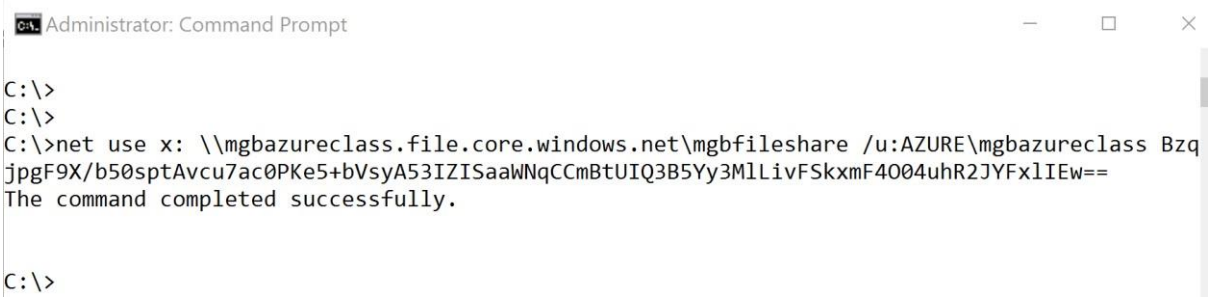
to create a new file share.
3. Give your fileshare a name of xxxshare (where xxx is your initials) and then click OK to crate your file share.

4. Select your file share and then click Connect,



5. You will be using the details in the Connecting from Windows section. Open a CMD prompt locally and use the NET USE command to connect to the fileshare. The Key

used to connect is Key1 that you copied earlier in exercise 1. Below is an example of how the command will look in the command prompt:



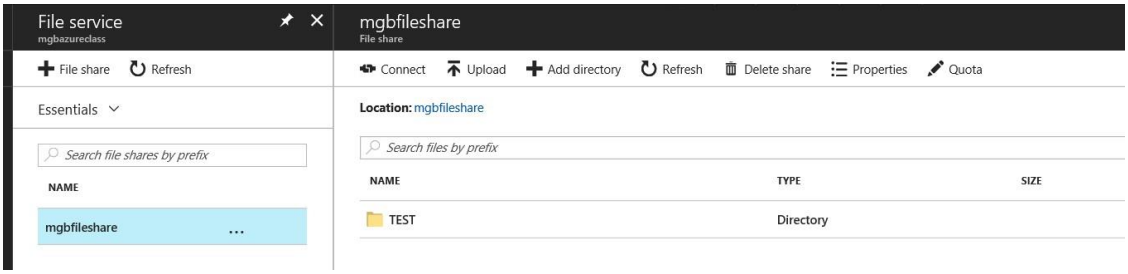
6. While still in the command prompt type X: to navigate to the newly mapped X Drive

```
C:\>X:

X:\>
X:\>
```

7. Then use the following to make a directory in your file share: MD TEST

```
X:\>
X:\>MD TEST

X:\>
X:\>
```

8. If you refresh your fileshare in the portal you should see the folder TEST



9. From the command prompt use the following to remove the mapping to your fileshare: NET USE X: /DELETE.

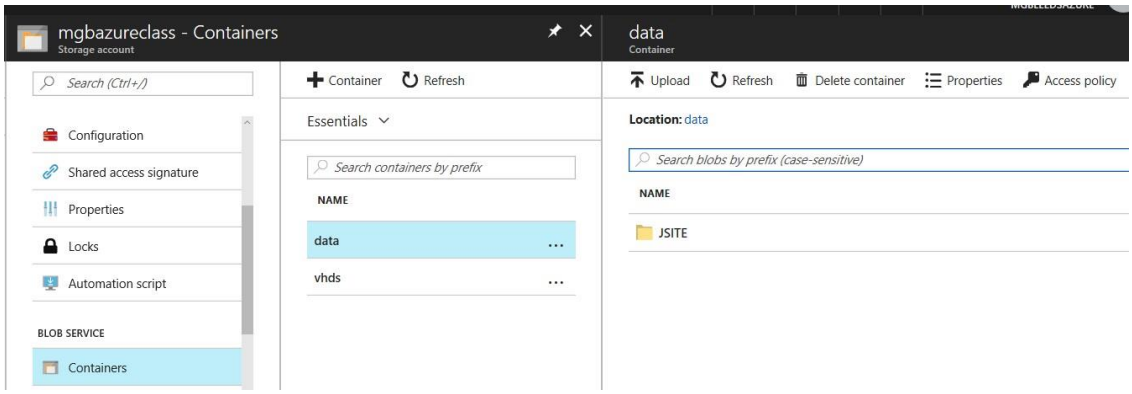
```
C:\>
C:\>NET USE X: /DELETE
X: was deleted successfully.


C:\>
C:\>
```

Try navigating to X:, this should fail because you have removed your mapping.

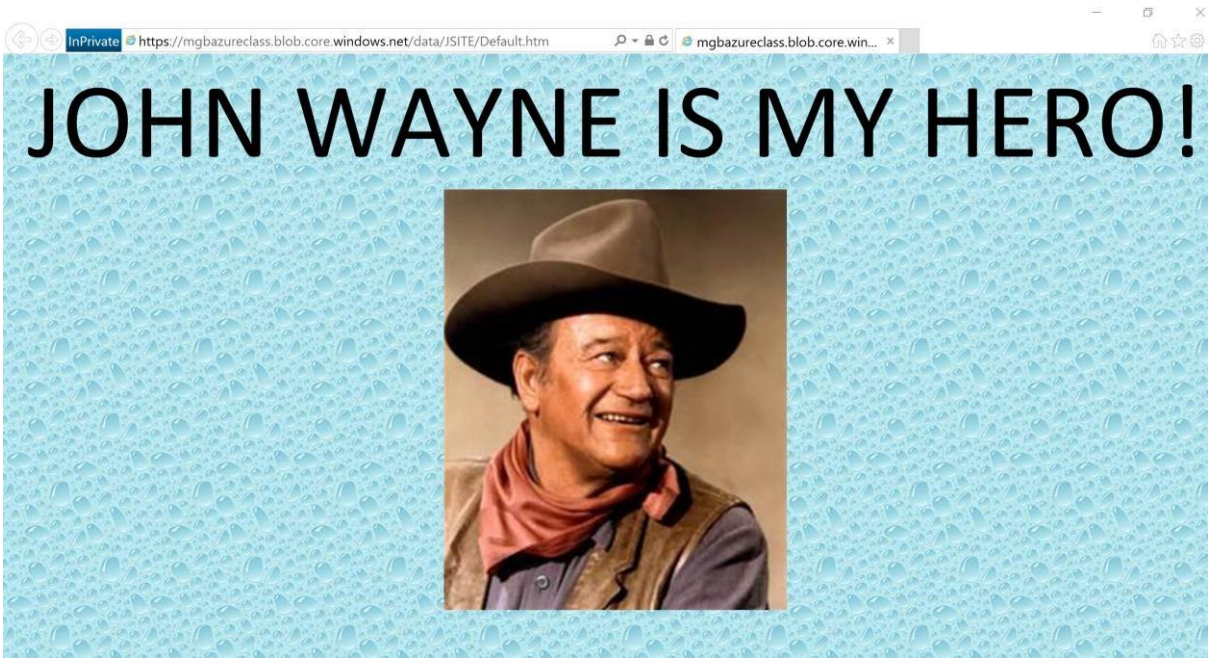
Exercise 3 Granting Access to a Blob Container

In this Exercise, you will change the Access Policy of a Blob container to allow anonymous read access.

1. Navigate to your storage account, Blob service and select the data container you created earlier.



2. With tour data container selected click  and change the access policy from Private to Blob and then click Save (This will grant anonymous read access to the container)
3. Select the data container and then select the jsite folder. Inside there you should find a file called Default.htm. select it and take a copy of its URL
4. Using IE open an InPrivate Browsing session (or use chrome and open and InCognito) session
5. Past in the URL for the Default.htm document. You should be connected to the document.



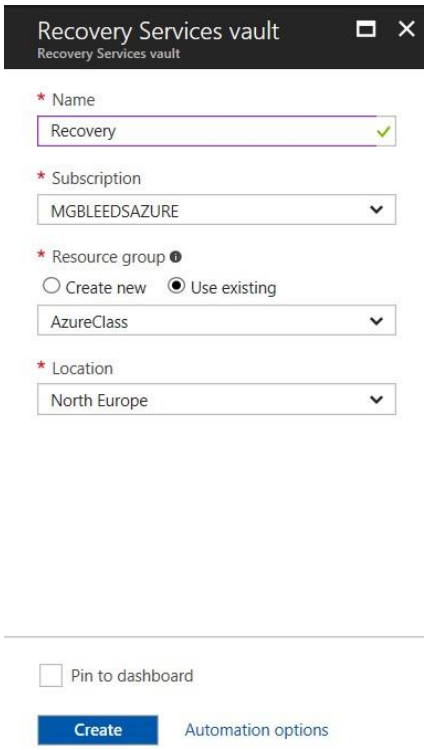
Try changing the Access Policy back to Private, close the IE InPrivate Browsing session and reopen a new one and try to connect to the URL again. Now the connection should fail.

Exercise 4 Working with Azure Backup

In this exercise, you will use Azure Backup to protect a VM running in Azure

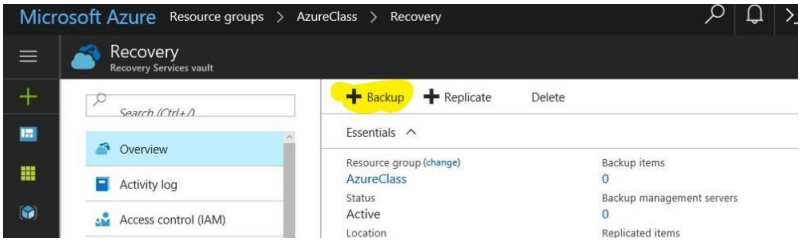
1. From the AzureClass resource group click 

and then in the search box type backup and choose Backup and Site Recovery (OMS) and then click create
2. On the Recovery Services Vault blade type a name of Recovery then click create

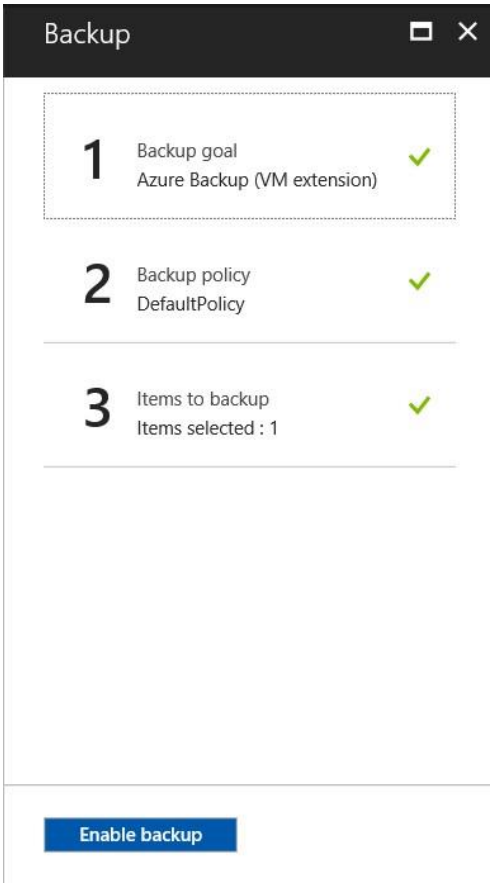


It will take a few minutes to create your new recovery services vault

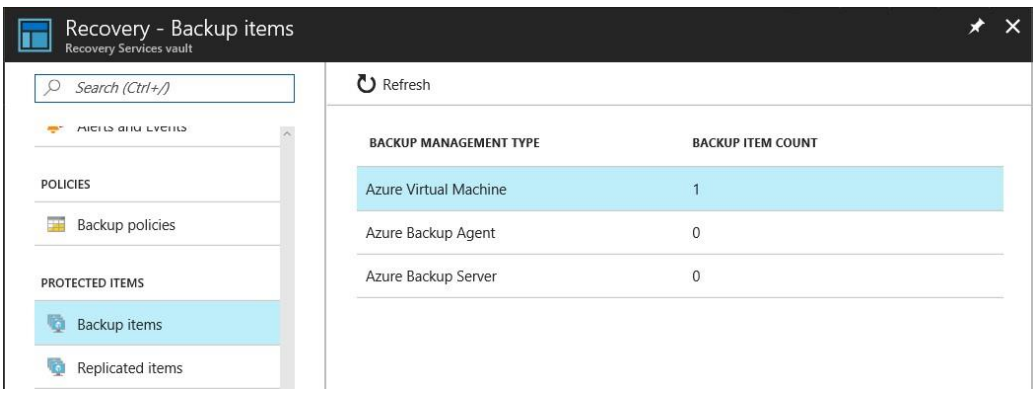
3. Once created navigate to your newly created recovery services vault and click



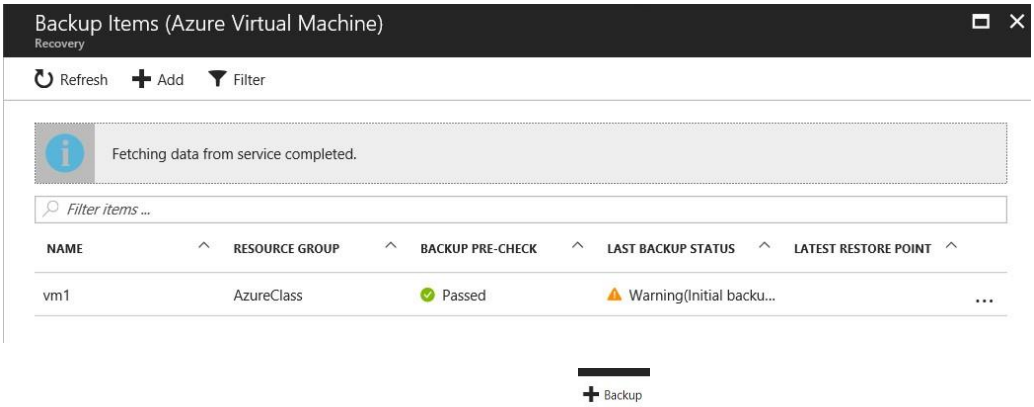
4. On the Backup Goal Blade Make Sure **Azure** is selected in *the where is your workload running?* Box and that **Virtual Machine** is selected in the *What do you want to back up?* Box. Then click OK
5. In the backup Policy bladed make sure *DefaultPolicy* is selected then click OK
6. On the Select Virtual Machine Blade choose VM1 then click OK
7. Finally with all three backup steps completed click Enable Backup



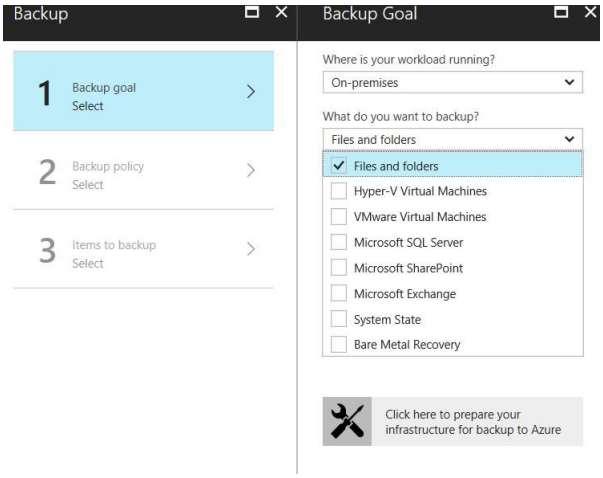
It will take a few minutes to enable protection on VM1. Once protected you can see the status of the protected VM by navigating to Backup items from the Protected items section of your backup vault. You should see that 1 item is protected.



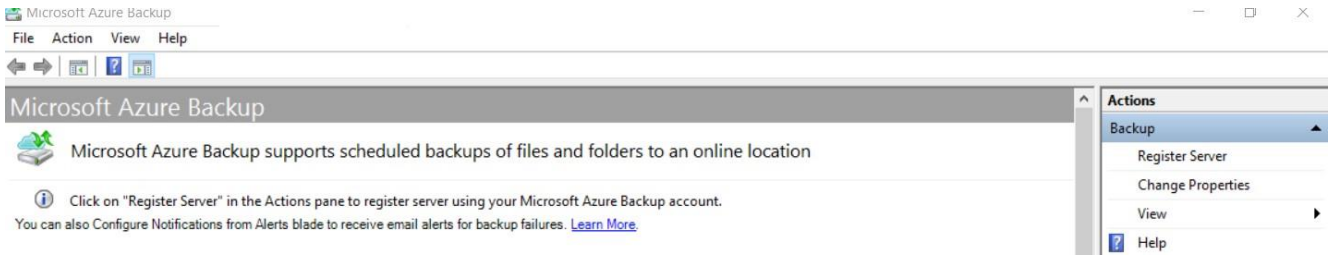
If you click Azure Virtual Machine you should see VM1 is being protected.



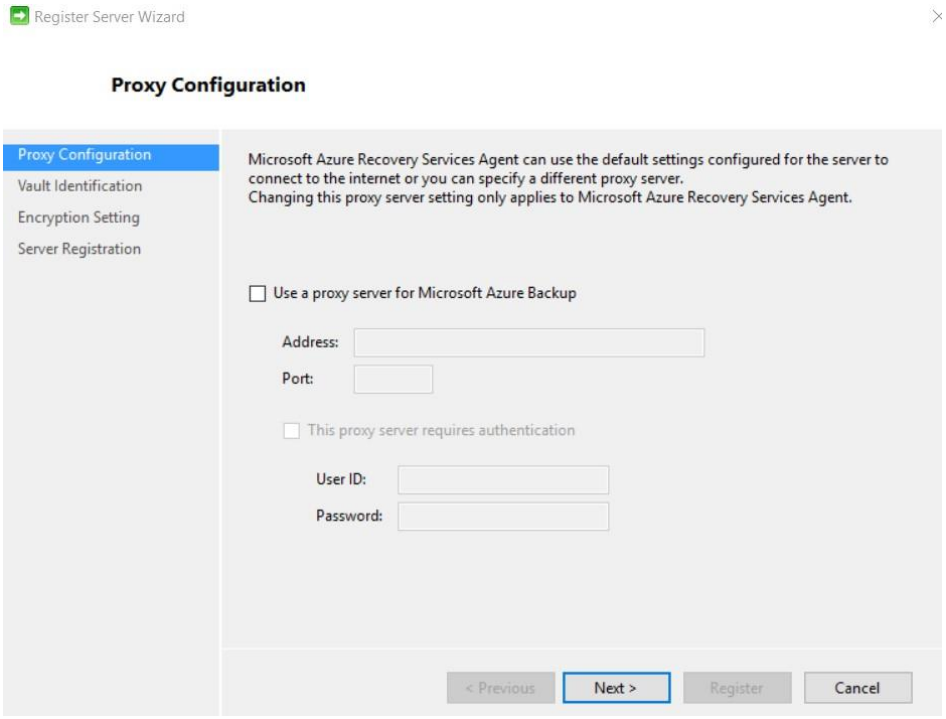
8. Go back to the overview page and click
9. This time in the *where is your workload running?* Box select On-Premises and the click Files and Folders in the What do you want to Backup? box



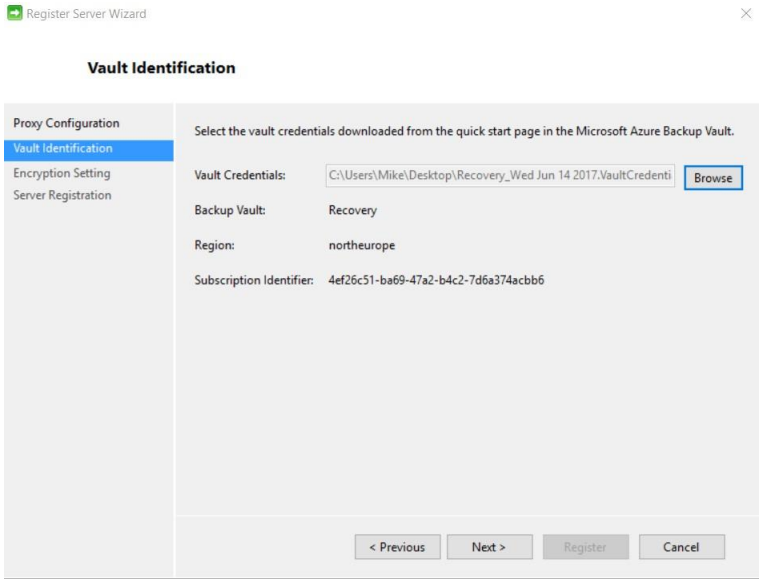
10. Next select the “click here to prepare your infrastructure for backup to Azure” box.
11. Download the Vault credentials, you will need them later
12. Download the Agent for windows server or client and install It locally
13. Once installed you must register the services with your backup vault. If the backup application doesn’t open, open it and click on Register Server



14. To register the backup application, you will need the Vault Credentials you download earlier. Click Next on the Proxy Configuration screen



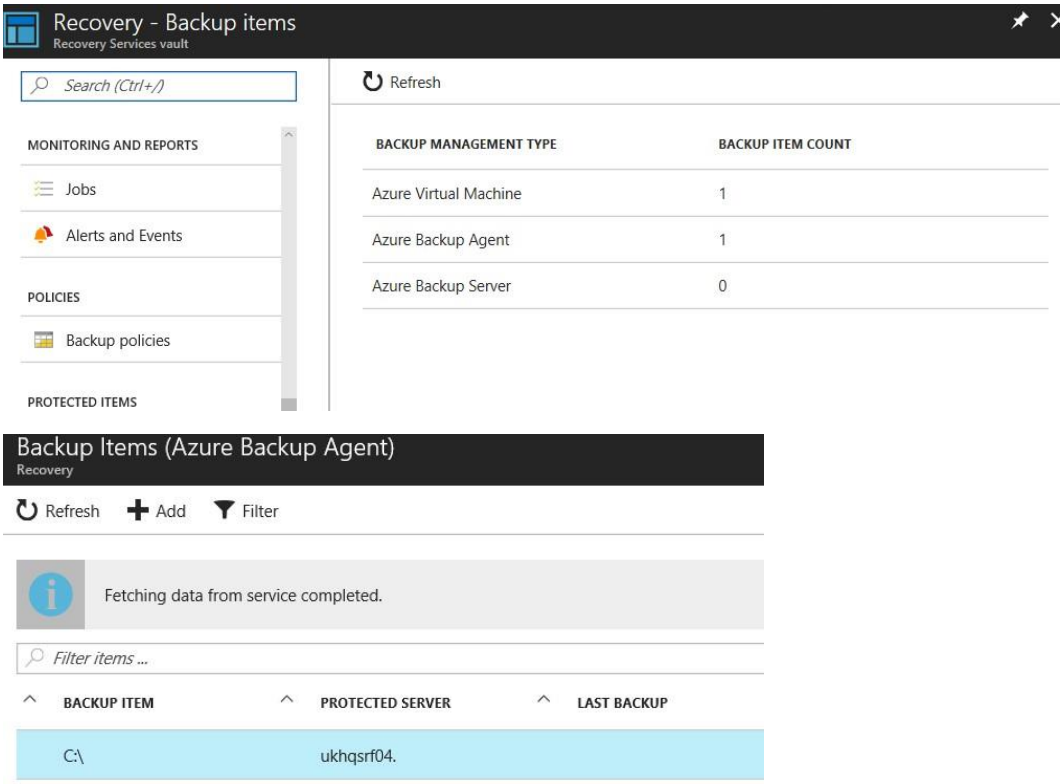
15. On the vault identification page click browse and navigate to the vault credential file you downloaded earlier. It might take a few minutes to validate the credentials but once finished you should see a screen like the on below:



16. Click Next then click Generate a Passphrase on the Encryption Setting page, click browse to save the file locally. Then click Register.

It will take a few minutes to register the server.

17. Now you have registered the server with the Backup Vault you can use the Schedule Backup wizard to backup some files to Azure. Use it to backup c:\jsite
18. When ready go back to the Azure portal and the backup Items section, you should see that you have one backup agent registered and if you click on that you should see the name of the server protected.



Lab 5 Identifying Elements of Azure PaaS Cloud Services

Azure provides several Services that fall under the PaaS heading, PaaS Cloud services are one of these. Today you are unlikely to deploy new services / applications using PaaS Cloud services instead you will probably use Web Apps and Web Jobs. Still you might come across PaaS cloud services in existing Azure Deployments or when migrating from Classic to ARM deployments. It is also important to be able to identify PaaS cloud elements for the 70-533 exam.

Exercise 1 Describing Cloud PaaS Elements

1. What elements would expect to find in the ServiceDefinition file?

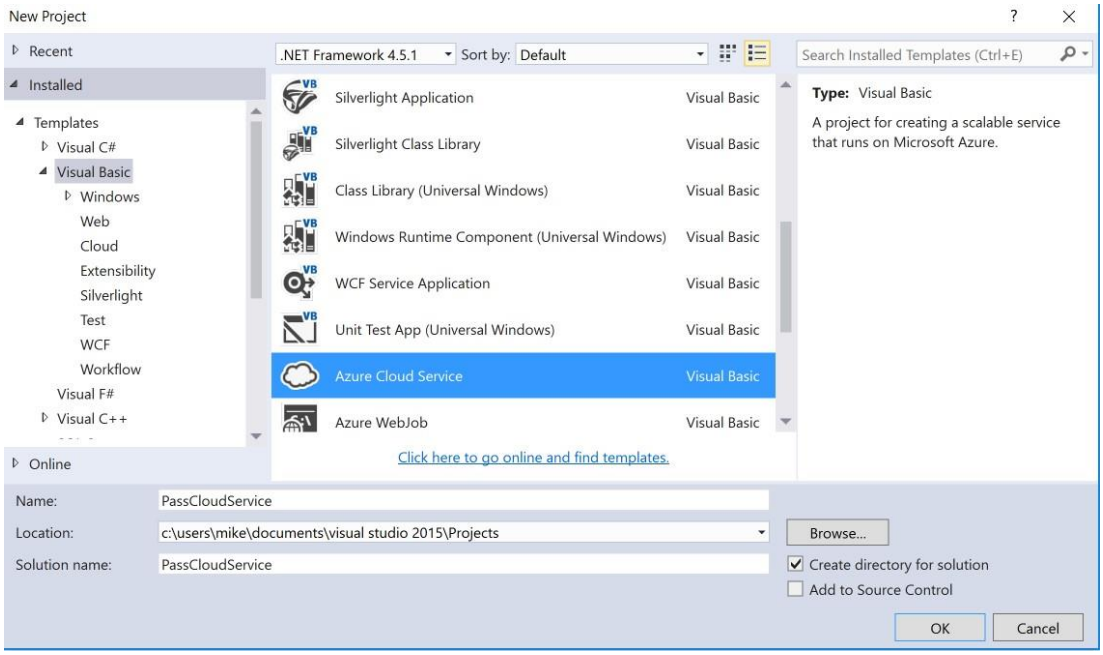
2. What elements would you expect to find in the ServiceConfiguration file?

3. Describe what the Web and Worker roles do?

Exercise 2 Deploying a Pass Cloud Service

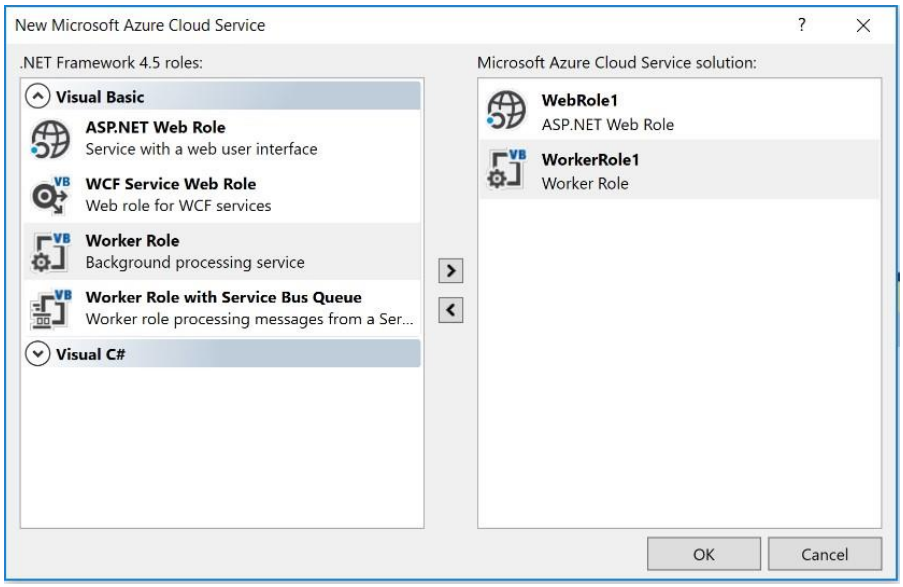
1. Open Visual studio and using Server explorer connect it to your Azure subscription. If you are unsure how to do this get your instructor to demonstrate how.

2. From the file menu choose New > Project

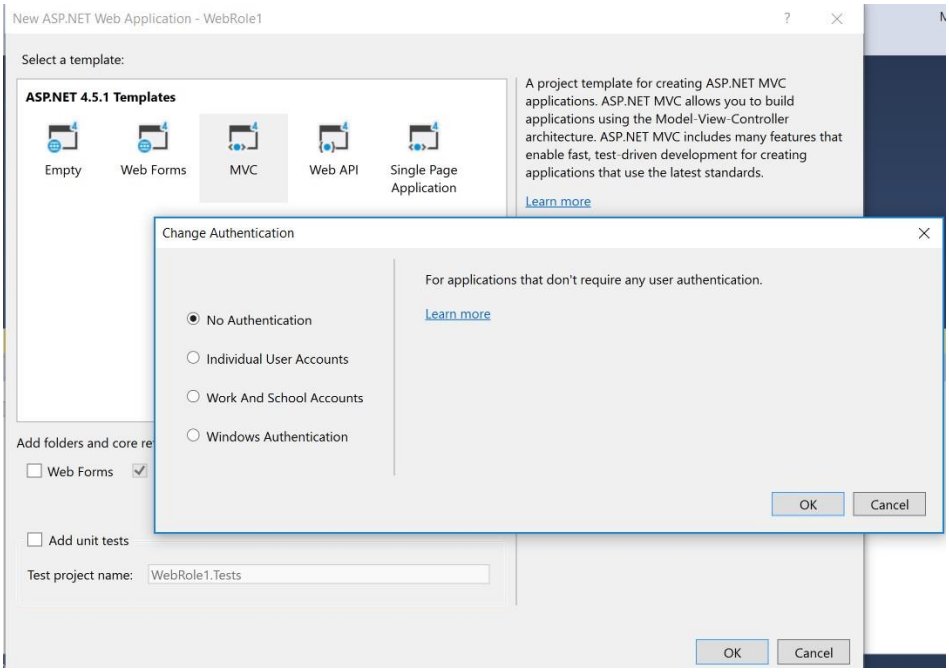


Select visual basic and Azure Cloud Services, give you project a name like PaaSCloudService and then click OK

3. On the next screen make sure you add both the ASP.NET Web Role and the Worker Role to your solution. Then click OK

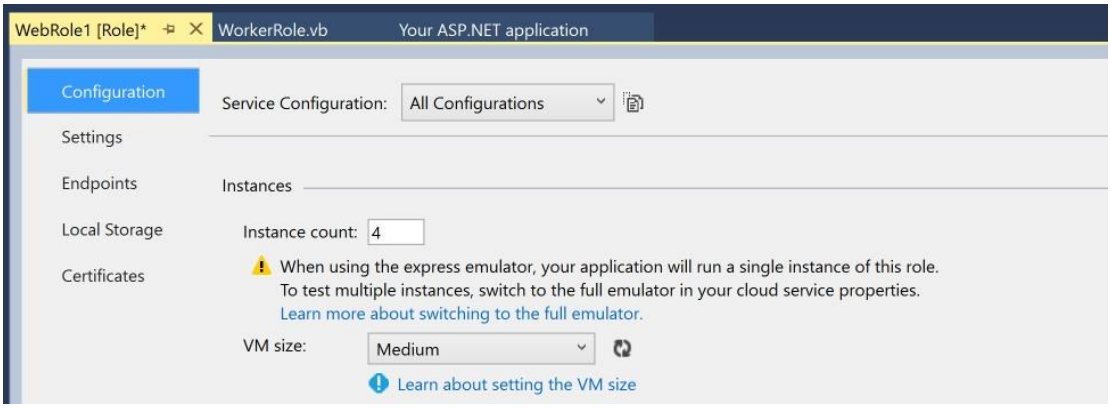


4. On the New ASP.NET Application screen make sure the MVC template is selected then select Change Authentication and make sure No Authentication is selected. Click OK and then OK again. Your project will now be created.

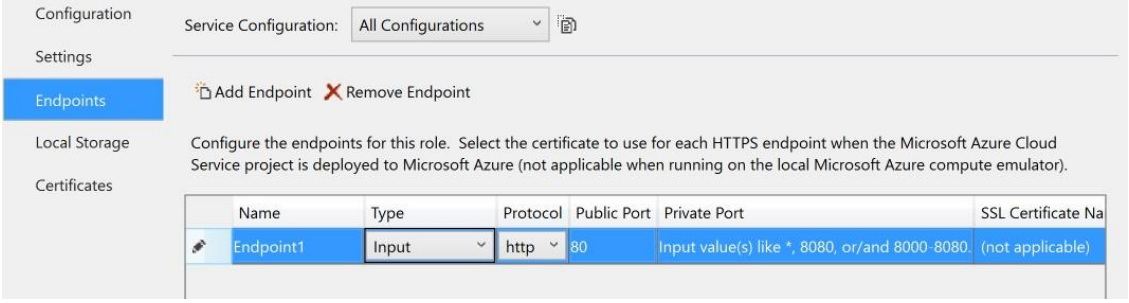


5. Once your project is created, from the view menu choose Solution Explorer. This will open the solutions explorer pane.
6. Using Solution explorer under the roles section double click WebRole1. This should open a page where you can configure the WebRole.

7. Using the WebRole1 page change the instance count to 4 and the VM Size to Medium.

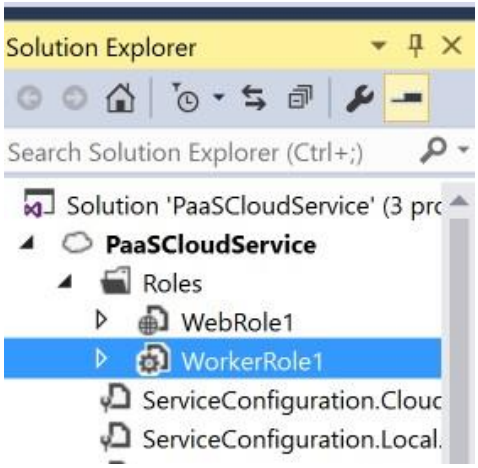


8. Click on the Endpoints section and review the endpoints

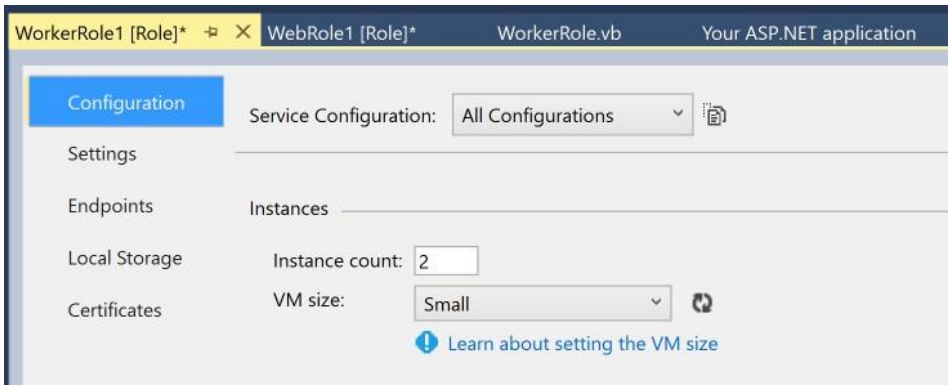


What are the differences between Input endpoints, Internal Endpoints and Instance Endpoints?

6. From solutions explorer double click WorkerRole1. This will open the WorkerRole pane.

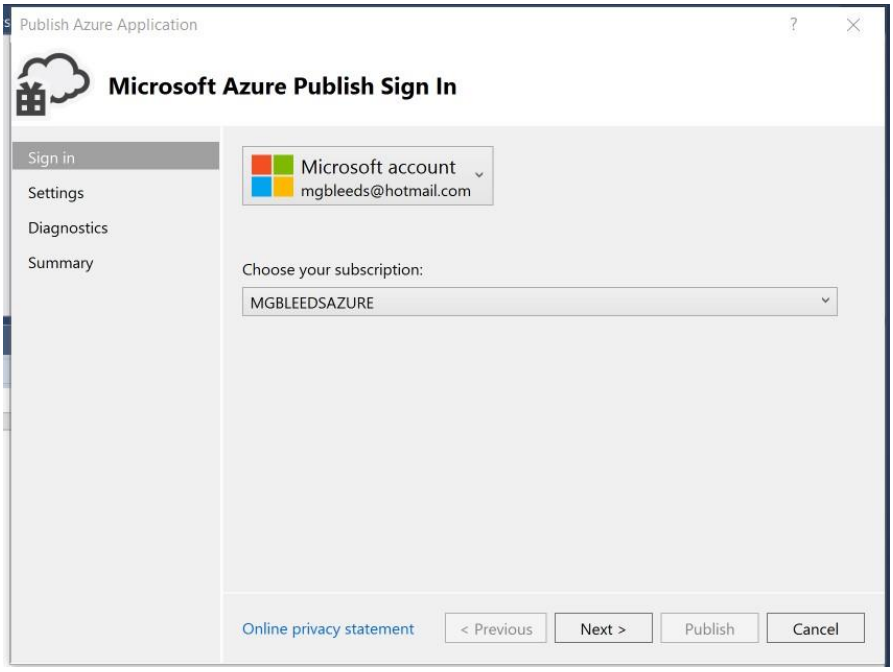


7. On the Worker role pane change the Instance Count to 2

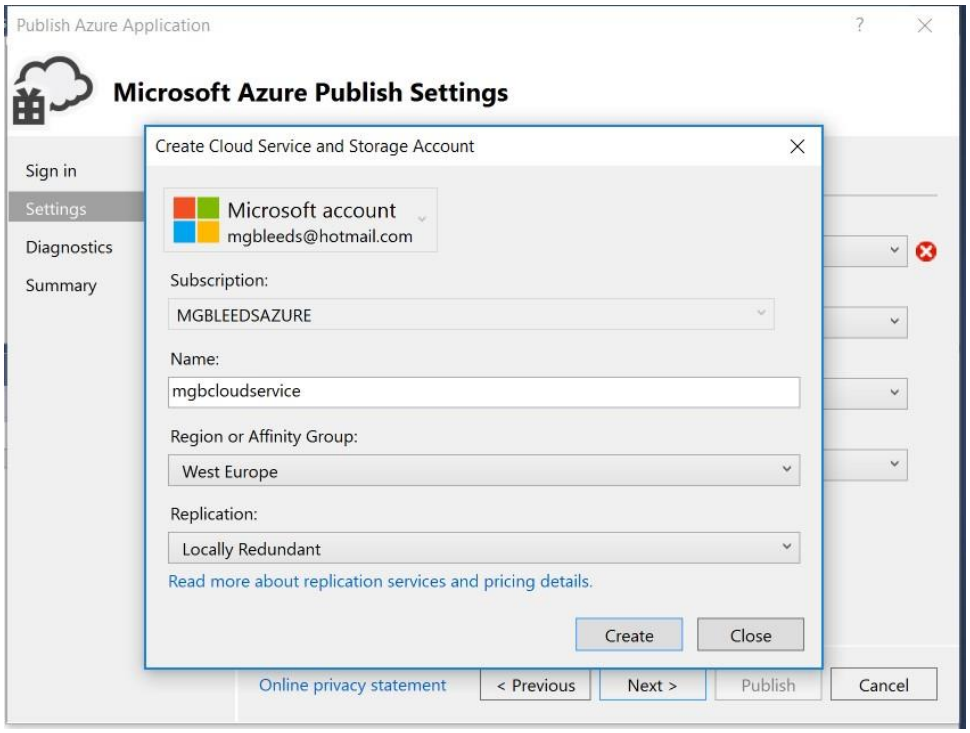


We are now ready to deploy our PaaS cloud service.

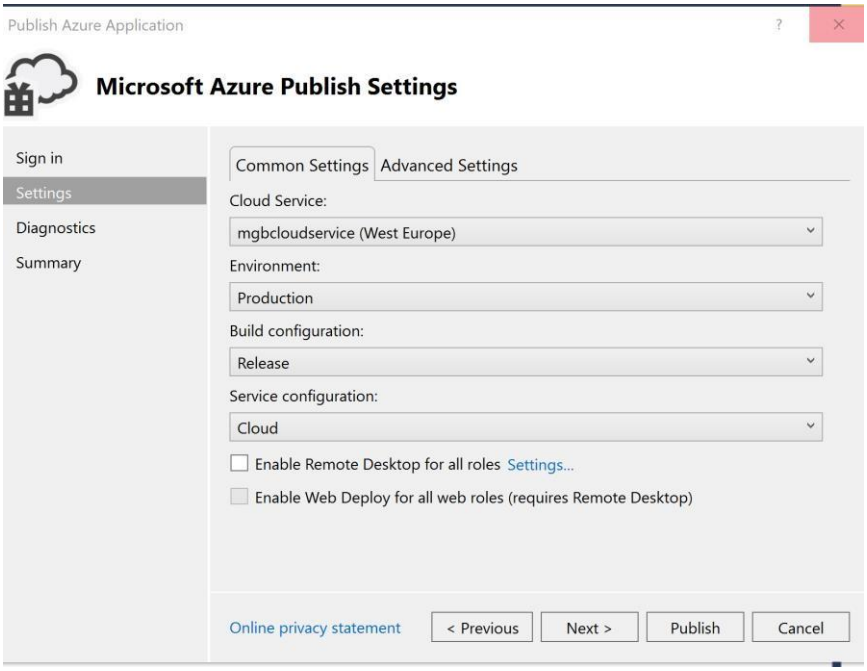
8. Right click the name of your solution and select Publish. On the Microsoft Azure Public Sign In page make sure your subscription is selected then click next.



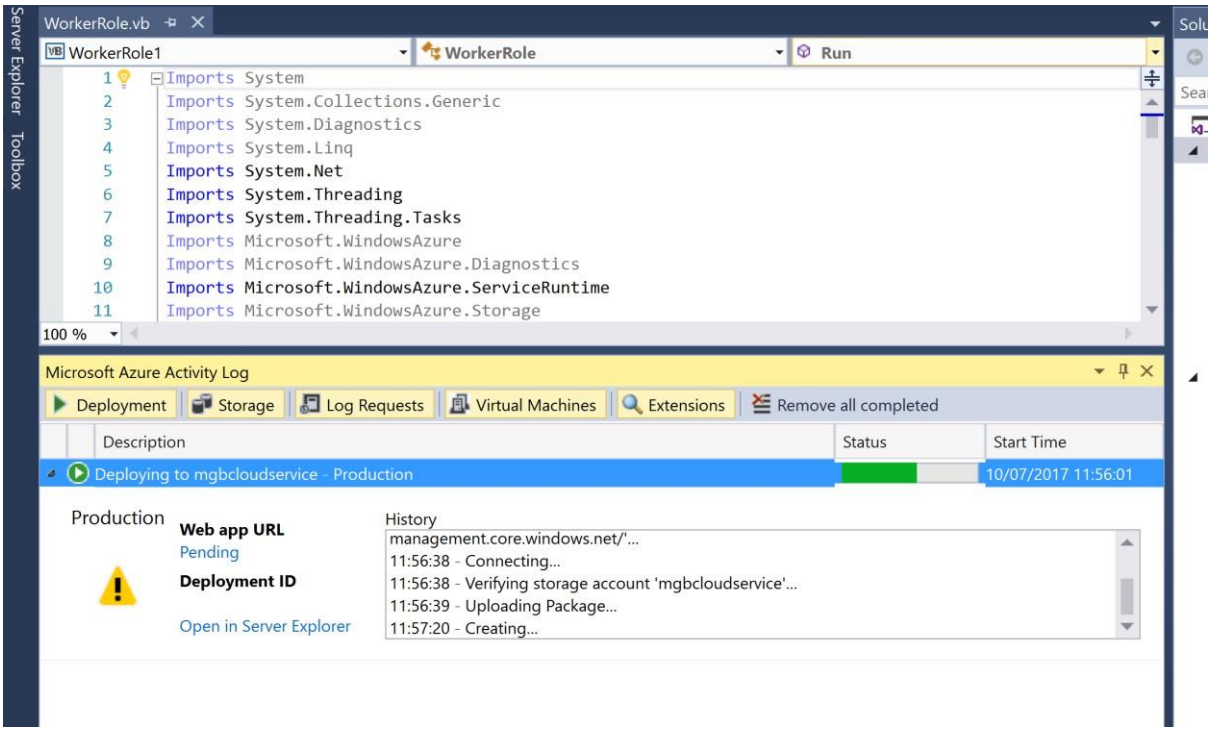
9. On the Create Cloud Service and Storage Account page provide a name for your cloud service, choose **West Europe** for the region and **Locally Redundant** for the replication type. Then click create.



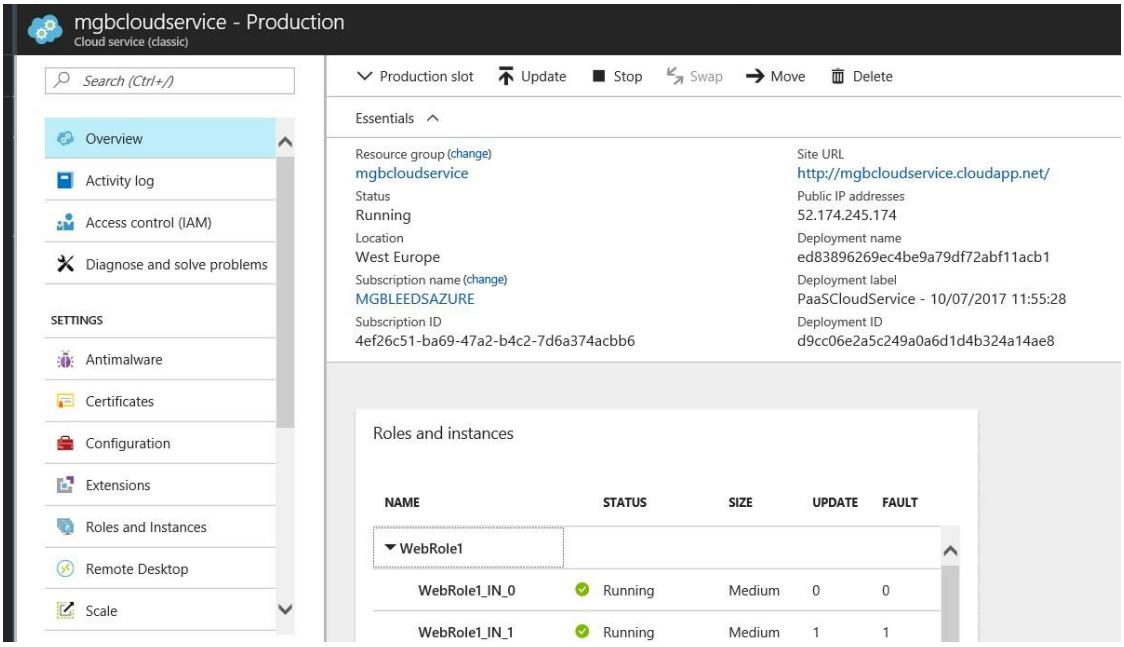
10. Once the cloud service is created click Publish



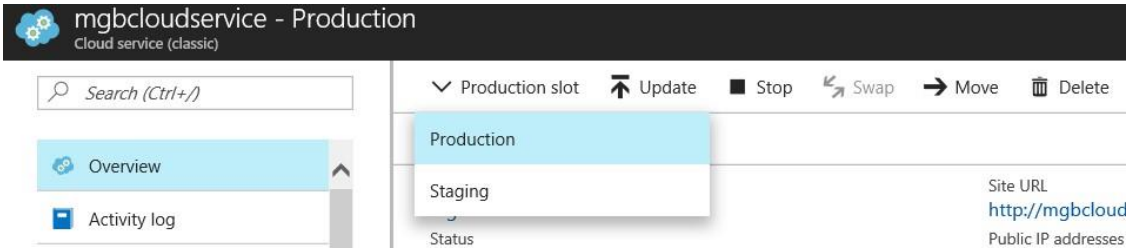
It will take some time to create your new Cloud Service. You can monitor the progress of the deployment through visual studio:



11. Once your cloud service is deployed login in to the Azure portal. You should find that a new resource group has been created with the same name as the deployed Cloud Service. Navigate to it and then select your cloud services.



12. Find the Site URL and click on it to open your new website.
13. Back in the Azure portal what is the difference between the Production and Staging slots?



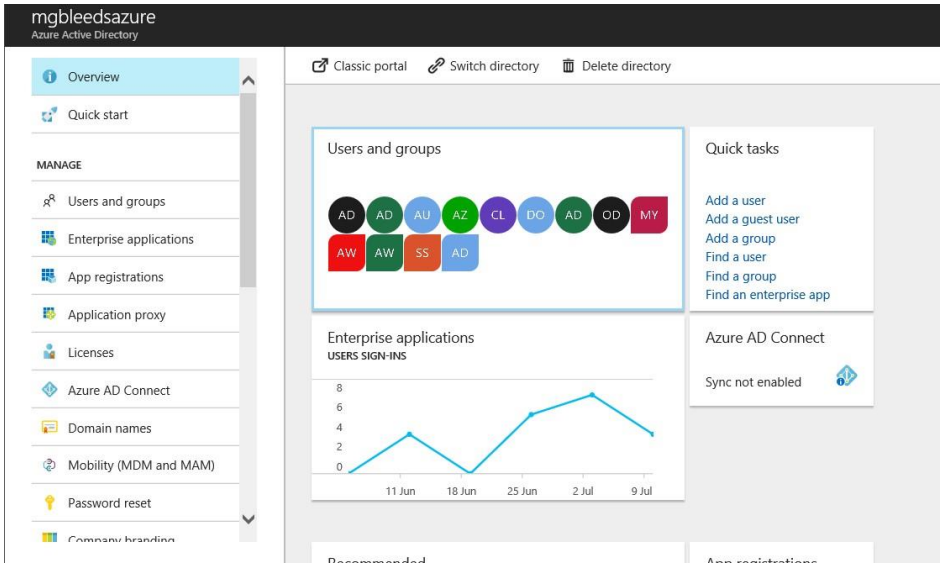
Spend some time looking at the different section of the Cloud service, when you have completed your investigation delete resource group that hosts this cloud service. By deleting the resource group, you also delete the cloud service.

Lab 6 Working with Azure AD

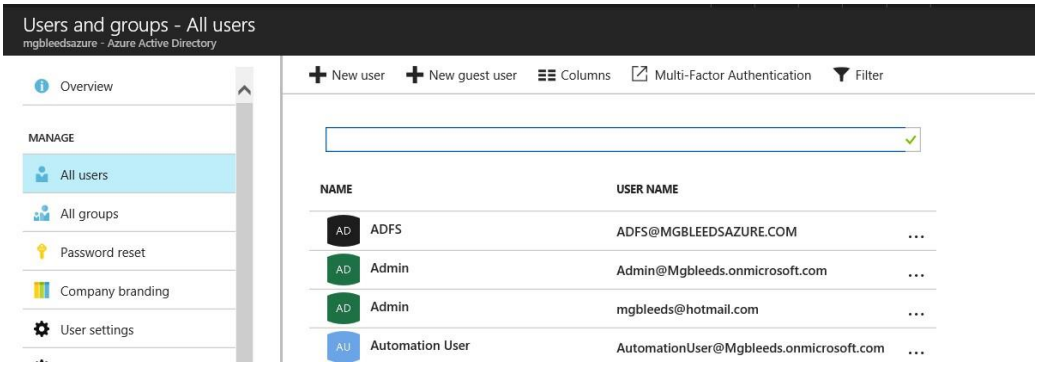
In this lab you will be working with Azure AD, you will be creating cloud only accounts and using RBAC to assign access to a recourse group. You will integrate an application with Azure AD and look at adding a custom domain.

Exercise 1 Working with Azure AD users

1. Login in to the Azure portal and access the Azure Active Directory section. You should see a screen similar to the one below.



2. Click on the Users and Groups section and then click on All Users



3. In the All Users section select 

4. Create a new user with the following details

Name: User1

User Name: User1@YOURDOMAINNAME.onmmicrosoft .com (example user1@mgbleeds.onmicrosoft.com)

Profile: Fill in some details

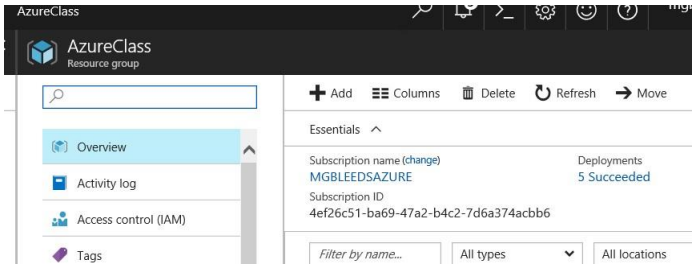
Properties, Groups and Directory roles leave as default

Password: check the show password box and copy and paste the auto-generated password to notepad.

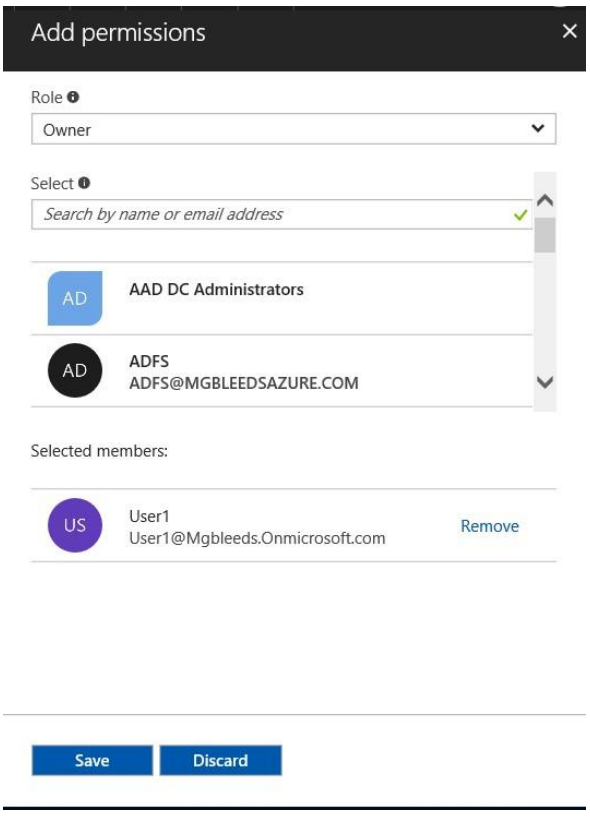
5. Click Create
6. Using either an IE InPrivate Browsing session or chrome Incognito session one a browser and go to: portal.azure.com and login as User1@YourdomainName.onmicrosoft.com. You will need the password you created earlier, you will be prompted to set a new password.
7. If prompted fill in the additional security information to sign in.
8. You will be logged in to the portal, navigate to resource group section.

Can you see the resource group? If not, why not?

9. Navigate back to the browser where you are logged in as an admin,
10. Navigate to the AzureClass resource group and then access the Accedes Control (IAM) section



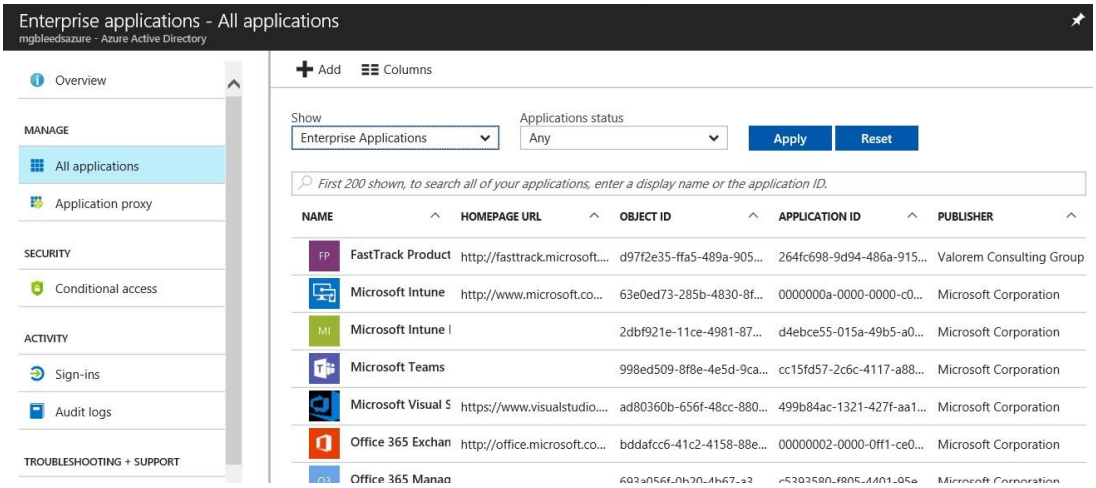
11. In the Access Control (IAM) Section click
12. In the Add Permission pane choose Owner as the role and make sure User1 is added as a selected member and click Save.



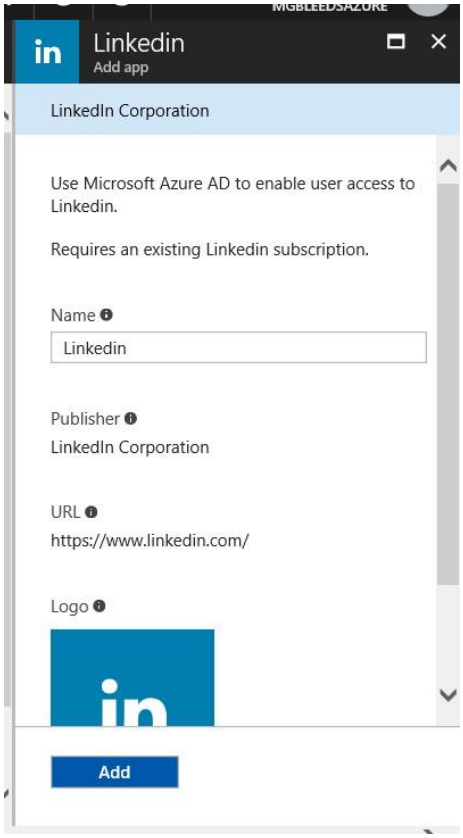
13. Once the now role assignment has been saved, navigate back to the browser where User1 is logged in and sign out and sign back in. navigate to the resource group section. Can you see the AzureClass resource group now?
14. Log User1 out of the portal.

15. Switch back to the browser where you are logged in as an admin and navigate to the then Azure AD section and the select Enterprise applications.
16. On the Enterprise Applications section choose all applications and then click

    + Add



17. On the Categories pane navigate to social and find the LinkedIn app and select it.
18. Leave everything at their defaults and click Add.

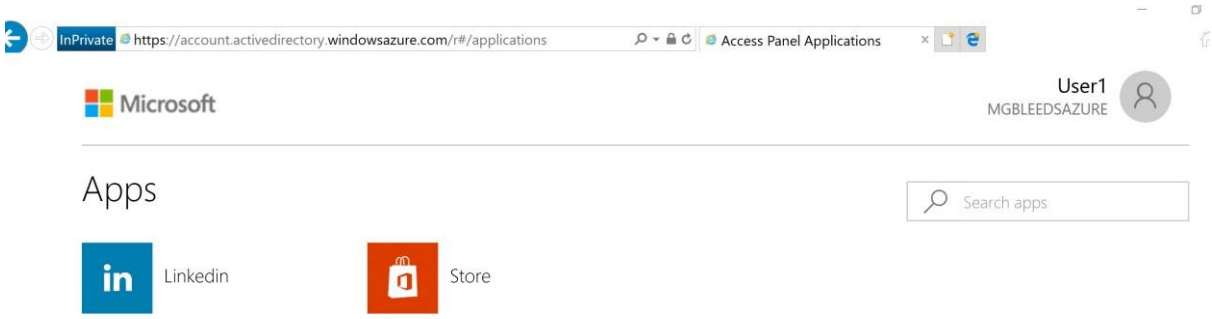


Once added the LinkedIn – Quick Start page should open

19. Using the Assign a User for testing section of the quick start Assign user1 to the app.

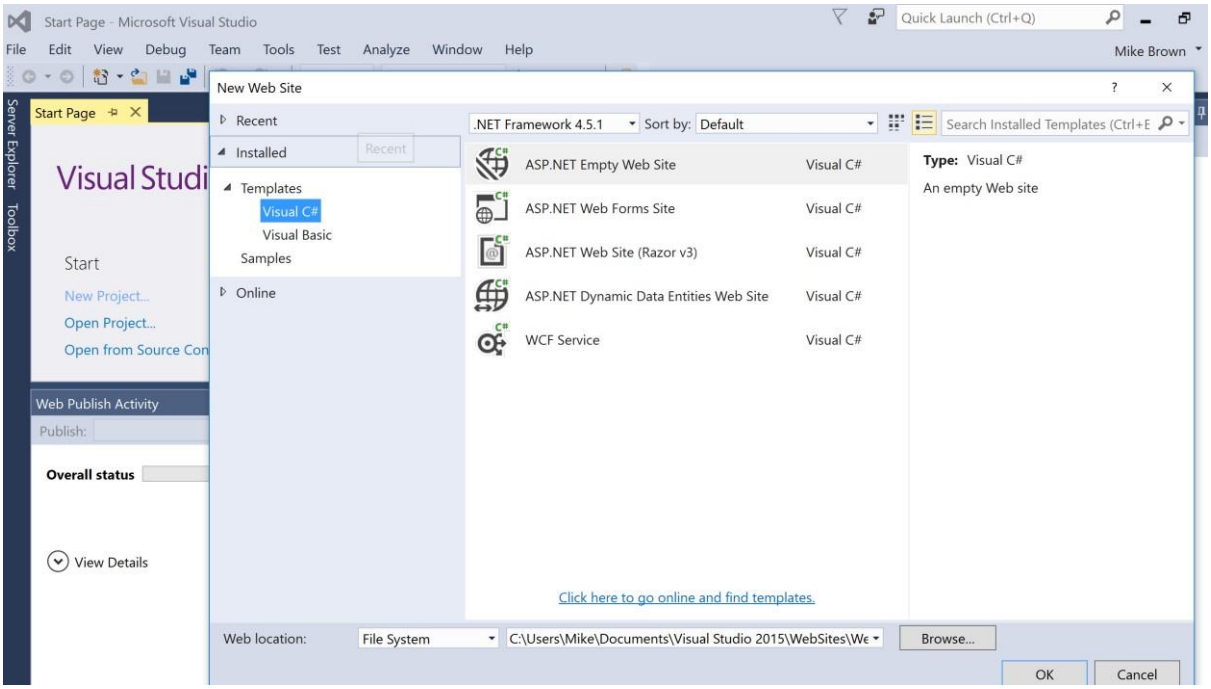
20. Using the Configure single sign-on section of the Quick Start make sure Passwordbased Sign-on is selected then click Save.
21. Using either an IE InPrivate Browsing session or chrome Incognito session one a browser and go to: http://myapps.microsoft.com
22. Login as user1

You should see LinkedIn as an application available to you

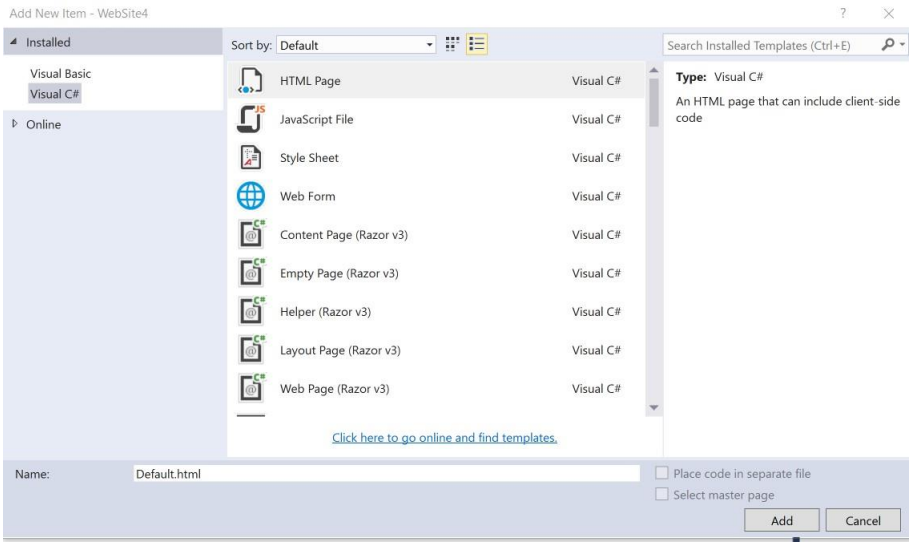


Exercise 2 Deploying a web app and integrating it with Azure AD

1. Start Visual studio and from the file menu choose new web site. Select Visual C# and then select ASP .NET Empty Web Site and click OK

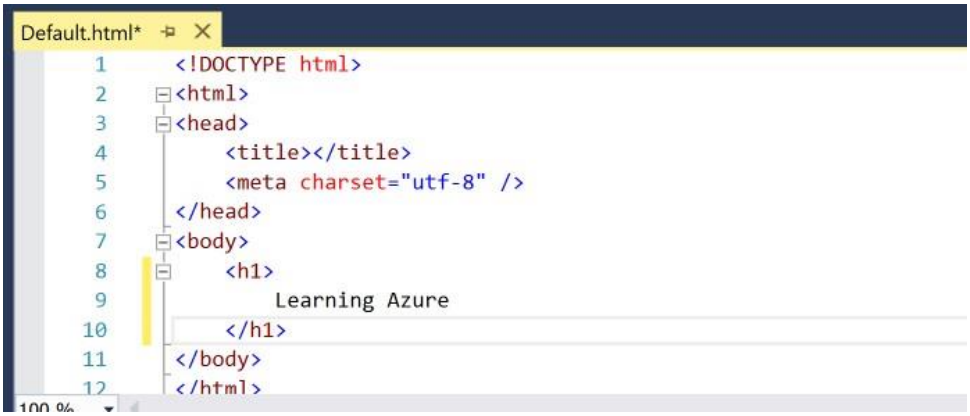


2. Once created right click the name of your web site name and choose add and then Add new Item and choose HTML Page, in the name box change the page name to Default.html then click ADD



3. Add some text between <body> </body> for example:

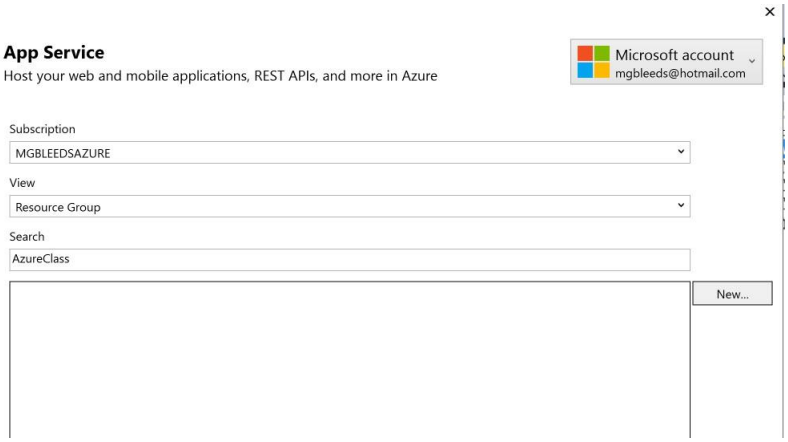
<h1> Learning Azure </h1>



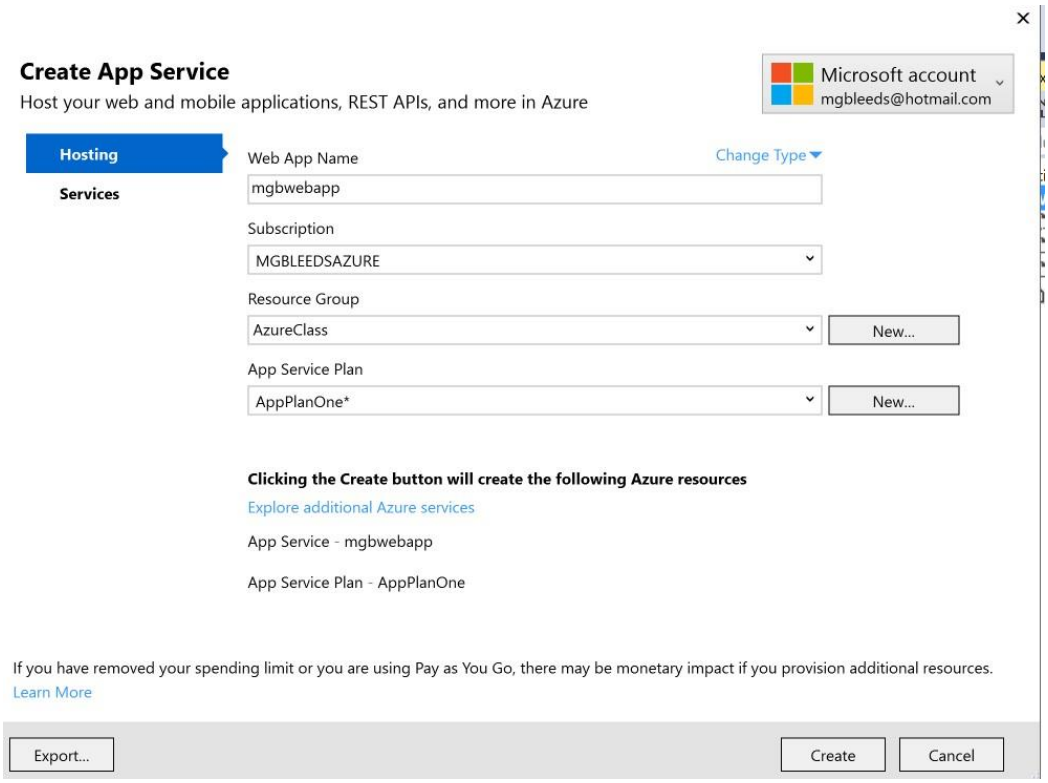
4. Click Save to save the changes to your page 

Next, we will publish the site to Azure and integrate it with Azure AD

5. In solutions Explorer right click the name of your site and select Publish Web APP
6. On the Publish page select Microsoft Azure App Service as the public target
7. On the next page choose your subscription and choose the AzureClass Resource group then click NEW

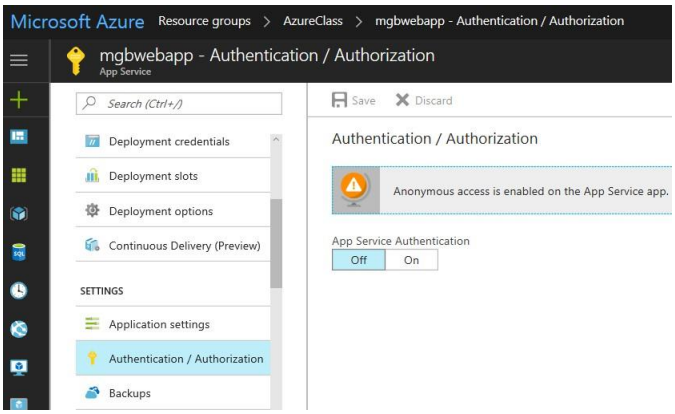


8. Create the new app service with the following settings:
   Web App Name: ***yourinitals*WebAPP** (for example mgbwebapp)
   Resource Group: AzureClass
   App Service Plan: Click New and create an App Service Plane with the following settings:
   App Service Plan: AppPlanOne
   Location: North Europe
   Size: S1 (1core 1.75 GB RAM)
9. Once you have entered all the details click create

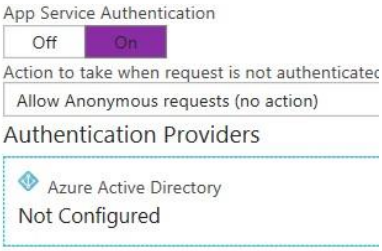


10. On the Publish page select Next, the next again then publish.
11. Once published navigate to the portal and go to the AzureClass resource group and select your newly created Web App.

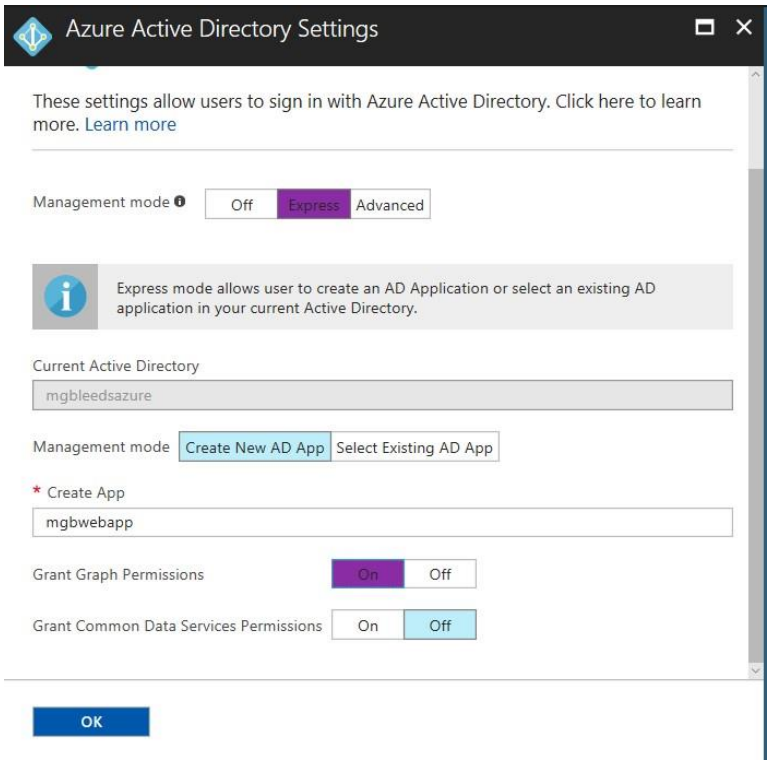
12. On the properties of your web app navigate to the Authentication / Authorization section and select ON



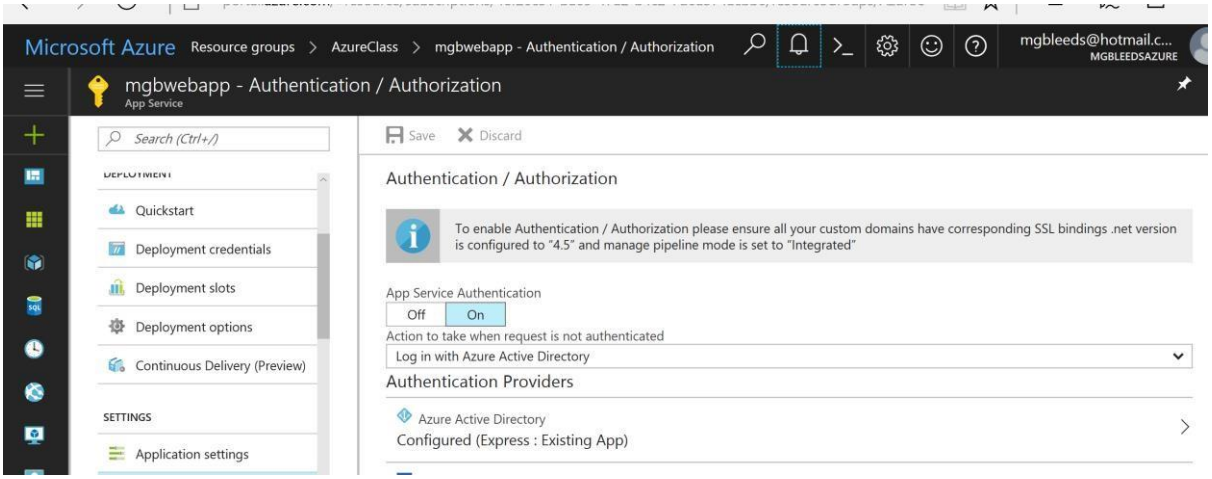
13. once authentication is turned on select Azure Active Directory Not Configured



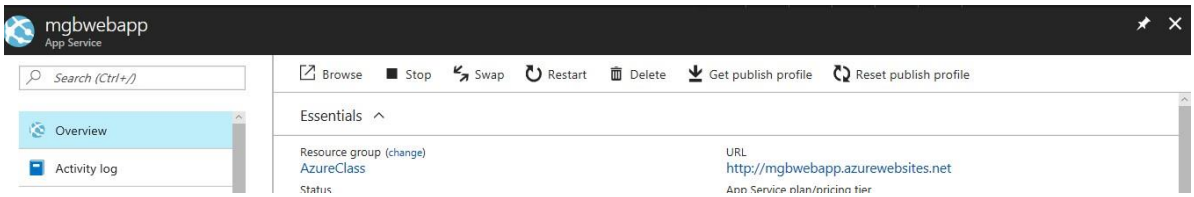
14. On the Azure Active Directory Settings screen change the management mode to Express, set the Grant Graph Permissions to **ON** then click *OK*. Then click **Save**.



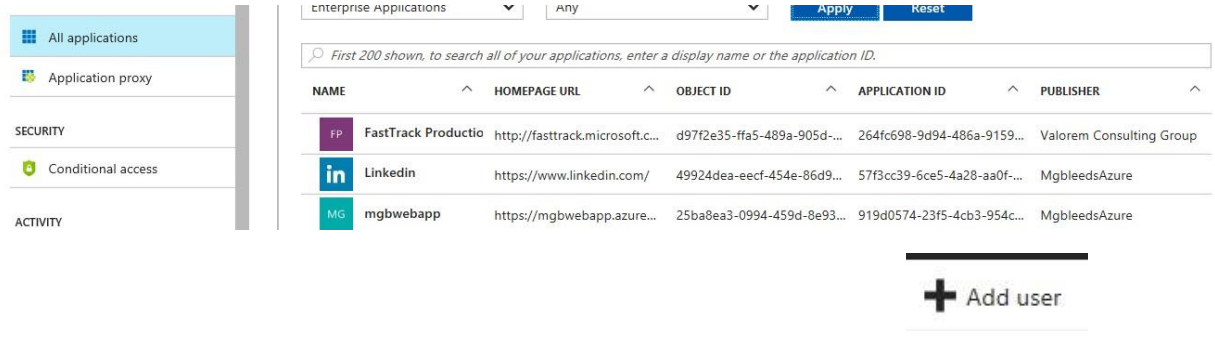
15. On the Authentication / Authorization page make sure you select **Login in with Azure Active Directory** in the Action to take when request is not authenticated drop down box. Then click **Save**



16. Scroll up to the overview section of your web app and copy the URL into Notepad. You will us this later.



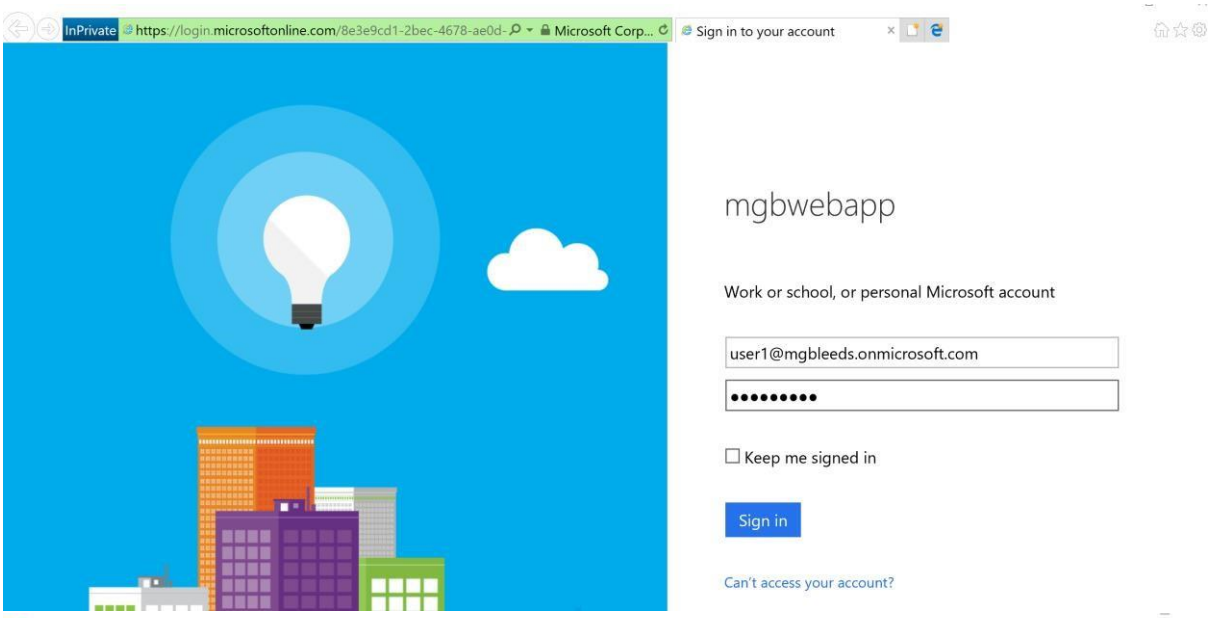
17. Go to the Azure AD section in the portal and go to the Enterprise Application section and then all applications. Your web app should be listed:

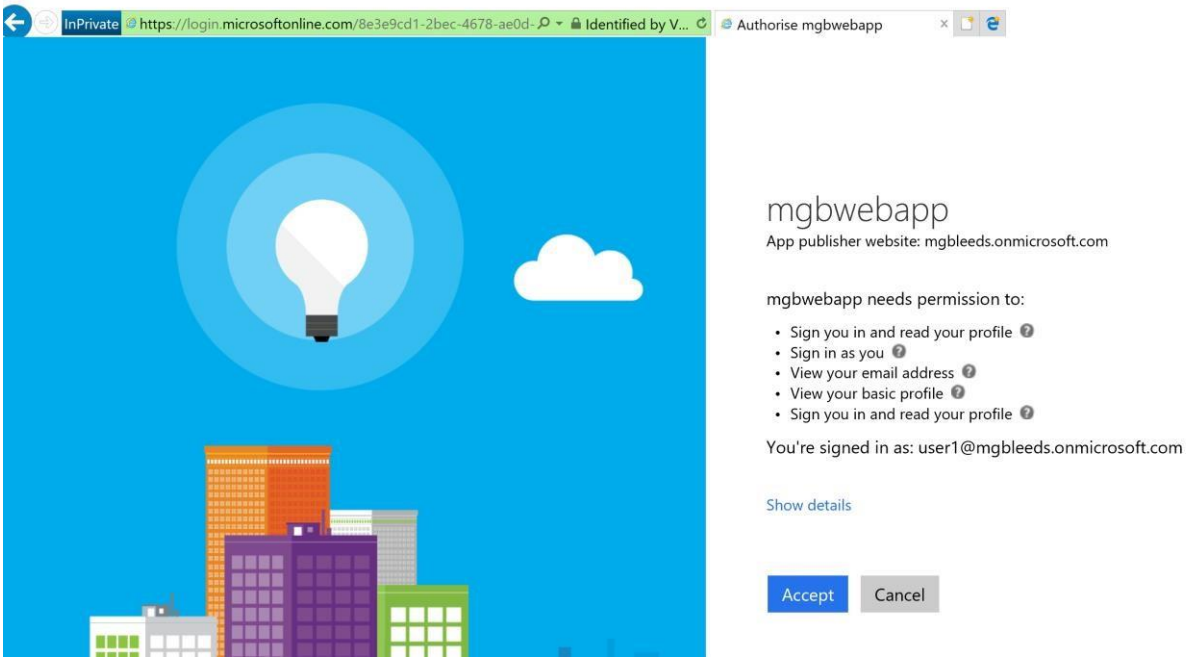


18. Select you web app and in the users and group section select
19. Add user1 as a user for the app and click **Assign**

You have now given user1 access to your application.

20. Using IE inPrivate Browsing or Chrome Incognito type the URL of your application. You will be asked to login, notice the name of your integrated application. Type in User1@***yourdomainname***.onmicrosoft.com as the username and type in the password. You will be asked to consent so the application to access your details. This is an example of integrating with the Graph API



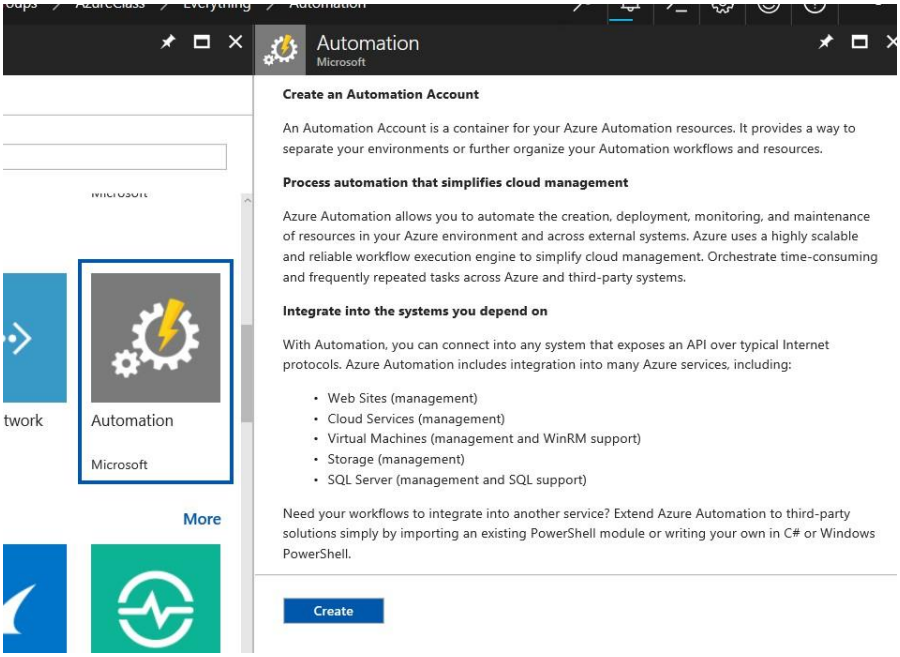


Lab 7 Working with Azure Automation

In this lab, you will be working with virtual machines you created in an earlier exercise. You will use Azure Automation to install IIS and Windows Backup on the virtual machines and to create a run book that takes actions against the Virtual machines.

Exercise 1 Creating an automation account and editing a Runbook

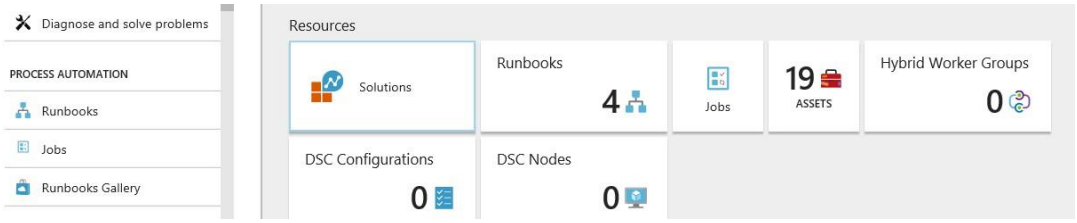
1. In the Azure portal navigate to your AzureClass resource group and click + Add 2. Search for and create an Automation Account:



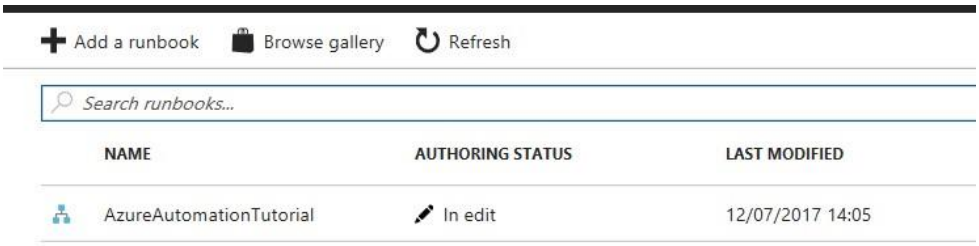
3. Provide a name for your automation account and then leaving everything else at their default settings click **create**

It will only take a minute to create your new Automation account. Once created access your automation account. You will be working with a sample Runbook in this exercise.

4. From the AzureClass automation account select RunBooks from the process automation section



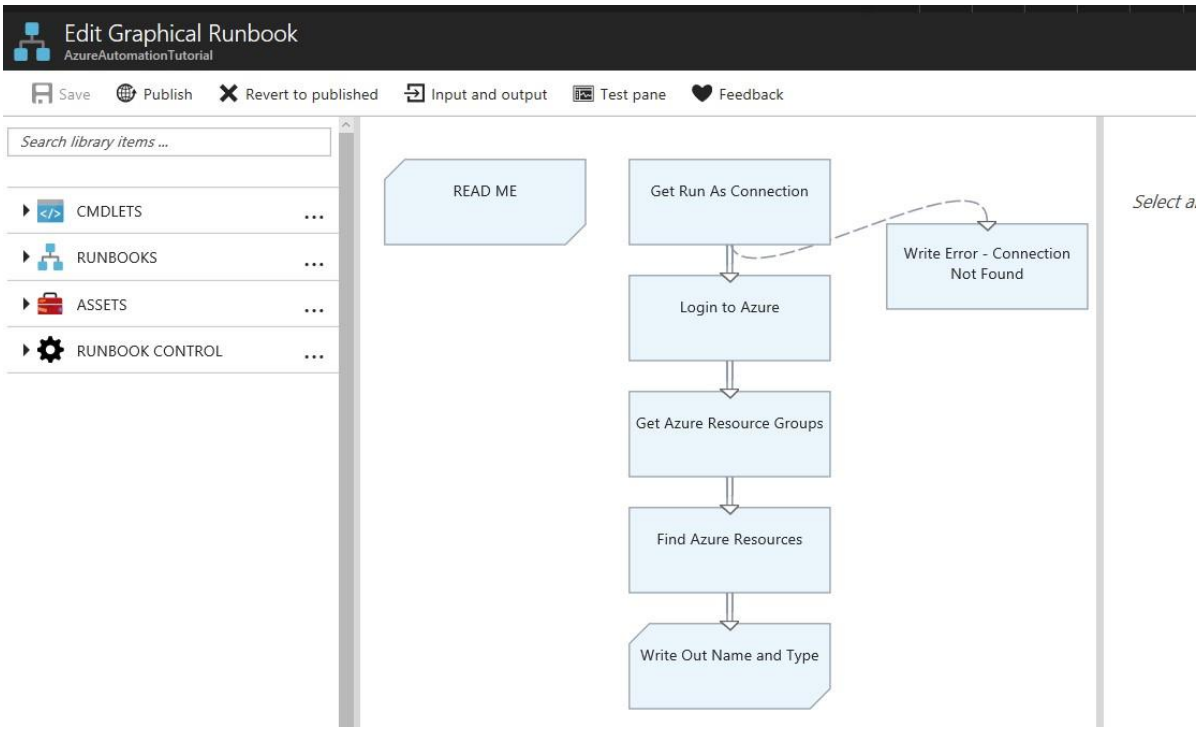
5. From the Runbooks section select **AzureAutomationTutorial** (it should be top of the list)



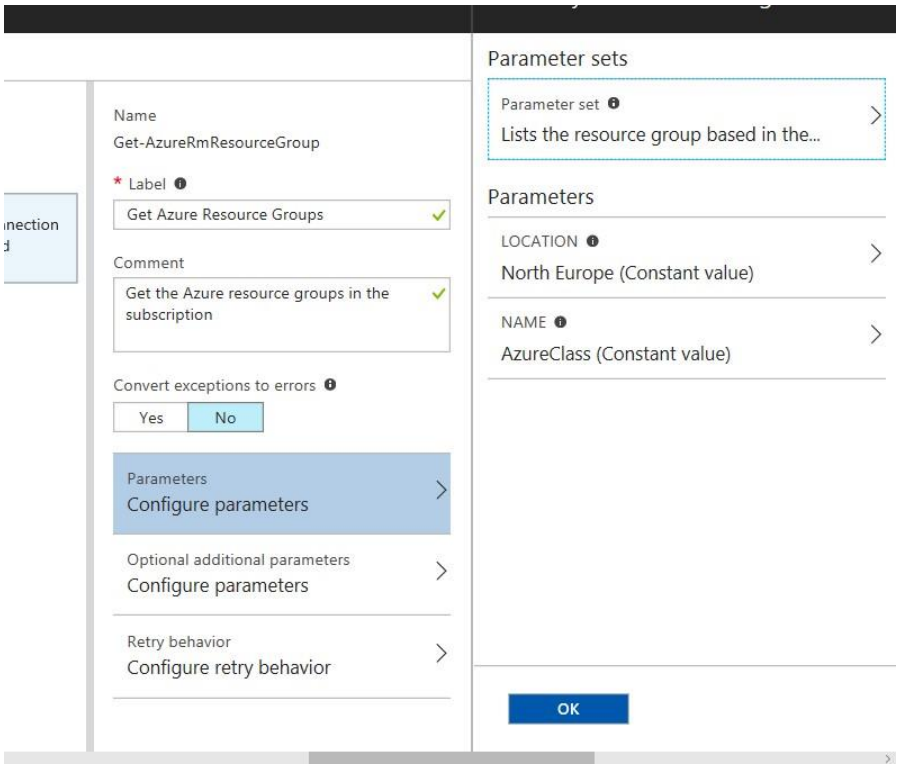
6. Next select edit 

The Edit Graphical Runbook pages should open up and you will see a screen like the one below. In this example, the runbook will use the Run AS account to login to your Azure

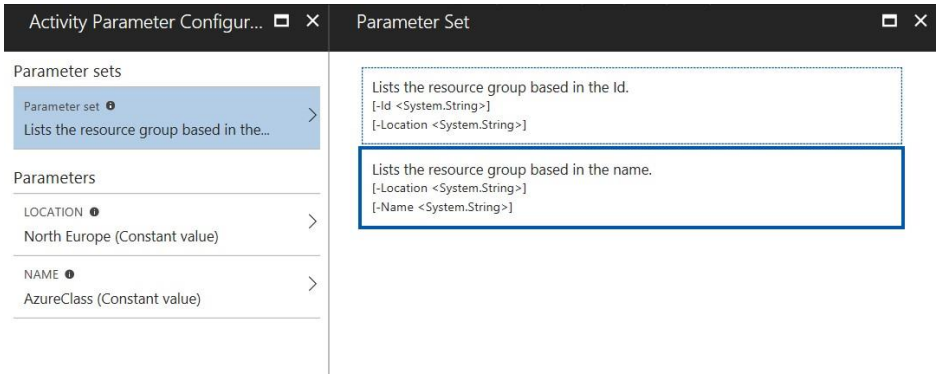
account a get a list of objects from a resource group. We must tell it which resource group to work with.



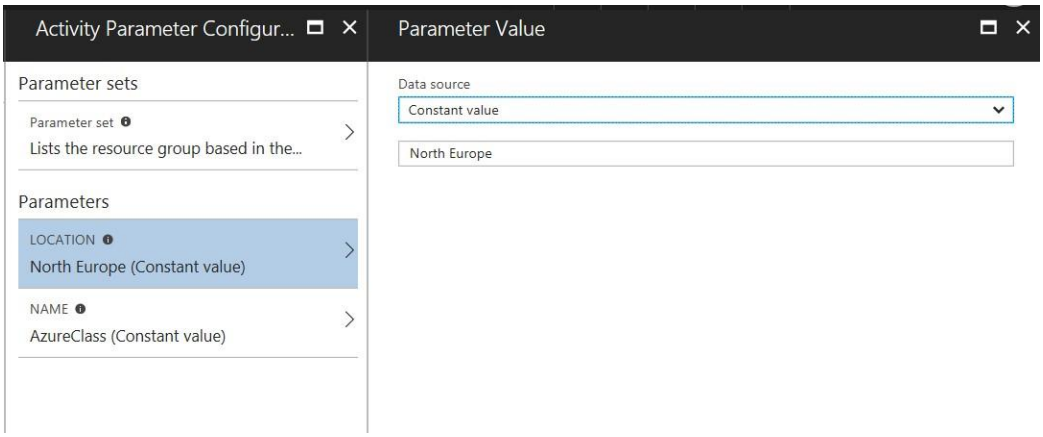
7. Double click the Get Azure Resource Groups step then click Configure parameters



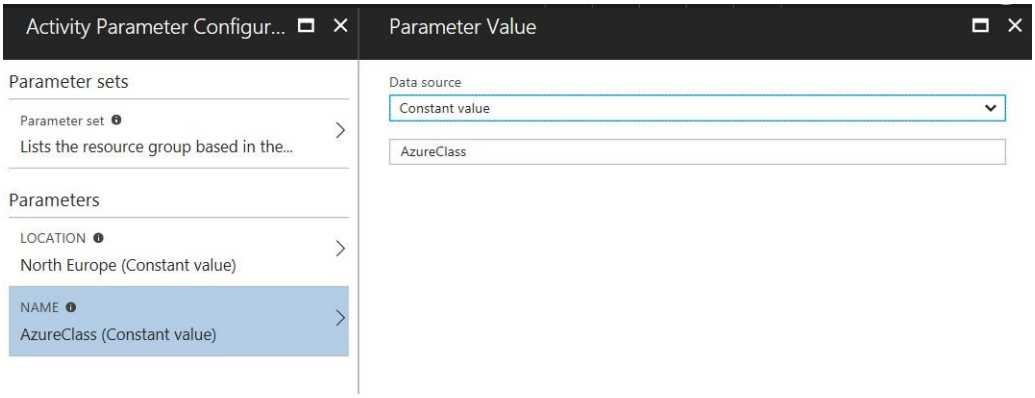
8. Next click the parameter set section and make sure that **List the resource group base in the name** is selected



9. In the Location Parameters section choose **Constant Value** as the data source and type **North Europe** as the value



10. In the Name Parameter section choose **Constant Value** as the Data source and then type **AzureClass** as the value.



11. Click OK to confirm the value then click save to save your changes.
12. Next click Test Pane and from the test pane click start to run your RunBook.

In this exercise, we have edited and run a runbook.

Exercise 2 Using DSC to manage Virtual Machines.

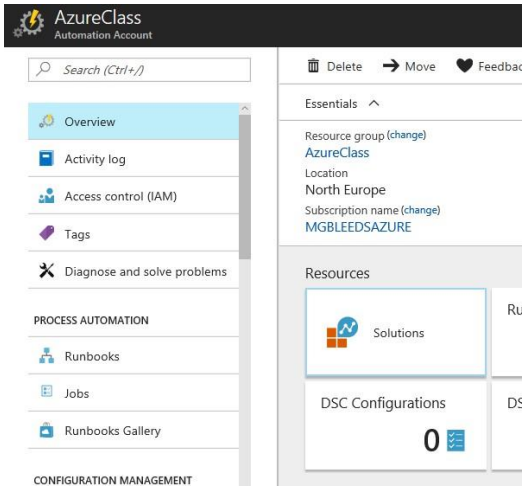
In this exercise, you will use Desired State Configuration to add roles to VM1 and VM2 that you created earlier.

1. From your local server, open Windows PowerShell ISE 2. Type in the following DSC code:

DSCEXAMPLE.ps1

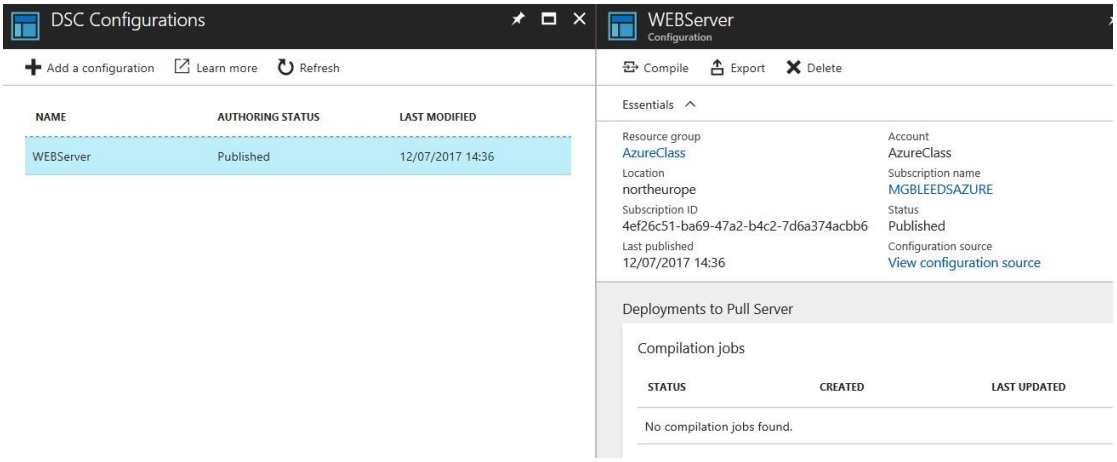
```
configuration WEBServer {

    Node ('Localhost') {

        WindowsFeature IIS{
            Name = 'web-server'
            Ensure = 'Present'

        }

        WindowsFeature Backup{
            Name = 'Windows-Server-Backup'
            Ensure = 'Present'
        }
    }
}
```

3. Save the file locally with the name DSCExample.ps1 4. Go back to the Azure portal and your automation account. 5. From the Overview page select DSC Configurations



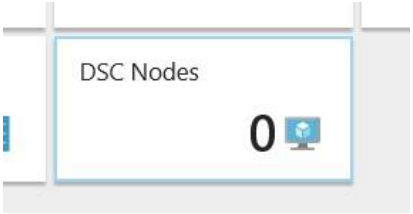
6. In the DSC Configuration screen select 

7. Next in the configuration file box browse for and select the file you created in step 3. Then click OK
8. Next on the DSC Configuration page choose your newly uploaded file and then select Compile 



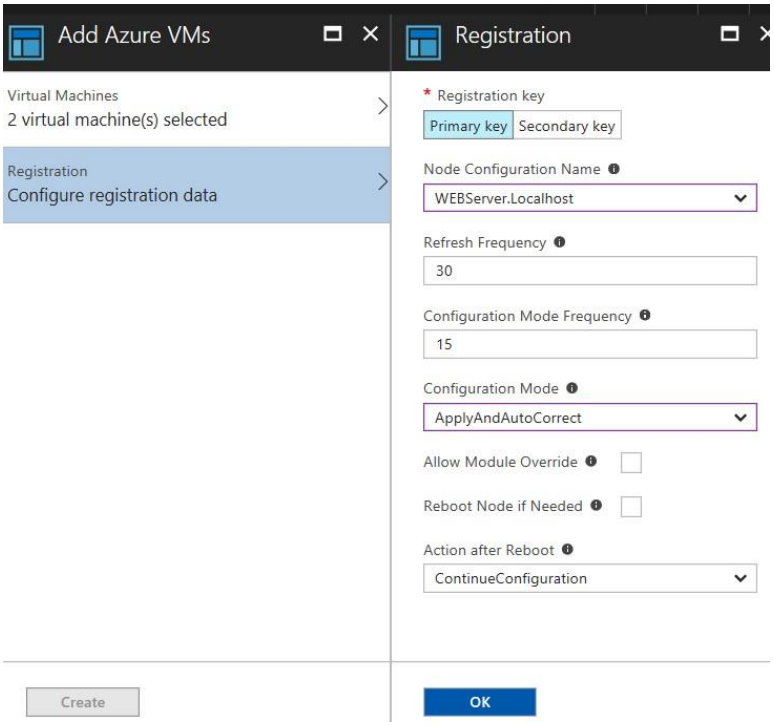
It will take a few minutes to compile your file.

9. Once your file is compiled go back to the AzureClass automation account Overview page and select DSC Nodes.



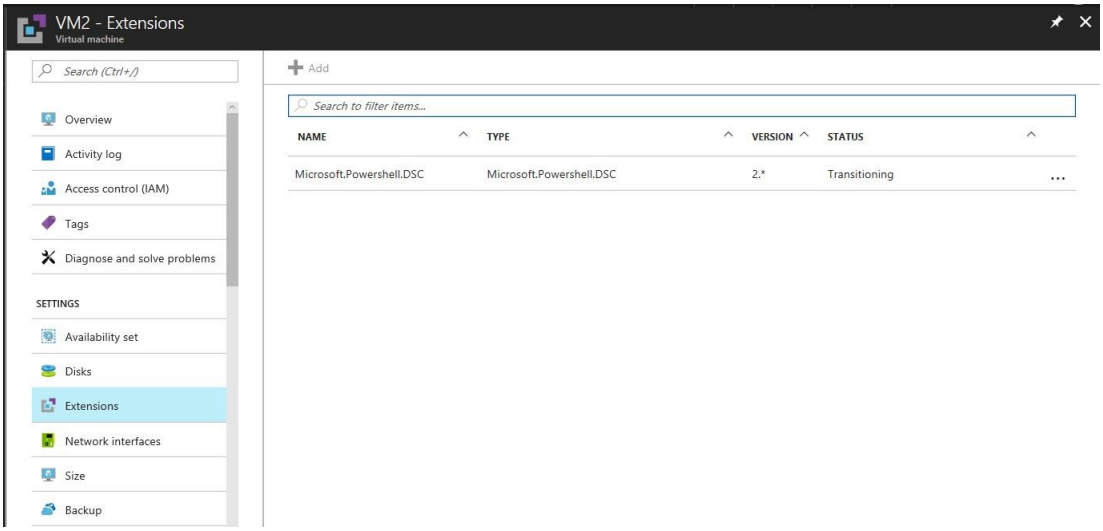
10. On the DSC Nodes pane select 

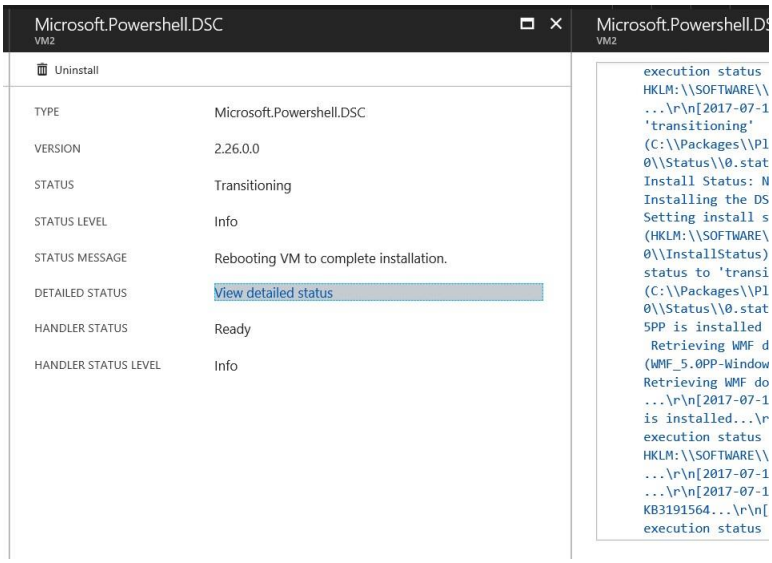
11. On the Add Azure VMs pane, select the Select Virtual machines to onboard section and choose VM1 and VM2
12. Next Select the Registration section, in there make sure **WEBServer.Localhost** is selected as the Node Configuration name and that **ApplyAndAutoCorrect** is selected as the configuration mode.



13. Next click OK and then click Create.

The DSC VM Extensions will now be enabled on VM1 and VM2. Once the DSC VM extension is registered your VMs will show up and DSC Nodes and the DSC configuration will be applied. If you go to the properties of one of your VMs can monitor the progress.





It will take a few minutes to complete, once it has completed IIS and window backup should be deployed. To test the configuration, login to VM1 and remove the IIS role manually. Because we have chosen **ApplyAndAutoCorrect** we should find that the configuration will be reapplied after a short period and IIS will be re-installed.

Once the DSC configuration has been applied you should see both VMs as compliant.