AZRIL203

Introduction to Azure Active Directory

This is an infrastructure lab, useful to both ITPro’s and Developers to learn the basics of Azure Active Directory. The main focus is on understanding the basics of the directory itself, how to create one, users and groups and one of the key scenarios for the ITPro which is connecting and synchronizing the directory with on-premise Active Directory. The lab will also enable Multi-factor authentication.
1. Login to the Azure Management Portal

The first task is to get you signed into the Azure portal. If you already have your own Azure subscription, you can use that for the lab. If you are an MSDN subscriber, you get some amazing Azure benefits – read about and activate these benefits here:


You can also sign-up for a free 30 day trial: https://account.windowsazure.com/signup?offer=ms-azr-0044p

For this event – free passes are available – just see a lab proctor for a subscription – this will be a Microsoft Account/Password that has been activated with an Azure subscription (the subscription has a limited number of 4 CPU cores and lasts until the Monday following the event)

On your lab machine, open Internet Explorer (which should open up at www.azure.microsoft.com)

1. Click on PORTAL and Sign-In to Azure with your Microsoft Account credentials.

2. IF you are using a free pass for the event, you might get this screen on the right.

   Click the SKIP THIS FOR NOW – and you will then be successfully logged in.

If you don’t get the skip this for now option or even if you do – you can press CONTINUE – and this will go into a workflow for you to validate the account – using a cell phone/SMS message. This will allow you to set your own memorable password – you will then no longer see the prompt if you need to login again.

Click through the start wizard if you get this and you should be on the management portal with something resembling the screen on the right.

The various Azure Services are in the left nav bar – you click these to see the instances you have of these services.

You click the +NEW option at the bottom left to create new services.

GREAT...!
2. Core Setup for On-Prem AD

Login New Account

Maybe we can have a single AD box and then others can just install the agent to get the sync to work..? – try this...

Add Azure AD -> Create New – Name and Domain (<alias>tedemo.onmicrosoft.com)

Enable AD Premium (click in the directory, click on Licences and enable it)

Create a New Network (leave all defaults)

Create a New VM (WS 2012R2) – name <alias>teeWSAD (Basic, A1) – account <alias>admin, Password1! – put in vnet

Create a New VM (WS 2012R2) – name <alias>teeWS (Basic, A1) – account <alias>admin, Password1! – put in vnet

Connect to teeWSAD, Connect to teeWS

Turn off IE ESC (Server Manager – Local Server) for both machines

On WSAD – Add Active Directory Domain Service Feature (accept all defaults)

After install – promote to DC (add new forest – root domain – contoso.com) accept all other defaults

In Azure – add in the DNS Server (name: <alias>teewsad, IP – of the machine) – SAVE – reboot the client and when it comes back up, connect and using IPCONFIG – check the DNS server is your WSAD box

Delete the DNS Forwarder Record (DNS Manager – Root Hints – Forwarders – Delete any entry)

Make sure you can ping from WS to WSAD (use FQDN – IPCONFIG /all to get the internal DNS name – Vmname.CSName.xx.internal.cloudapp.net)

Create a new OU=Marketing, create a new user in the marketing OU (Bob Smith – bsmith – set password – never expires, don’t change on logon etc)

Join the WS to the domain (control panel – system – use contoso.com as the domain – and use bsmith as the domain member authorized to join – and add in bsmith as an admin on the box) – restart machine and login as contoso\bsmith
To install AD sync (which you have to do on the teeWS machine – NOT the domain controller) – you need .NET 3.5 SP1 and .NET 4 (latter is part of .NET 4.5 so is already installed) – add app server role and select the .NET 3.5 install

On the teeWS – login to Azure, goto the directory – activate dir sync (directory integration) and download the sync tool and install it. Then configure it as per the Wizard.

### 3. Get your network and VM’s provisioned

Before you understand WHAT and WHY you are doing certain things, you need to get going on creating some core infrastructure and a few VM’s as these take a bit of time to spin up. You will get to understand WHAT/WHY later in the lab. Let's get you started – don’t worry about understanding anything about what you are doing right now 😊.

1. Click **NEW -> COMPUTE -> VIRTUAL MACHINE -> QUICK CREATE**. Enter a DNS Name – use `<your email alias>-quickvm`.
2. Choose the **Windows Server 2012 R2 Datacenter** Image. Leave the size the default (A1).
3. Enter a username and password – this will be the admin account on the VM you will create – suggest `<your alias>admin` and Password1! – e.g. psmithadmin. Yes it’s just for training – you wouldn’t do this for real would you?
4. Choose any region you like and click ok.
5. Next, you will create a Virtual Network. On the Management Portal, select **+ NEW -> NETWORK SERVICES -> VIRTUAL NETWORK -> CUSTOM CREATE**
6. Enter a NAME (which must be unique – suggest your email alias and –vnet – e.g. psmith-vnet).
7. **REGION** - Select a datacenter location for the network – Pick EITHER West US or East US. (the reason is that Virtual Machines are only available in these two regions in the US).
8. In the Availability Group box, enter the same name as your network name adding “–ag” after the name – e.g. psmith-vnet-ag
9. Check the “configure a point-to-site VPN” option on the next screen and then click past the point-to-site connectivity page.
10. On the Virtual Network Address Spaces page, click the add gateway subnet button and click ok. Your virtual network will start creating.

When it has **finished** – Status = Created, click the network, click on the dashboard tab and then click the CREATE GATEWAY button at the bottom of the portal.

Ok, great… Now you have that all going, let’s explain what you just did...
The first thing you did was create a simple standalone VM (the white server icon in the Azure cloud in the picture below). Then you created a virtual network – a specific type called a point-to-site network. This allows individual computers to connect to the network.

Why do you need a network? It is common for an application to span across multiple servers and these servers need to communicate with each other and maybe even between cloud and on-premise systems. Often, you need to connect to this network from your own data center or in the case of this lab – from your computer. The GATEWAY that you created is the logical way this happens – think of a gateway as creating a door into your network from the outside. There are a variety of ways depending on your needs to connect to an Azure Virtual Network as the picture shows.

Now you don’t have to put VM’s inside networks, they can happily live by themselves (the very first VM you created is like this). When you want to expose connections to your VM over the internet (as opposed to from inside a virtual network) you control this access via ENDPOINTS which you will learn about later. By default when you create a VM you get access to the VM via remote desktop (and an endpoint with remote desktop enabled) - the REMOTE DESKTOP dotted line in the picture. Never forget - YOU have choice and control around how visible, what traffic flows, who can access the machines and from where. Let’s continue and spin up some more VM’s but this time do it the long way, so you understand a bit more about what is going on.

Click +NEW -> DATA SERVICES -> STORAGE -> QUICK CREATE

Use a name such as <alias>vmstore – e.g. psmithvmstore (unlike some other services, the name can only contain LOWER CASE letters and numbers).

Select the Affinity Group that was created when you created the Network. Select Locally Redundant replication and finish (this will take around 1 minute to provision).

Just like a regular virtual machine you may have used on your own computer or in your company data center, a VM has one or more .VHD files which represent the hard disks of the computer. These disks are stored in another service on Azure – BLOB Storage – a highly resilient, scalable store that can be used for many things such as backup, application content, media, and documents. In your case – virtual hard drives for your VM. BLOB storage maintains multiple copies of any files in it and additionally can maintain GEO-REDUNDANT copies between two Azure Data Centers. When you created your storage account you set the replication level to Locally Redundant – you don’t need geo-level replication for training. Click on Storage and click the odd storage name and then click containers and then the container called vhds. What you will see there is a 127GB file with a .vhd extension – this is the main disk, the c:drive, for the VM you created at the start of the lab. You created a new storage account above because you want to control the name. The other thing to consider is that store accounts have limits in terms of IO (read/write operations and bandwidth) – not individual files in the account. If you put many .vhd files in the same account, you will come across bandwidth constraints more quickly.
Affinity Groups – this concept is a logical grouping of resources. Data Centers are full of huge racks of computers and clusters of racks that host a particular service. As you have applications that require multiple services, you need a way of saying to Azure – “put all these close together in a data center”. That’s what affinity groups do. For a VM, you certainly want the place that stores the VM’s disk, the VM itself, and even the network the VM’s are in – to be very close together.

Affinity groups are hidden away in the management portal – to see your - click on and select AFFINITY GROUPS to see your groups.

Now create another Virtual Machine – the long way...
Click +NEW -> COMPUTE -> VIRTUAL MACHINE -> FROM GALLERY.

14. You will provision TWO virtual machines – one with just Windows Server in it, the other with SQLServer in it. Start with the plain Windows Server Machine. The first machine will actually be used to installed a domain controller in part 2 of this lab.

First you will notice in the gallery that there are lots of machine images to select from with images you would expect – various Microsoft server software and operating system choices and maybe some you would not expect – flavors of Linux and software from other vendors.

You can also have your own images available. Select Windows Server 2012 R2 Datacenter from the gallery and click NEXT

16. Enter a name for your VM. Use your email alias followed by “-dcm” – e.g. psmith-dcm

17. Select the BASIC tier and pick an A1 SIZE Machine (if you are using a supplied trial account, there is a limit on the number of CPU cores. Picking a large instance size might exhaust this limit and you won’t be able to create the second machine).

18. Enter a USER NAME and a PASSWORD (suggest you use the same credential when you created your quickvm in the very first lab step.

19. For the CLOUD SERVICE option – select “Create a new Cloud Service”. For the Cloud Service DNS name – use <alias>-dcs e.g. psmith-dcs.

20. In the Region/Affinity Group/Virtual Network box, select your VIRTUAL NETWORK – the one you created above. Leave the subnet as the default and select the STORAGE ACCOUNT you also created.

21. In the ENDPOINTS section, DELETE both the REMOTE DESKTOP and the POWERSHELL endpoints (click the x on the right). Go to the next screen - if your Cloud Service DNS Name is not unique – this will get flagged here and you will need to change it. Finish the wizard leaving everything else the default values. Your VM will start the process of getting created.

Now, create a second VM with SQL in it. Repeat the above steps, selecting the SQL Server 2014 RTM Enterprise image from the gallery and selecting a different name such as alias-sqlvm – e.g. psmith-sqlvm.
Also create a new cloud service for this VM, something like <alias>-sqlcs. Make sure to add this to your network and use your storage account.

So at this point you have TWO Virtual Machines spinning up inside a Virtual Network and the first VM – which should have finished provisioning – as a standalone VM.

When you created these VM’s, in the creation workflow, you created a new cloud service and gave it a name – what is that thing? A Cloud Service is a container that you can put multiple VM’s into. There are some benefits of putting VM’s in the same cloud service. For example – you get easy DNS name resolution between the VM’s and you can configure more advanced settings – like load balancing. When you create a VM, a cloud service is always created for you 1 for 1 by default. If you click on Cloud Services in the portal, you will see your three services. Later you will actually create another VM and put this into one of these cloud services.
4. Connecting to Your First VM

Right – let’s get you connected to the first VM you created and understand a bit more about what you have...

Click on Virtual Machines in the portal on the left and check that your “-quickvm” is in Running State. Click on the VM and then click dashboard. The dashboard gives some some basic details, metrics and configuration of your VM. Scroll down and you can see how many disks you have. Look on the right and you get all the DNS and IP address details.

1. Click CONNECT and select OPEN for the .rdp file download that get’s presented. Follow the options and you will be presented with a logon dialog. Enter the credentials for the VM you established in the very first step – should be something like <alias>admin (e.g. psmithadmin) and Password1! – Or whatever you created. Follow other prompts and eventually a remote desktop session to the machine will launch full screen. **NOTICE:** the circled value (and this will also be in the title bar of your RDP connection). This value is the port number that the RDP connection is using – it was randomly assigned when the VM was created. So this value must be known to connect to your VM, in addition to the DNS name (which is in the above screen shot – psmith-quickvm.cloudapp.net). Remember this is the public DNS name of the cloud service – the container for your VM, not the VM itself.

2. You should be connected to your VM. One little trick is to RESTORE your VM to a Window (click the middle button) and then select smart sizing (right-click the top left application icon). This allows you to change the size of your VM and stops you being confused about which actual machine you are using especially when you might have several windows open to different machines.

3. This is just a computer now – yours – it happens to live in Azure – it’s virtual. In Server Manager in the VM, click on Local Server. In the top panel, right hand side – find the IE Enhanced Security Configuration, click the value and turn it off – this allows you to use IE without any security warnings.

4. Open up a Web Browser now and go to http://azure.microsoft.com – it’s just a computer, it happens to only have Windows on the box. You can now configure and install whatever software you like on the machine. You might find the machine is a little sluggish... Two things – you are only running on a single core with not a lot of memory. The machine has also just started up so many of the services and still spinning up and doing their thing for the first time. It will get more responsive. Typically an A2 size machine – 2 cores and 3.5GB RAM – will perform much better.

5. Open Computer Manager (Server Manager -> Tools (top right) -> Computer Manager) and then click on Disk Management. Also open up file explorer and look at your drives (don’t ask why a virtual computer
sat in a data center with almost no people has a DVD drive). The Local Disk (C) is where the operating system is installed and your applications. The D Drive is a temporary drive actually on the physical host computer. This drive is not persisted in a VM reboot/failure and some management operations – like changing the size of the VM also erase this drive. So it’s useful for working set data – but don’t put anything on this drive you can’t afford to loose. You would generally want application data to be stored on another disk(s) – for example if you had a database. You might create two more disks – one for the database and one for the log files. How do you do this?

Go to the Azure Portal – you should be on the dashboard for your VM. Click the ATTACH button and select Empty Disk. Change the disk name (something like <alias>-quickvm-data). For the size enter 20 and click ok. Wait until the operation has completed (20-30 seconds) and on the dashboard you will see a new disk (of type data disk) appear in the disks section.

Switch back to your VM and switch to Computer Manager/Disk Management. You will see your new 20GB disk show up. Right-Click the disk and select Initialize and accept the defaults. Then right-click the unallocated space and select New Simple Volume and then click your way through all the defaults. In a moment you will have a new 20GB disk. You might get an extra dialog pop up asking you to format the disk – just cancel this one. Go to File Explorer, click on the new disk, create a new file of any type – say a simple text file.

Let’s think about what you just did – it’s sort of amazing this is even possible. You created a VM, with a full copy of Windows Server in a data center somewhere in the world, the VM is accessible over the internet from anywhere in the world (and on multiple devices). You connected to the VM and you just added a 20GB disk to the VM. You did this all in about 20 minutes.

Back in the Azure portal, for your VM, click on CONFIGURE. Some of the same settings are available when you created your VM. You see you can change the tier/size of your VM. There is a setting called Availability Set – which you didn’t use. An availability set allows you to put multiple machines into a group for availability purposes. Say you have a web server farm – put the VM’s all in an availability set and whenever there is any maintenance say to the underlying host machine, Azure will ensure it does not take all the machines in your availability set down at the same time. Also, the machines in the group are spread across different underlying physical hardware racks – so any hardware failure at the rack level won’t affect the entire set of machines.

Click on the EndPoints tab. You will see two EndPoints – one for PowerShell and one for Remote Desktop. EndPoints are the way into your VM through Azure from the Internet. Your VM has to ALSO allow access. In this case, the Windows Firewall has to allow RDP traffic – you can check this in the Windows Firewall. EndPoints map the random public port to the private port – which that specific protocol uses – an extra layer of security. When you created you second two VM’s you deleted the EndPoints – there is no way to connect to these machines now directly from the Internet.

Click on the Remote Desktop EndPoint and click the Manage ACL button at the bottom. As you can see – there is another level of security here which is to only allow access to this endpoint from certain network/IP addresses.
Let’s finish up and do something scary – you are going to **DELETE** your VM. Go back to the **dashboard** in Azure for the **VM** and click the **SHUTDOWN** button. Your Remote Desktop session will terminate, your VM will shut down. The VM is not deleted though it’s all there or rather the disks are there and the configuration of the VM – it’s just NOT provisioned on a physical machine now. You are paying only for the cost of storing the .vhd files in blob storage.

12. Once the VM has shutdown, you will be able to click on **DETACH DISK** – this will separate the DATA DISK from your VM – it will just live by itself and you can then attach it to another VM.

Now click **Delete** and select the option to **DELETE ATTACHED DISKS**. This will delete the VM and the underlying .VHD File. You still have your **data disk** though – anything stored on that disk is persisted and available. You can attach the disk to another VM, you can copy the vhd file back to your own data center and mount it to a VM there – whatever you want – it’s a 2 way street.

13. When your VM is deleted, click on **Cloud Services**. Notice the cloud service is still there for the VM you just deleted. You could have deleted the Cloud Service and you would have an option to delete the VM and attached disks – although weirdly – this option does not delete the underlying .VHD files. So either way you have some clean up to do. **Delete the Cloud Service.**