How-To-Guide

Windows Virtual Desktop

Step-by-Step Guide

August 21st, 2019

Contents

Windows Virtual Desktop

Overview

For customers to deploy WVD there are multiple dependencies they need to have in their environment. In this lab, we will walk you through how to set up Windows Virtual Desktop Public Preview as well as the dependencies it has and other features that enable the customer to have a working tenant.

This lab will guide you through

- How to set up a Virtual Machine which will act as your Domain Controller
- How to Configure your Domain Controller
- How to configure AAD Connect
- Instructions on how to deploy Windows Virtual Desktop
- Installation and Configuration of FSLogix
- Installation of OneDrive in per machine mode
- Automatically Turn on and Turn off your Virtual Machines so you reduce consumption

Solution Architecture

At the end of the lab attendees will have deployed WVD in their Azure subscription and have FSLogix and OneDrive enabled with the machines being on an automatic stop and start schedule

Virtual Machines

VM Name	IP Address	Description
adVM	10.0.0.4	Domain Controller in Azure for contosowvd01.com, DNS Server, AD, AD Sync, Profile Store

Requirements and Notes

1. We assume attendees have an Azure Subscription.
2. We also assume you have Global Admin rights to your Azure subscription.
3. Additional URL

Description	Vanity URL
Github template	**Aka.ms/WVDGithub**
Web client	**Aka.ms/WVDWeb**
OneDrive Installation File	https://onedrive.live.com/about/download/

Create your DC and setup Azure AD Connect

The main benefits of WVD is M365 integration in the virtual environment such as OneDrive, Outlook. Here we create an Azure AD tenant with the Required licensing to be able to deploy OPP & Onedrive

With Windows Virtual Desktop, your VM's needs to be Domain joined. In this lab we will deploy a Domain controller as it is cheaper to run than Azure AD DS and we will configure a hybrid identity in the environment as WVD will use User identity for Apps Publishing.

For the Azure AD Connect architecture we will go for a Password Hash Synchronization but here are the 3 available choices:

- **Password hash synchronization (PHS)**
- **Pass-through authentication (PTA)**
- **Federation (AD FS)**

Create a Domain Controller in Azure

Go to your azure AD portal (Use your Visual Studio subscription or AIRS subscription)

Create a VM with the following settings.

Create a virtual machine

Basics | Disks | Networking | Management | Advanced | Tags | Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image.
Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization.
Looking for classic VMs? Create VM from Azure Marketplace

PROJECT DETAILS

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

* Subscription: Microsoft Azure Internal Consumption
* Resource group: (New) Lab-WVD
Create new

← New RG

INSTANCE DETAILS

* Virtual machine name: Lab-WVD-DC01
* Region: (US) East US 2

Same Region as VM

Availability options: No infrastructure redundancy required
* Image: Windows Server 2019 Datacenter
Browse all images

2019

* Size: **Standard DS1 v2**
1 vcpu, 3.5 GB memory
Change size

ADMINISTRATOR ACCOUNT

* Username: Local-Admin
* Password: ••••••••••••••••
* Confirm password: ••••••••••••••••
Login with Azure Active Directory (Preview): On / Off

INBOUND PORT RULES

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

* Public inbound ports: None / Allow selected ports
* Select inbound ports: RDP

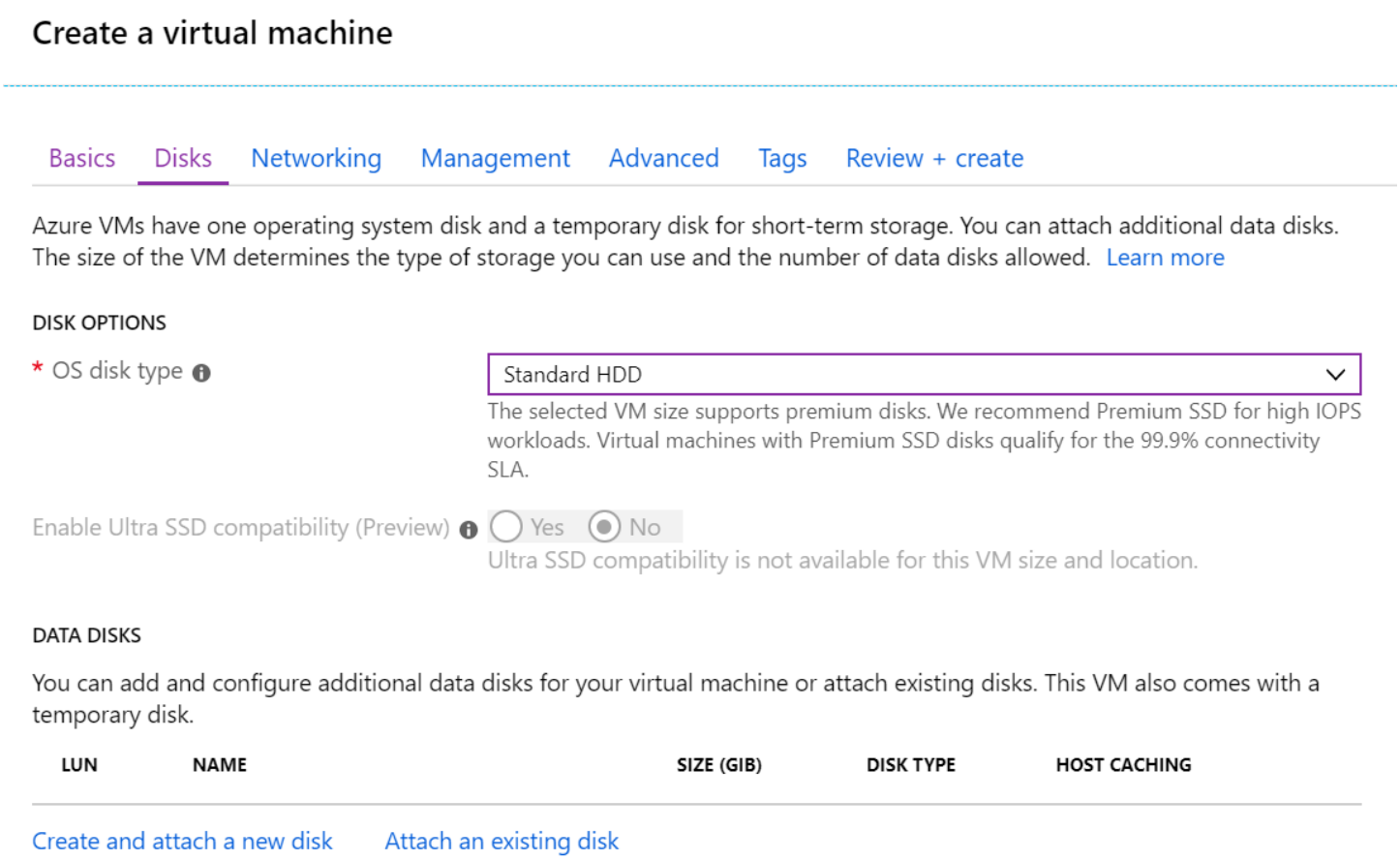
allow RDP or OTA

These ports will be exposed to the internet. Use the Advanced controls to limit inbound traffic to known IP addresses. You can also update inbound traffic rules later.

SAVE MONEY

Review + create | Previous | Next : Disks >

Choose cheapest Disk Drive

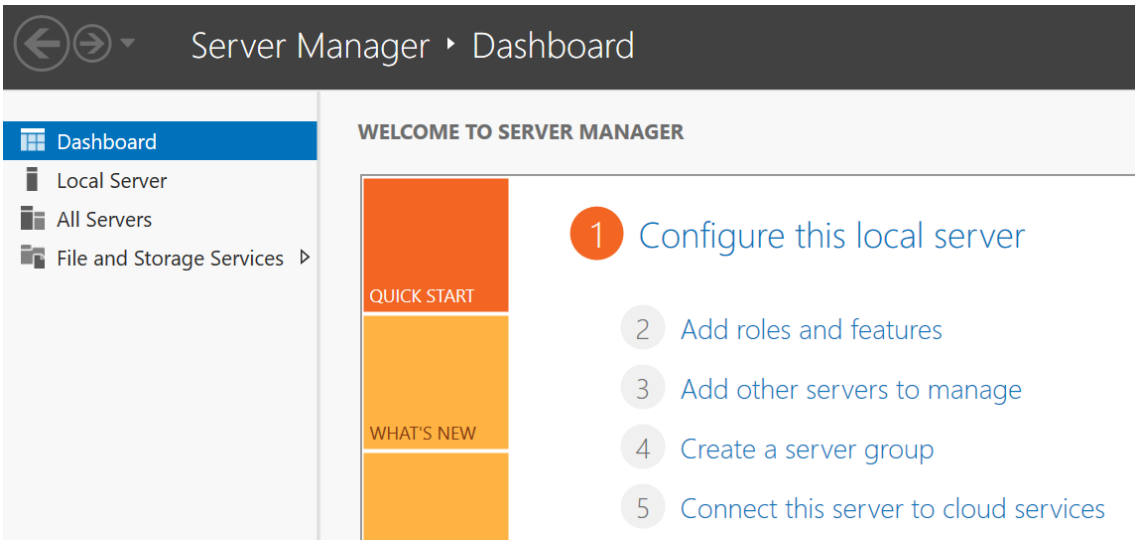


Press "Review & Create" and the "Create"

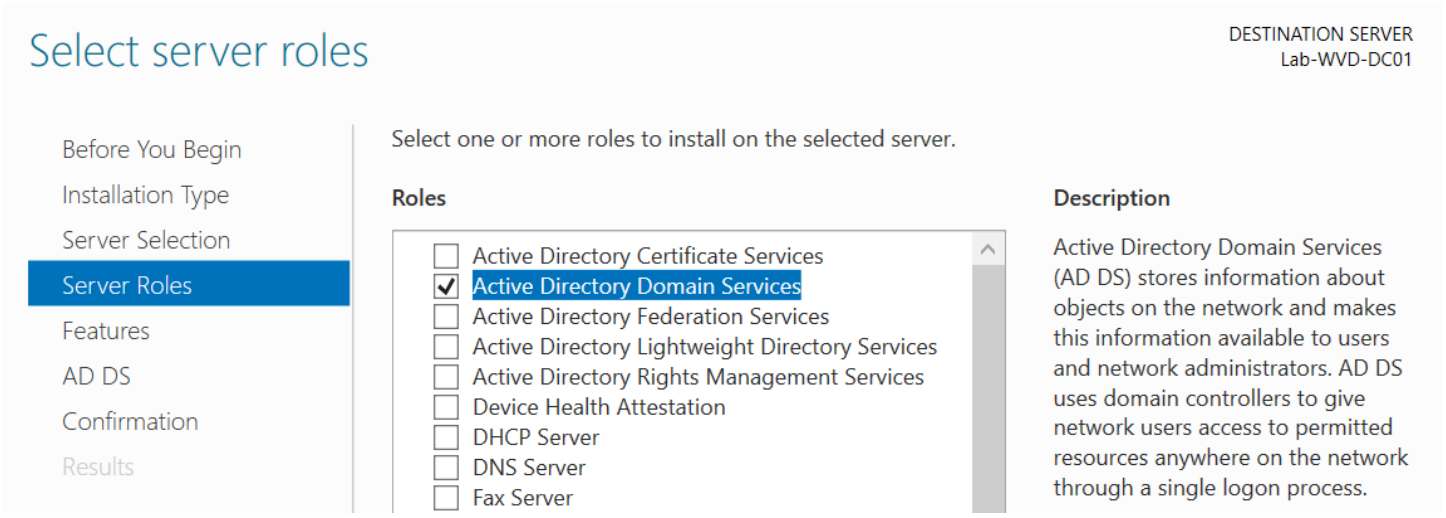
Configuring the Domain Controller

Open an RDP session to your DC01

Add a new role

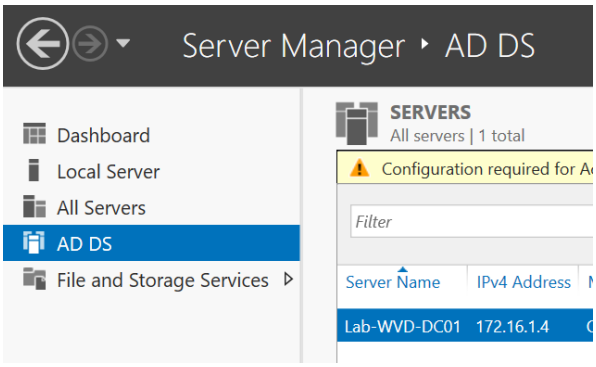


Choose "Active Directory Domain Services"

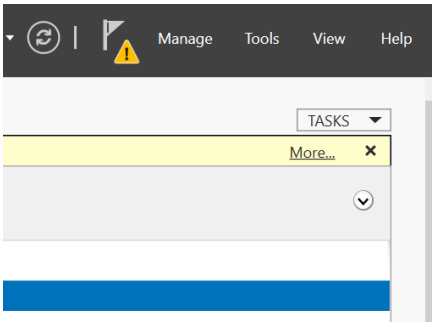


Press "Next" 3 times and then "Install"

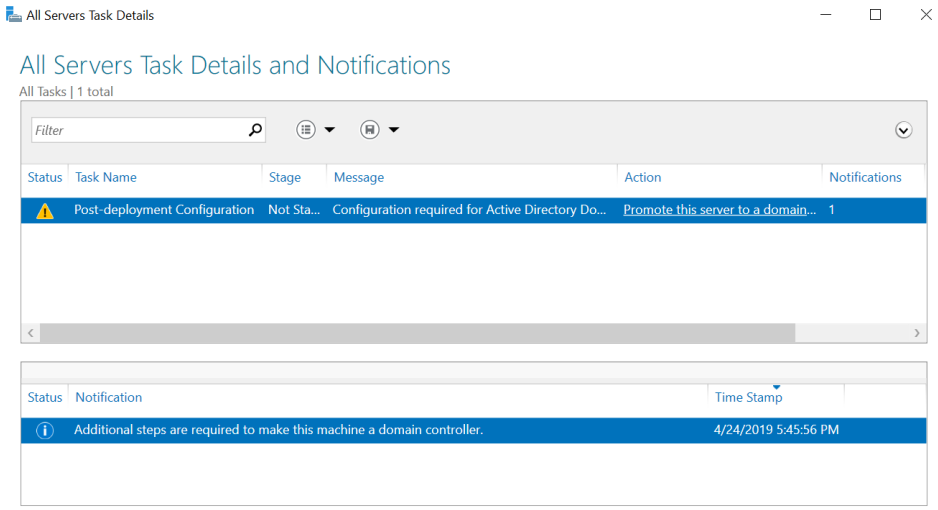
When the role is installed go to the AD DS management pane



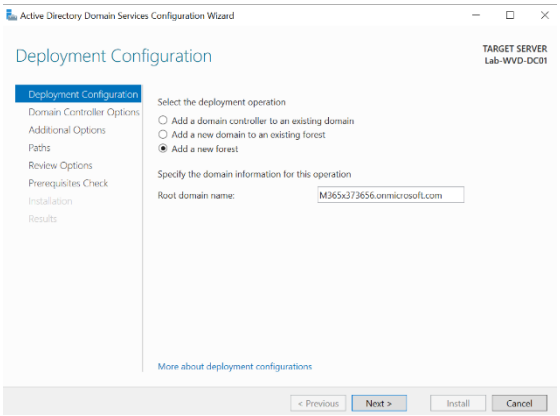
Click on More in the top right



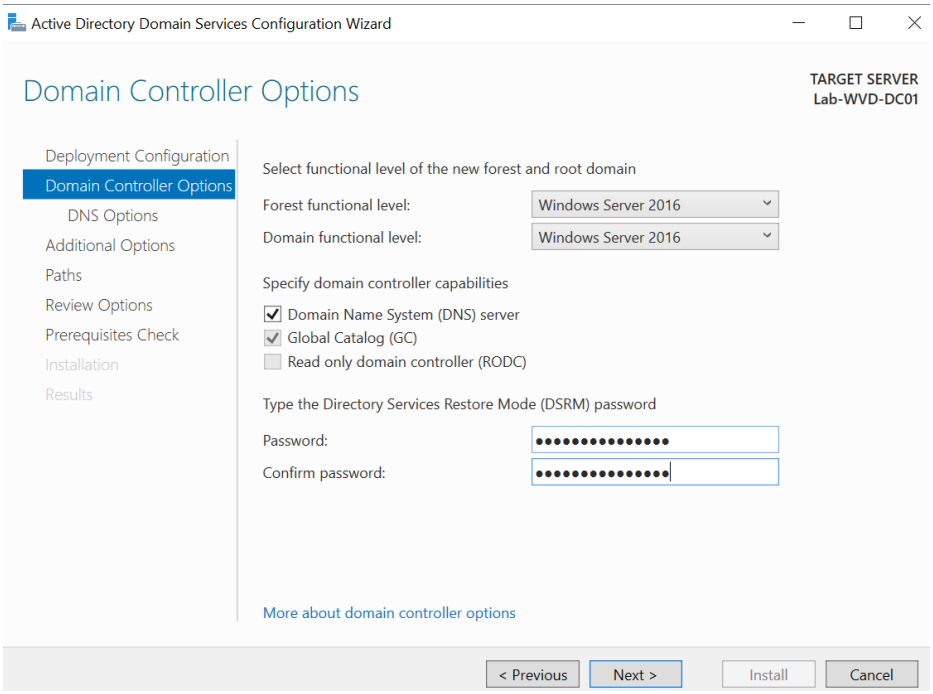
Then click on "Promote this server"



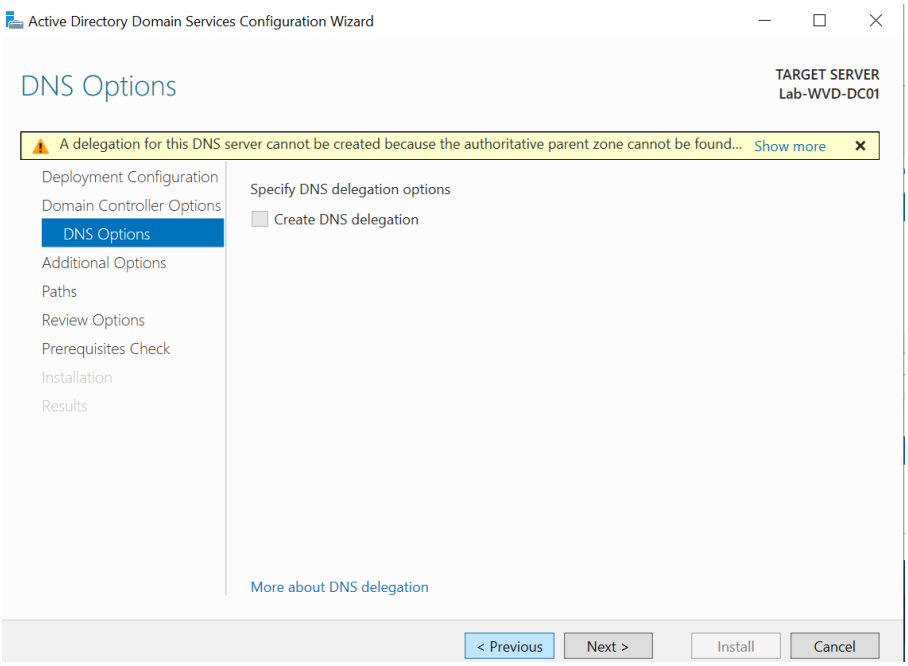
Choose "Add a new Forest" and type in the FQDN of your demo.microsoft.com tenant name and then "Next"



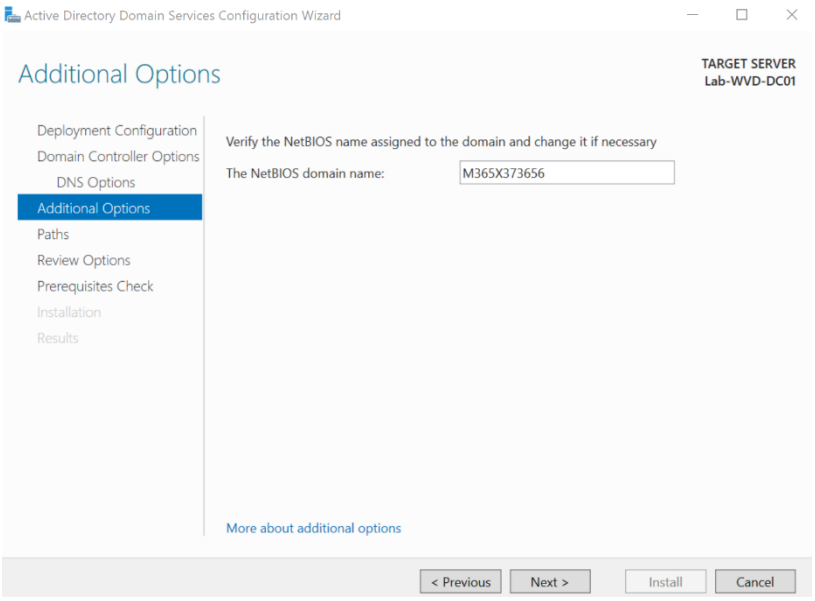
Type in a new DSRM password



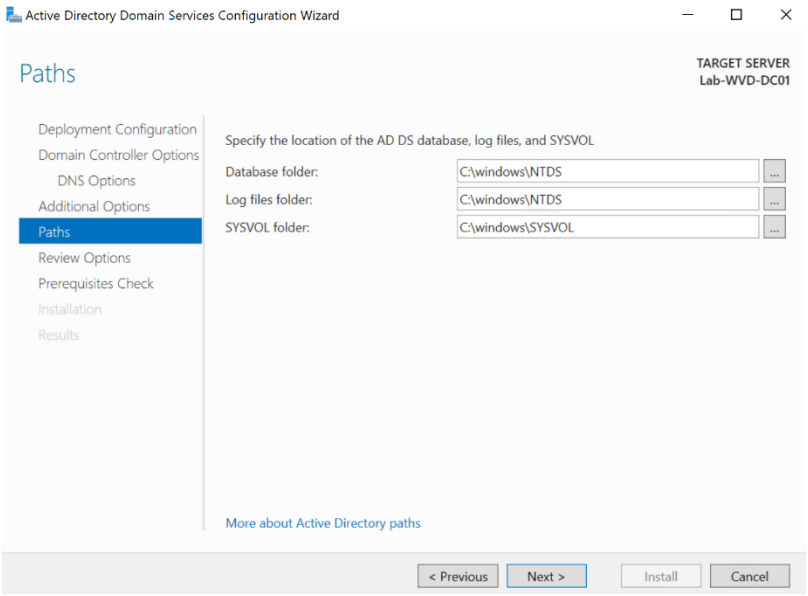
Press next



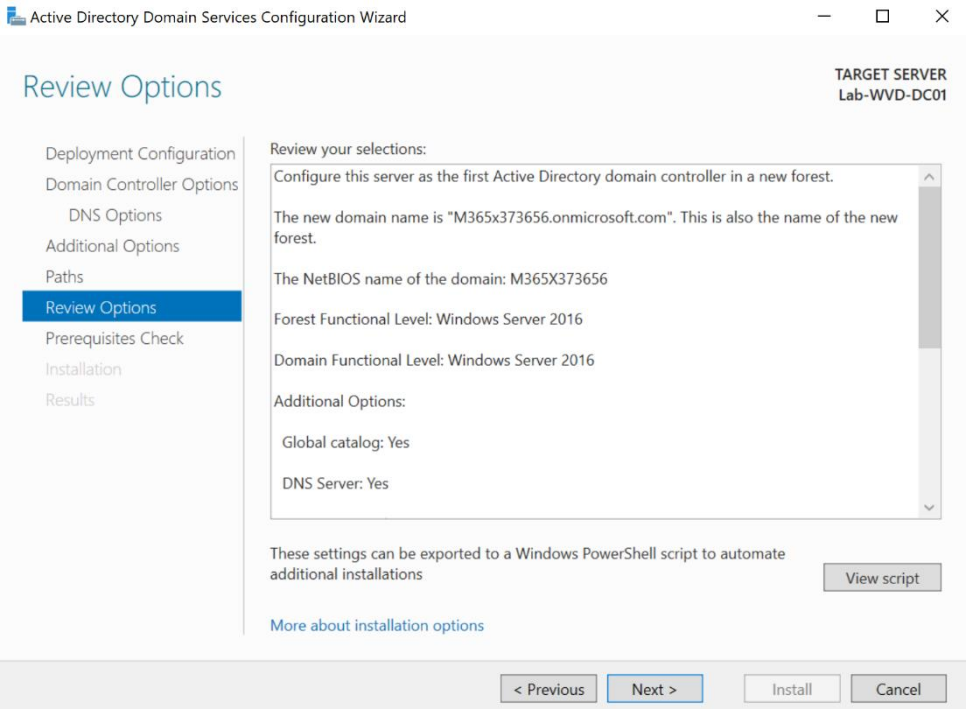
Press Next



Press Next



Press Next



Press Install

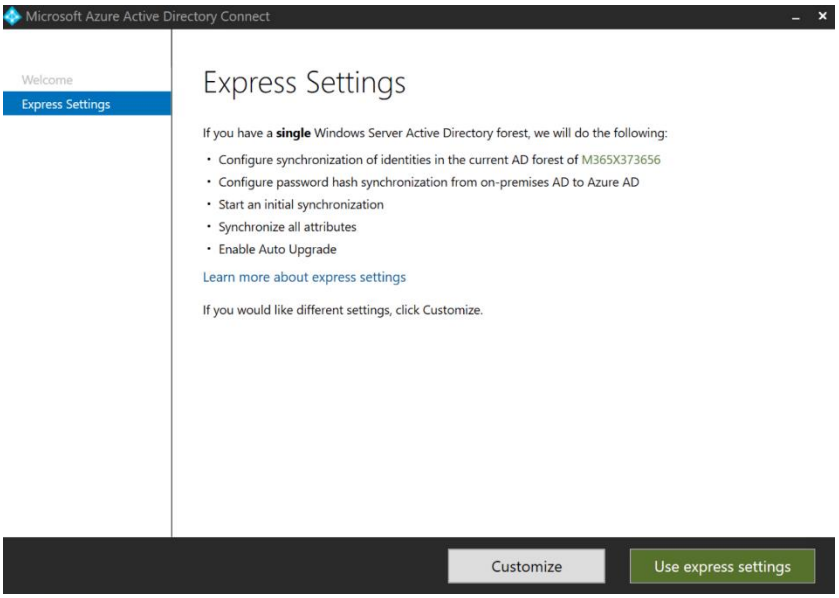
When installation is finished, the VM will reboot.

Install & Configure Azure AD Connect

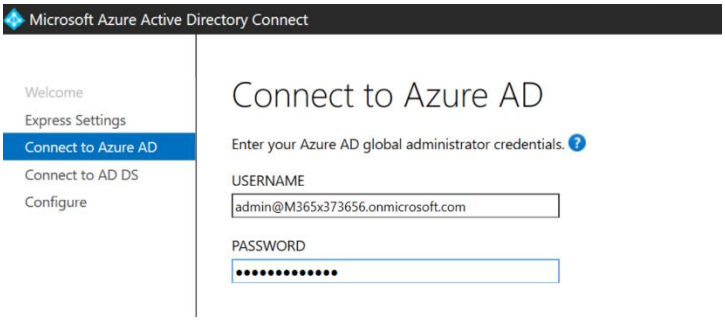
Download Azure AD Connect: https://go.microsoft.com/fwlink/?LinkId=615771 (can be easier if you download it from your computer and then copy paste to the DC VM)

Launch Azure AD Connect Installation

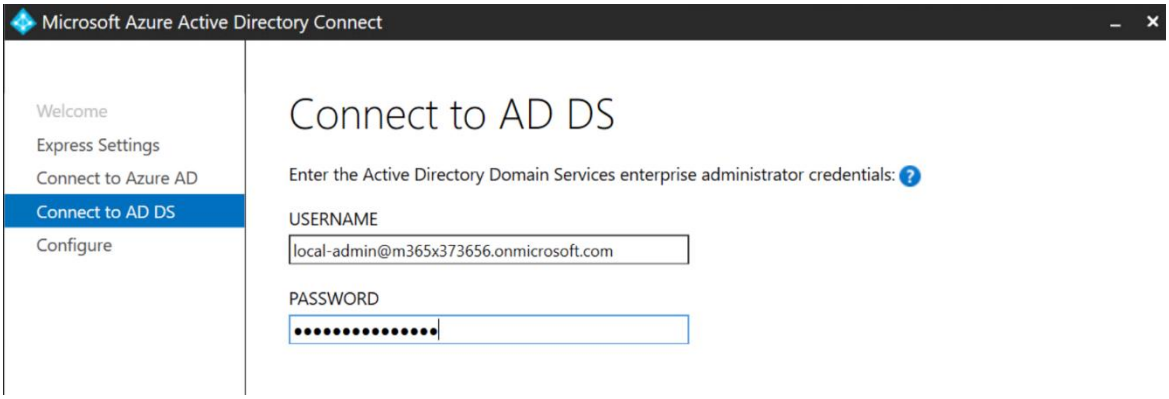
Choose "Express Settings"



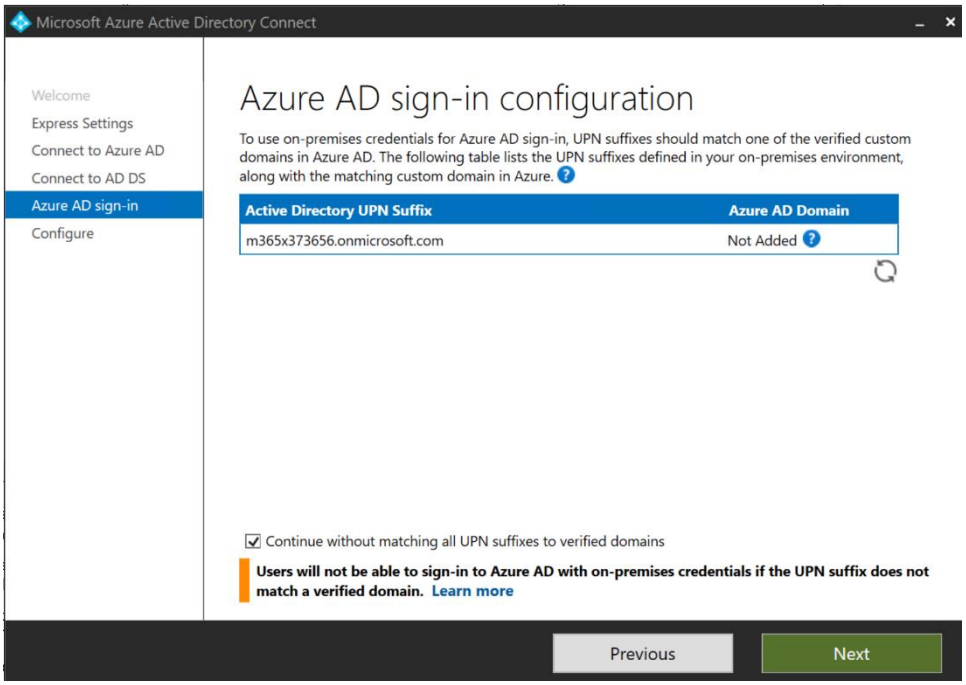
Type in your Azure AD Admin. (it can be found on the demo.microsoft.com portal)



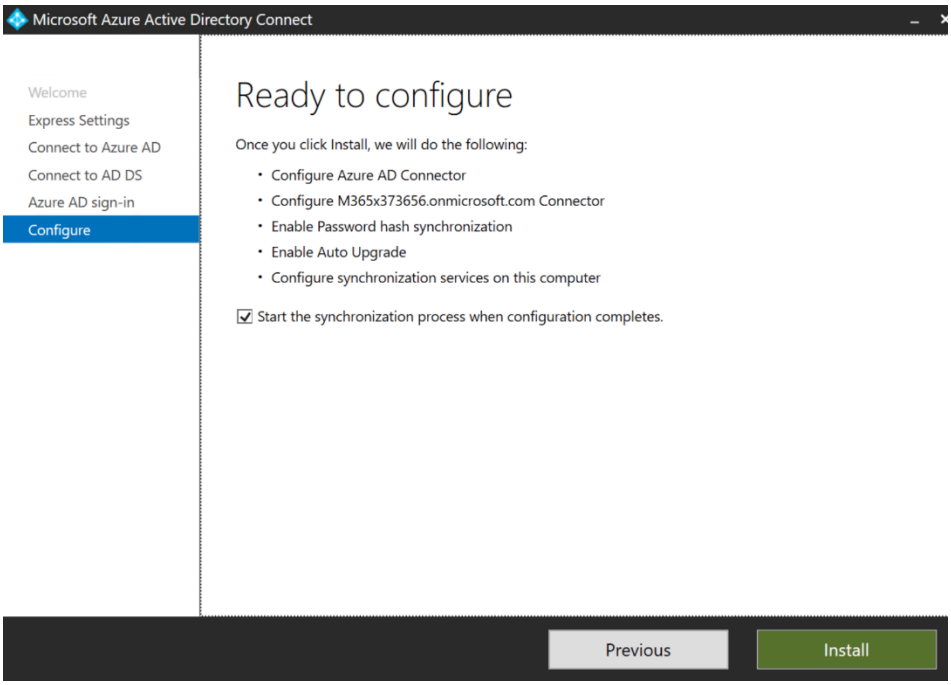
Add your Admin Domain credential. (the Local account that you have created during VM deployment should work)



Clock on the check box "Continue without matching all UPN suffices" and then "Next"



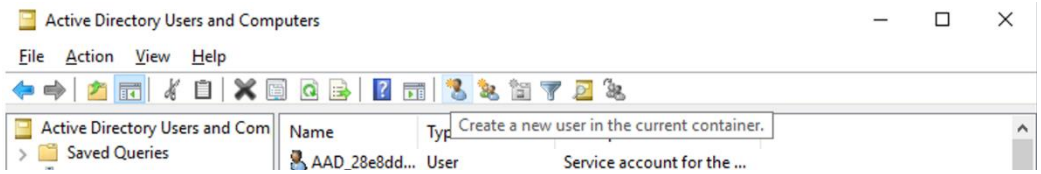
Click on "Install"



Create test users

While connected to the domain controller VM create test users.

Open **Active Directory Users and Computers** select your domain in the left hands side and expand the **Users** folder.
Use the **Create User** icon to create additional users.



Once users are created, they will need to synch to Azure AD.
Navigate to portal.azure.com and sign in with the non-AIRs subscription global admin.
Open **Azure Active Directory** blade and select **Users.** Confirm users created in AD are displayed.

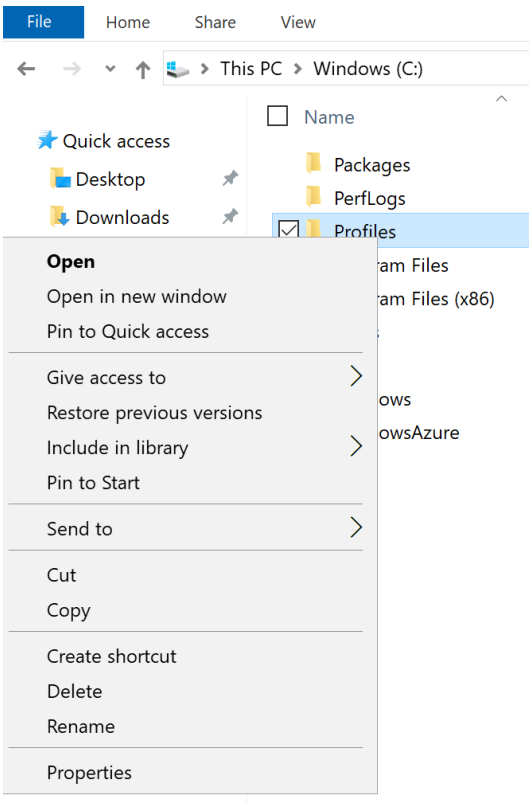
Create a SMB Share

FSLogix agent will redirect the user profile to a VHD store on a SMB share. SMB whare will be created on the DC previously created.

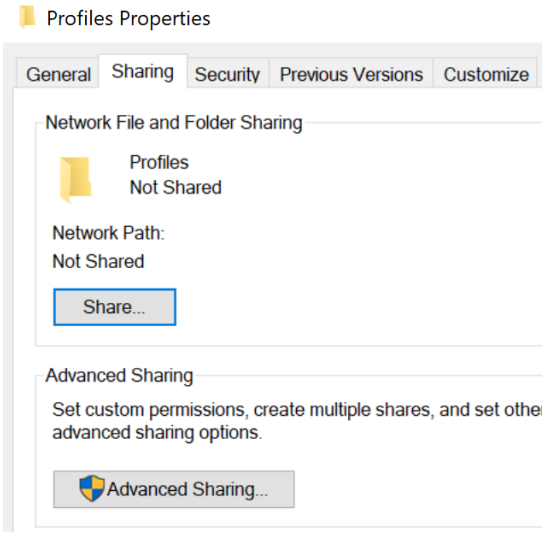
Logon on to your Domain Controller

Create “C:\Profiles”

Right Click Properties

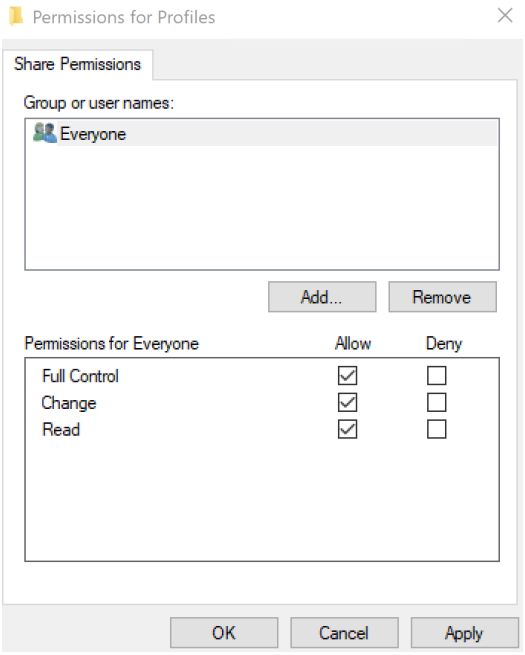


Go to “Sharing” and then “Advanced Sharing”



Select “Share this folder”, then select Permissions.

Give Full Control to “everyone” and then press “OK”



Write down your DC IP address and Network Interface name that is available in your VM azure portal.

Host pool VM will need to be in the same V-Net than the DC, therefore they need to be deployed in the same Azure region.

Deploy Windows Virtual Desktop

Please not this section is a copy of the blog by Pieter Wigleven and has only been included in the guide for reference and continuity. Please reference this blog as well as current Microsoft Docs and Techcommunity articles for the latest information.

Getting started with Windows Virtual Desktop

03-29-201903:38 PM

With the public preview for Windows Virtual Desktop now available, we wanted to provide a quick overview of the steps required to get your environment up and running.

Before I begin; however, I'd like to take a few moments to explain what Windows Virtual Desktop is and how it can help you deploy and scale Windows and Office on Azure in minutes, with built-in security and compliance.

Windows Virtual Desktop is a comprehensive desktop and app virtualization service that runs in the cloud. Here is a quick list of some of the key features and functionality:

- Infrastructure services like gateway, brokering, licensing, diagnostics are provided as a service in Azure. There's no need to deploy and maintain any on-premises infrastructure.
- Windows Virtual Desktop can leverage Azure Active Directory (Azure AD) as the identity provider, allowing you to leverage additional security controls like multifactor authentication (MFA) or conditional access.
- Once a user is connected to Windows Virtual Desktop service, access to Active Directory joined virtual machines (VMs) will be provided using Azure AD identities. In environments where Active Directory Federation Services (AD FS) is implemented for single sign-on (SSO), the user won't be prompted for credentials when connecting to the VM, providing a seamless sign-on experience.
- Reverse connect technology means your destination VM doesn't need any inbound ports to be opened. Even the default RDP port, TCP/3389, doesn't have to be open. Instead, an agent creates an outbound connection using TCP/443 into the Windows Virtual Desktop management plane. Azure is your reverse proxy for RDP traffic.
- Virtual machines in Windows Virtual Desktop are not exposed to the Internet directly. They can run using a private IP address and run isolated from other workloads or even the Internet. (The reverse connect technology allows the VMs to be accessed.)
- Windows Virtual Desktop introduces Windows 10 multi-session, allowing you to offer a Windows 10 Enterprise experience where multiple users can log into the same Windows client VM simultaneously via RDP. (Multi-session was historically only possible on Windows Server operating systems.)
- Access to FSLogix technology, making your Office experience in a non-persistent environment feel like you are using a traditional PC.
- Windows Virtual Desktop supports full desktop, RemoteApp, and persistent or non-persistent, dedicated or multi-session experiences.
- Organizations with "Windows 10 Enterprise E3 Per User" licenses or better (e.g. Windows 10 Enterprise E5 or Microsoft 365 E3, E5, F1, or Business) or RDS CALs can use Windows Virtual Desktop for no additional charge apart from Azure compute/storage and network usage billing. Reserved instances can be used to reduce Azure costs up to 80%.

Now let's move on to the steps you need to take to get started.

Windows Virtual Desktop prerequisites

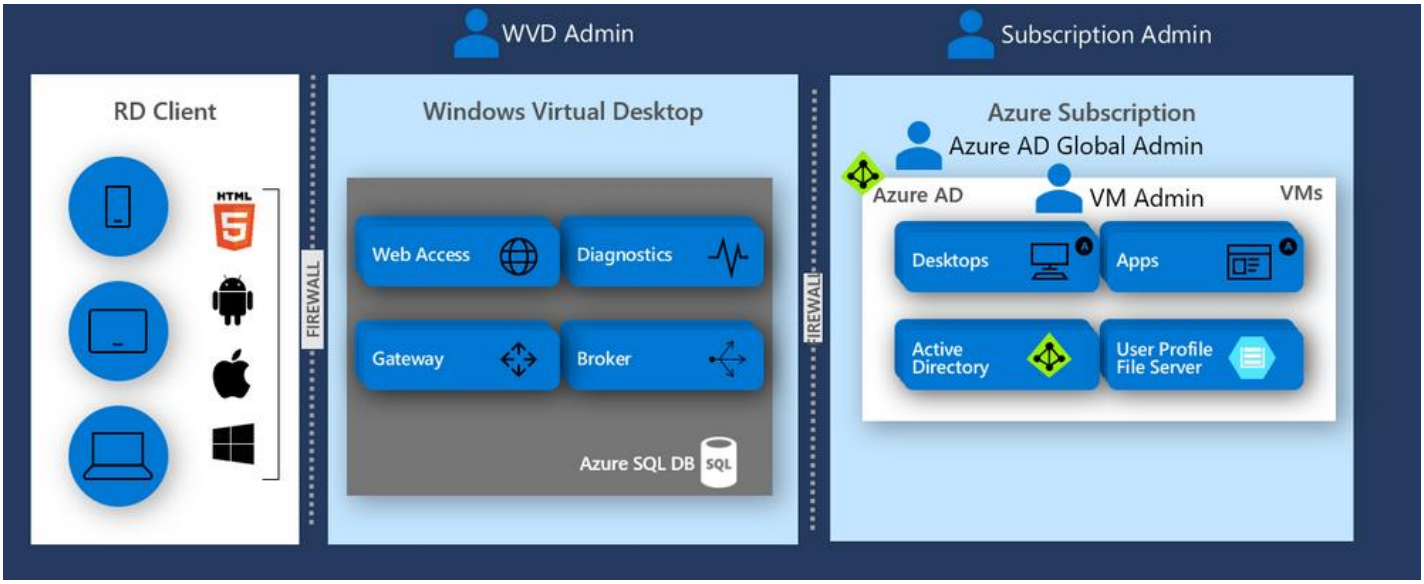
To set up Windows Virtual Desktop, you will need a few resources and to complete a few initial setup steps:

- An Azure subscription with sufficient credit (needed to host resources).
- Download and install the Windows Virtual Desktop cmdlets for Windows PowerShell on a device.
- Make sure your virtual network in Azure is configured in such a way that new VMs have your Domain Controller or Azure AD Domain Services (Azure AD DS) set as the DNS (otherwise the domain join step will likely fail). For guidance on how to configure DNS when using Azure AD DS, see Enable Azure Active Director Domain Services. For guidance for using a Domain Controller, see Name resolution for resources in Azure virtual networks.
- Make sure all Azure resources are in the same region.
- If you require seamless SSO (HTML5 client excluded), you will need AD FS or users will have to authenticate when gaining access to the VM. (Steps on how to enable this with AD FS will follow at a later stage.)
- An Active Directory to which you can join your VMs. **For this, you have three options:**

Option	Pros	Cons
Use Azure AD DS.	Great for test or isolated environments that do not need connectivity to on-premises resources. Azure AD will be your leading source for identities.	AD DS will always be running, resulting in a fixed charge per month.
Spin up a DC in your Azure subscription.	Can sync with on-premises DCs if VPN or ExpressRoute is configured. All familiar AD Group Policies can be used. Virtual machines can be paused or stopped when needed to reduce costs.	Adds additional management of a VM and Active Directory in Azure.
Use VPN or ExpressRoute and make sure your on-premises DCs can be found in Azure.	No AD DS or Domain Controller required in Azure.	Latency could be increased adding delays during user authentication to VMs. This assumes you have an on-premises environment, not suitable for cloud only tests.

In our test environment, we decided to spin up a new VM with Windows Server 2019 for our domain controller and install Azure AD Connect to sync identities to Azure AD.

Finally, you'll also need to make sure you have the right credentials. Here's an overview of the accounts being used throughout the deployment process:



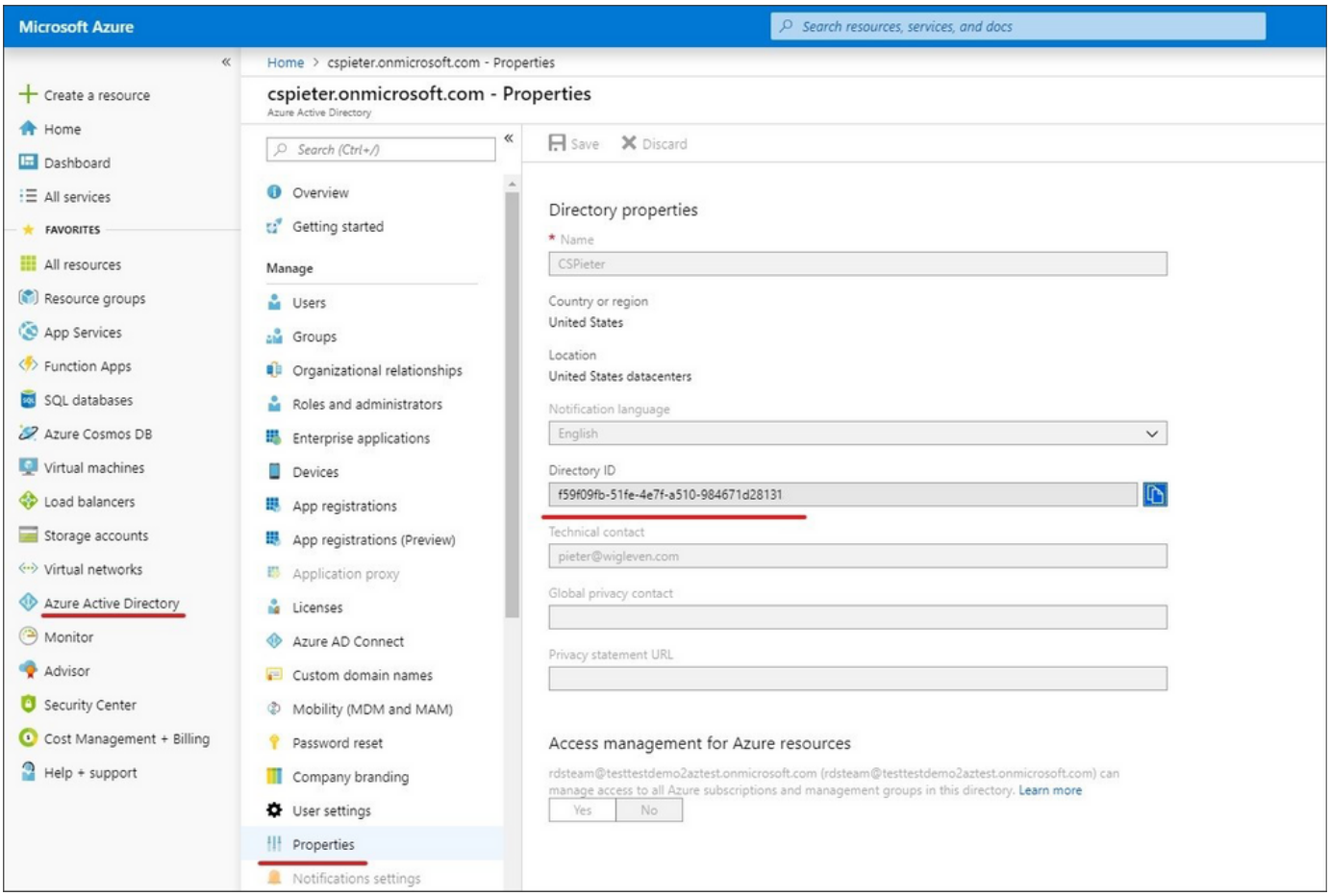
Once the prerequisites have been met, you can move on to the initial setup of Windows Virtual Desktop. Once these steps have been completed, you will be ready to deploy your initial VMs:

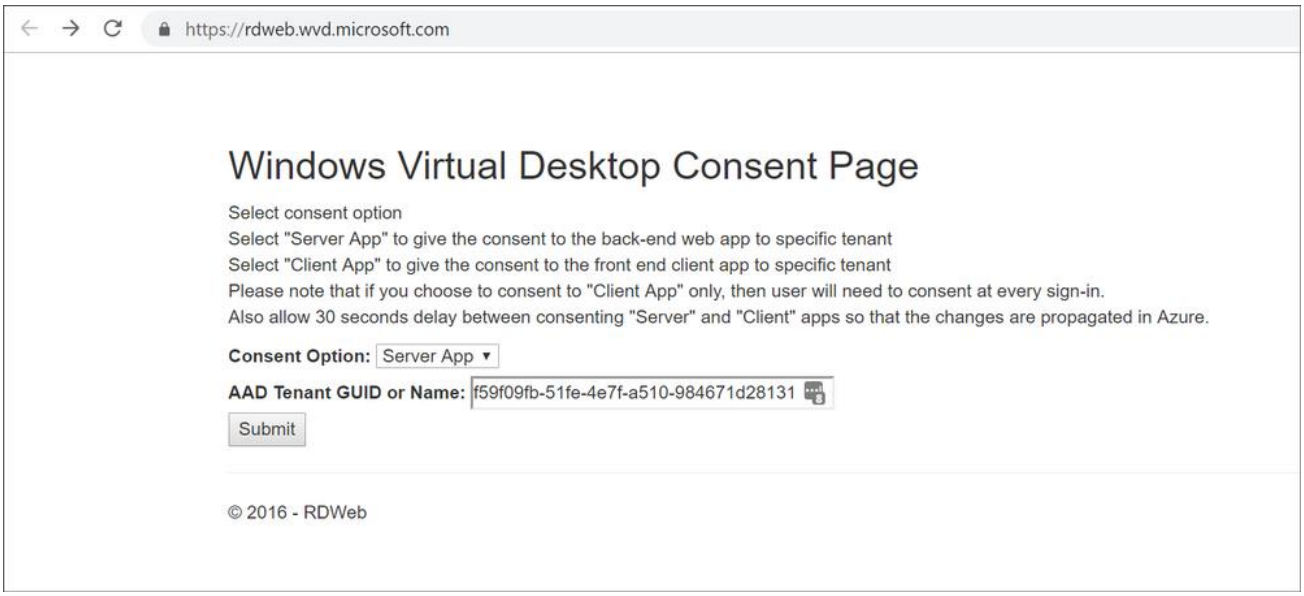
1. Allow the Windows Virtual Desktop service to access Azure AD.
2. Assign the "TenantCreator" role to a user account.
3. Create a Windows Virtual Desktop tenant.
4. Deploy your first Windows Virtual Desktop host pool.
5. Test if a user can access a full desktop session.

Allow the Windows Virtual Desktop service to access Azure AD

Before you can create a Windows Virtual Desktop tenant, you must allow Windows Virtual Desktop services to access your Azure AD tenant. The way Windows Virtual Desktop is designed requires explicit Azure AD consent. The process is much like how Azure requires you to enable non-standard resource providers before being able to use them.

1. Navigate to https://rdweb.wvd.microsoft.com.
2. Add your Azure AD tenant ID, also referred to as the Directory ID, and hit **Submit**. (Your Azure AD tenant ID can be found by visiting the Microsoft Azure Portal and navigating to **Azure Active Directory > Properties > Directory ID**, or by using whatismytenantid.com.)





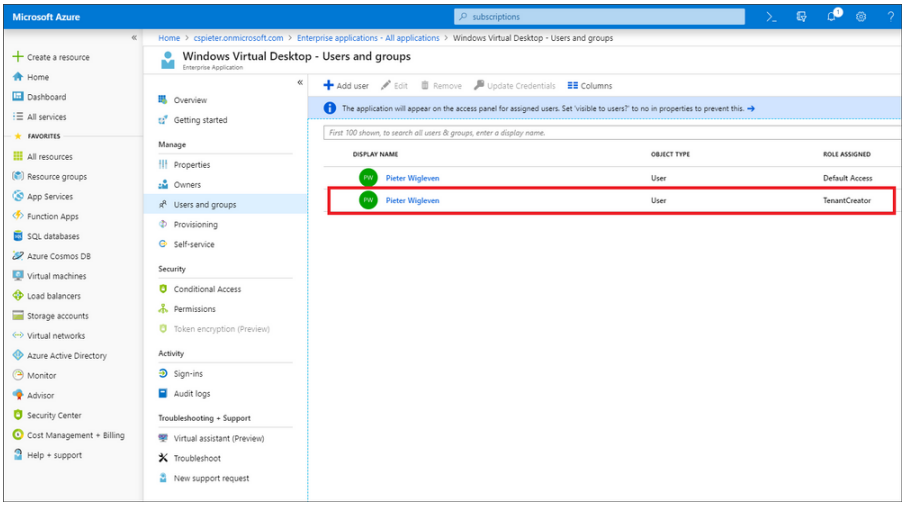
3. Wait a moment for the consent options to refresh, then change **Consent Option** to **Client App** and enter the same Azure AD tenant ID to the field for **AAD Tenant GUID or Name**. Click **Submit** to continue.

Assign the “TenantCreator” role to a user account

Once you have granted access to Azure AD, you will need to grant permissions for a user to create a Windows Virtual Desktop tenant as follows:

1. Log in to the Microsoft Azure Portal.
2. Navigate to **Azure Active Directory** from the left menu.
3. Under **Manage**, click on **Enterprise applications**.
4. Search for and select **Windows Virtual Desktop**.
5. Under **Manage**, select **Users and groups**.
6. Select **Add user**, select **Users and groups**, and search for the user to whom you want to grant permissions to perform the Windows Virtual Desktop tenant creation.
7. Select the user and hit **Select,** followed by **Assign**.

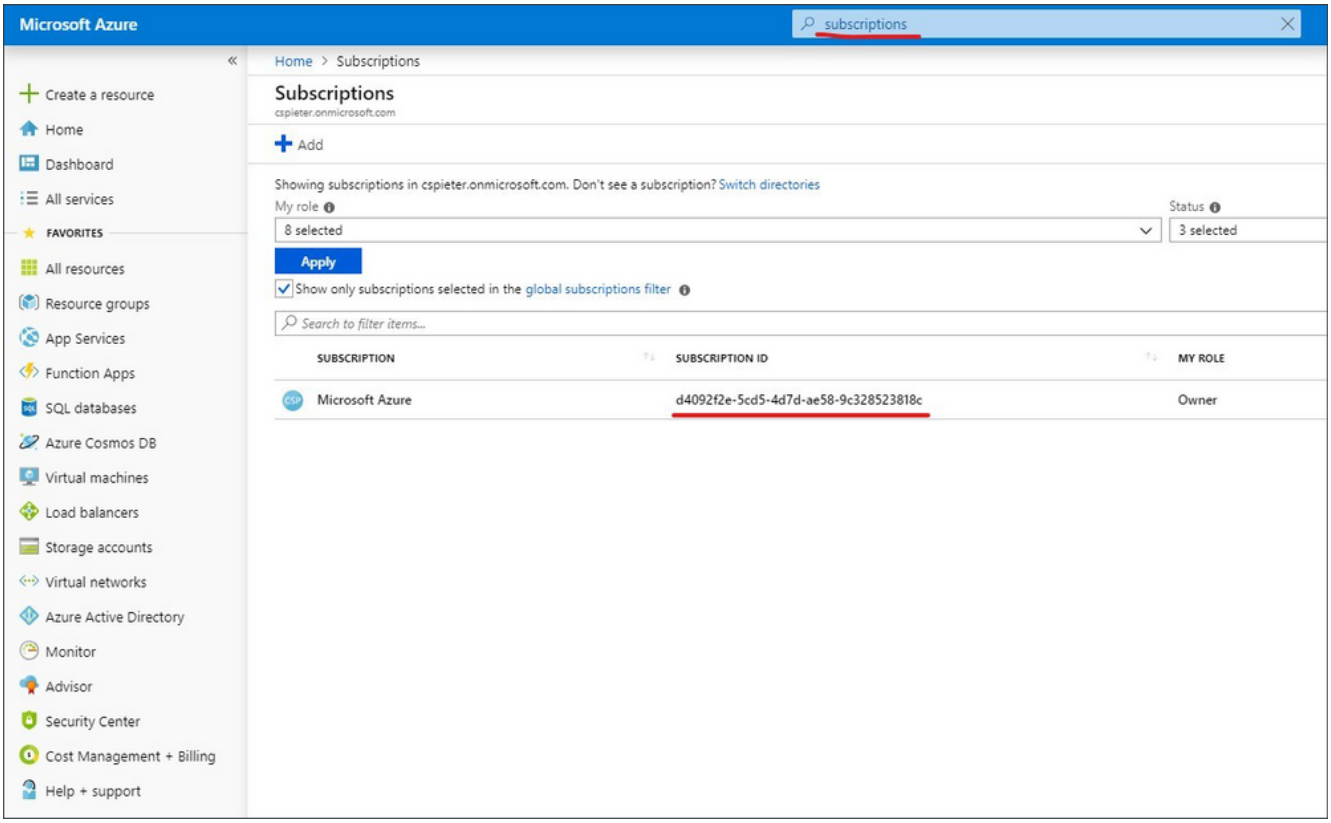
Your user should now have the role of “TenantCreator.”



Create a Windows Virtual Desktop tenant

Now that you have a user with the right permissions to create a Windows Virtual Desktop tenant, let’s go ahead and create it. During this step, you will need two IDs:

- Your Azure AD tenant ID (again).
- Your Azure subscription ID, which can be found by visiting the Microsoft Azure Portal and doing a keyword search for “Subscriptions.” Select **Subscriptions** from the search results and your subscription ID, assuming you have an active subscription, will be displayed below.



Once you have these two IDs, you can create the Windows Virtual Desktop tenant.

Note: Before proceeding, make sure you import the Windows Virtual Desktop cmdlets for Windows PowerShell as described in the prerequisites section above. If you haven't completed this step yet, see [these instructions](#).

Create a new PowerShell script, modifying the **bold** variables to reflect your tenant ID and subscription ID, and execute the following commands. When prompted, sign in using the admin account that was assigned to the TenantCreator role.

```
#Install PowerShell modules

Install-Module -Name Microsoft.RDInfra.RDPowerShell

Import-Module -Name Microsoft.RDInfra.RDPowerShell


# Setting Deployment context

$brokerurl = "https://rdbroker.wvd.microsoft.com"

$aadTenantId = <value from #1 above>

$azureSubscriptionId = <value from #2 above>

Add-RdsAccount -DeploymentUrl $brokerurl
```

Next, use the following command to create the Windows Virtual Desktop tenant.

```
New-RdsTenant -Name Contoso -AadTenantId $aadTenantId -AzureSubscriptionId $azureSubscriptionId
```

For our example, these were the commands used:

When you are prompted to login, use the following credentials:

***Username**: wvdlabadmin@silabs01.onmicrosoft.com*
***Password**: Azureteam@123*

```
# Setting Deployment context

$brokerurl = "https://rdbroker.wvd.microsoft.com"

$aadTenantId = "42920879-823b-486d-9482-aaca95b4dfda"

$azureSubscriptionId = "e8ba9b14-39ed-4dcc-91bb-f66f40a3592d"

Add-RdsAccount -DeploymentUrl $brokerurl


New-RdsTenant -Name AdminXX -AadTenantId $aadTenantId -AzureSubscriptionId $azureSubscriptionId
```

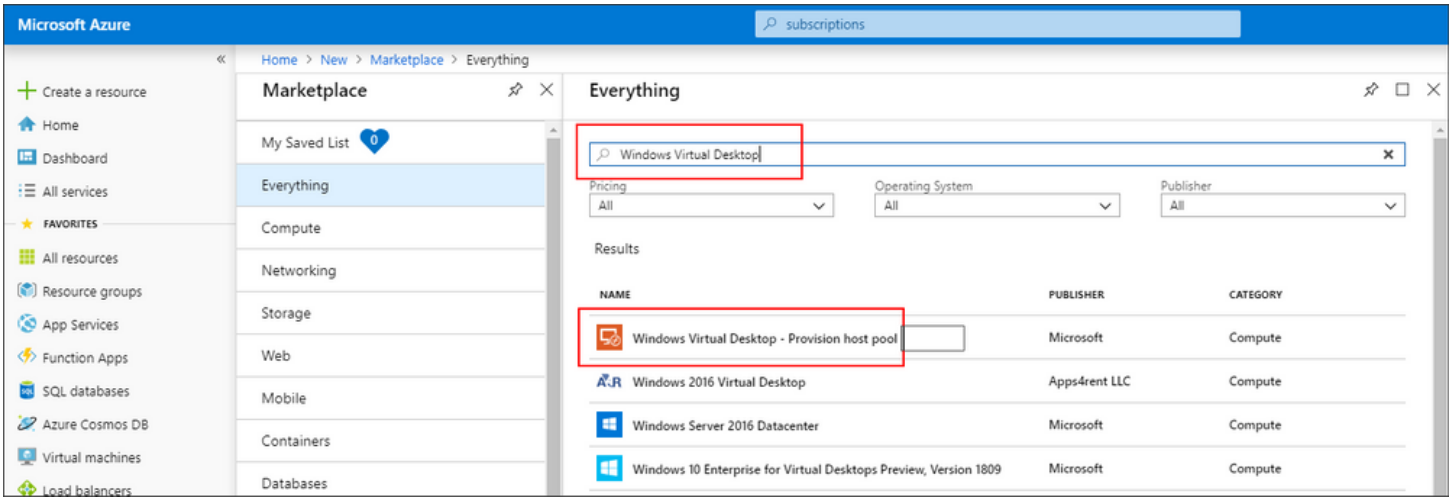
```
PS C:\WINDOWS\system32> Install-Module -Name Microsoft.RDInfra.RDPowerShell
Import-Module -Name Microsoft.RDInfra.RDPowerShell


PS C:\WINDOWS\system32> $brokerurl = "https://rdbroker.wvd.microsoft.com"
$aadTenantId = “f59f09fb-51fe-4e7f-a510-984671d28231”
$azureSubscriptionId = “d4092f2e-5cd5-4d7d-ae58-9c328523828b”


PS C:\WINDOWS\system32> Add-RdsAccount -DeploymentUrl $brokerurl


DeploymentUrl                    TenantGroupName      UserName
-------------                    ---------------      --------
https://rdbroker.wvd.microsoft.com Default Tenant Group Admin@cspieter.com
```

Deploy your first Windows Virtual Desktop host pool

A host pool is a collection of VMs that offer a similar service, such as a full desktop experience. When users connect, they can access a virtual desktop session on any of the hosts in the pool.

Step 1: Configure basic settings

To get started, visit the Microsoft Azure Portal, select **Create a Resource** and search for **Windows Virtual Desktop**. Select **Windows Virtual Desktop – Provision a host pool**.

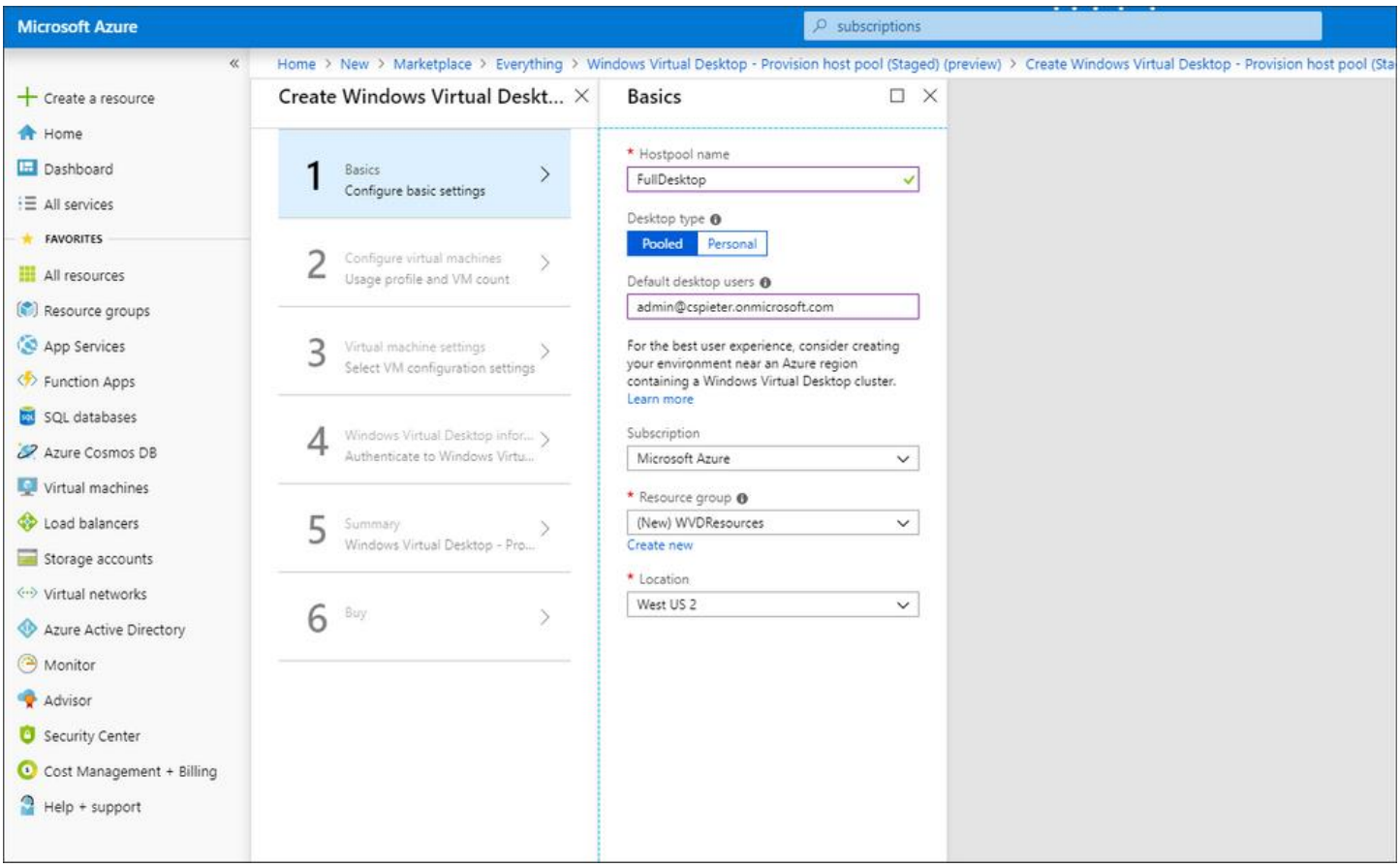


Select **Windows Virtual Desktop – Provision a host pool** and click **Create**. Enter details as follows:

- **Hostpool name** - Choose something descriptive for the pool of hosts, e.g. “FullDesktop”
- **Desktop type: Pooled** or **Personal** - Choose **Pooled** unless you are deploying a virtual desktop infrastructure (VDI) configuration wherein every user has their own dedicated VM.
- **Default desktop users** - Add a comma separated list of users. (Group support will follow later.) You can also use PowerShell to add users to this host pool at a later point.
- **Subscription** – Select **Microsoft Azure**.
- **Resource group** - enter a name to create a new one.
- **Location** - Enter the location where the resources, such as the VMs. will be created. This can be any existing Azure region of your choice.

! Important

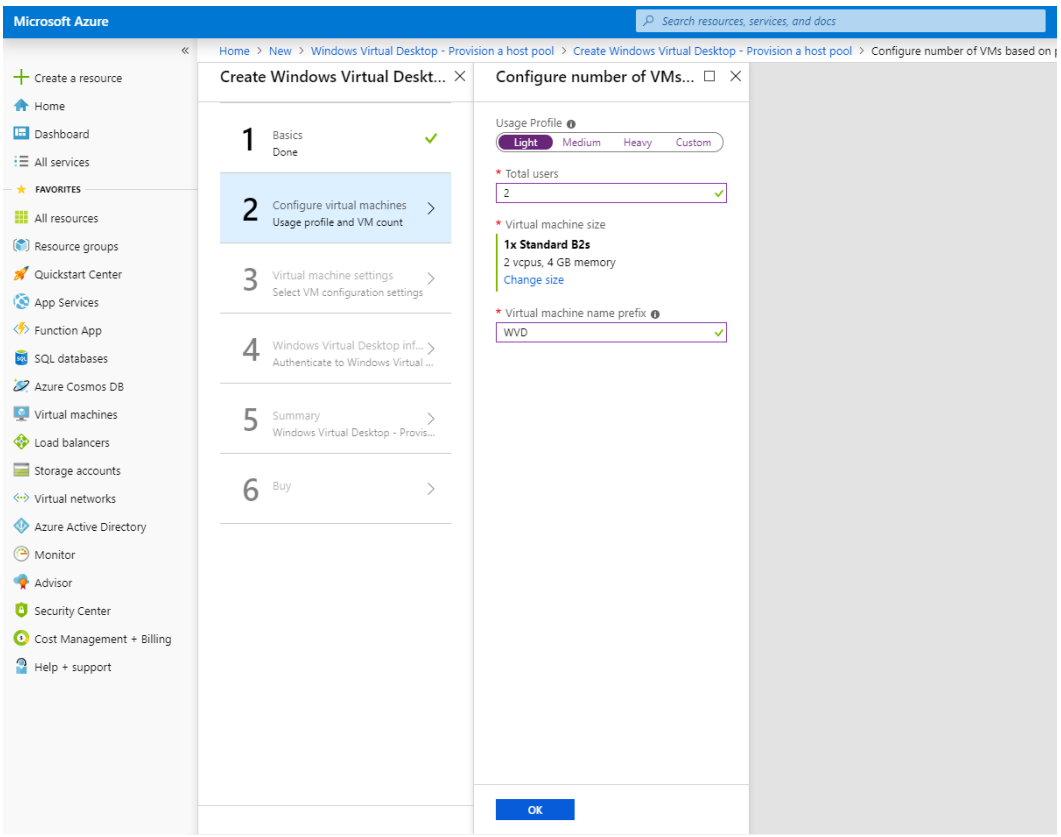
Add the domain users created on the Domain Controller that have Synced to Azure Active Directory with a comma to separate the users.



Step 2: Configure the virtual machines

Next, you'll enter the VM size details:

- Select a **Usage Profile** that matches your environment: **Light**, **Medium**, **Heavy**, or **Custom**.
- Enter the number of **Total users** that will be using this host pool.
- If desired, change the **Virtual machine size**. For your test environment, which will likely have very few users, you could opt for a smaller size. You can find additional examples and size guidance in the Windows Virtual Desktop pricing guide.
- Add a prefix for the VMs. Note: make sure your prefix is unique; don't reuse your prefixes.

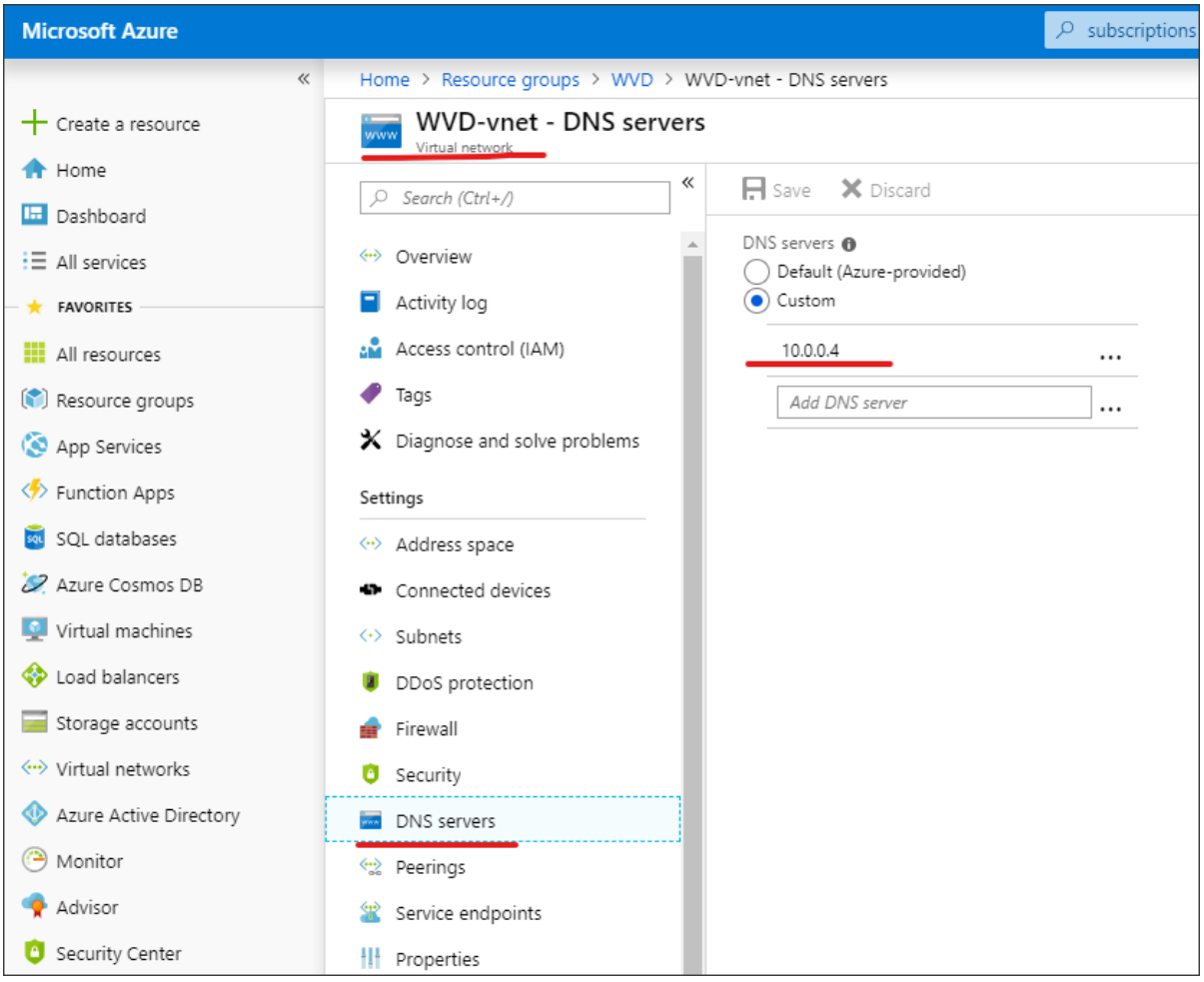


Step 3: Configure VM settings

To configure the VMs for Azure, you will need to:

- Select a custom image from **Blob storage**, a **Managed image** in Azure, or one from the **Gallery**. Our recommendation would be to test "Windows 10 Enterprise multi-session with Office 365 ProPlus" from the Azure Gallery. Office 365 ProPlus has been preconfigured for the ideal state of Windows 10 multi-session.
- Select the **Image OS**

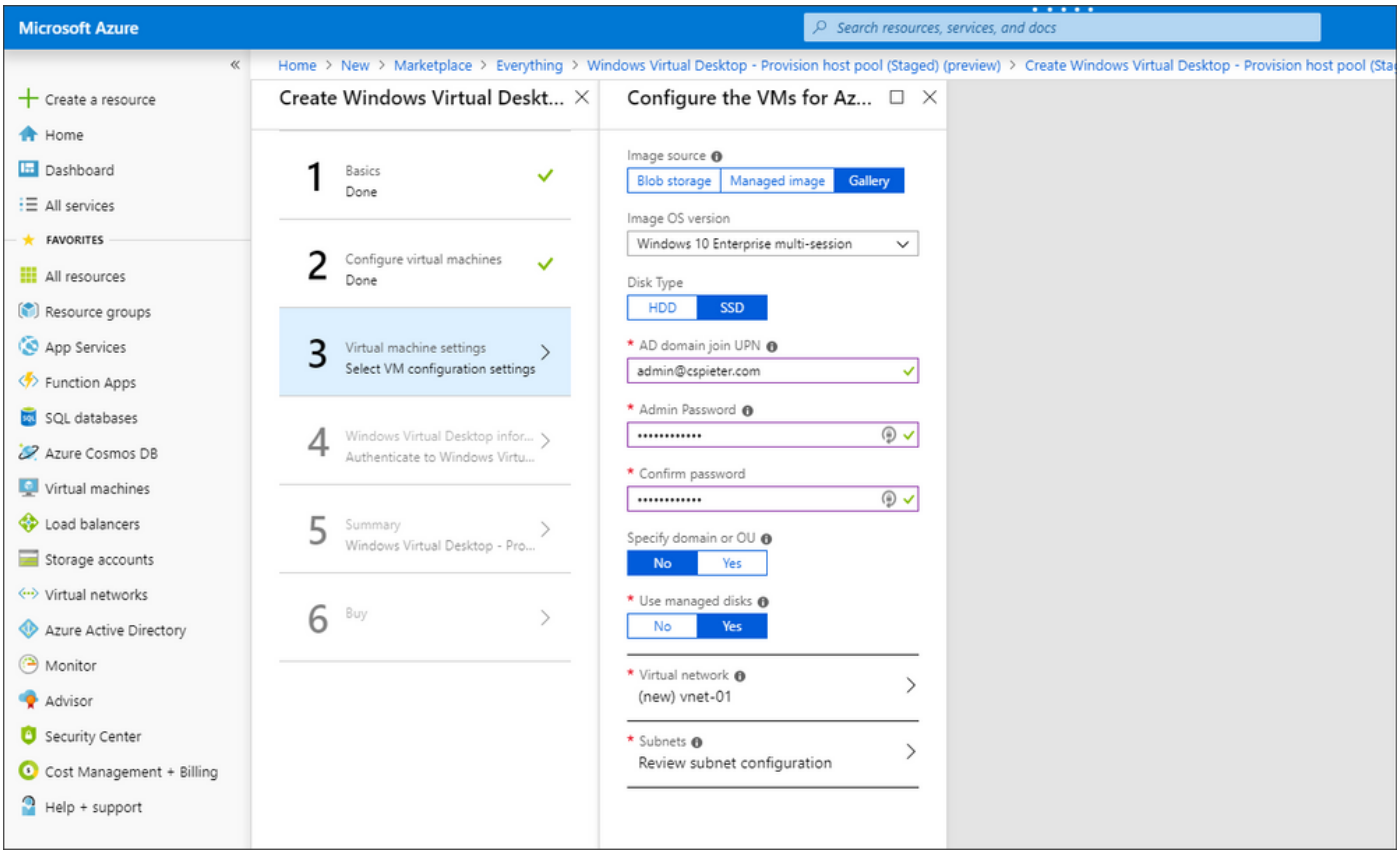
- Select the **Disk Type**. HDD is recommended for use with MDSN Subscription.
- Enter credentials that have permissions to join a VM to Active Directory.
- (Optional) Specify the domain and/or OU.
- (Optional) Use managed disks.
- Configure the virtual network and subnet. Pay close attention to this step as this wizard will spin up virtual machines and join them to AD. This means the virtual machine must be able to locate the Domain Controller. Consequently, we recommend opening a separate tab in your browser and validate that:
    - The DNS server IP address that is assigned to the VM points to the DC or AD DS; this can be configured in multiple locations including on your virtual network.
    - The DC, VM, and network resources are in the same Azure region. (Otherwise, your deployment is likely to fail.)



Here an example of what step 3 of the wizard could look like:

! Important

To keep cost down you may want to use HDD as the Disk Type

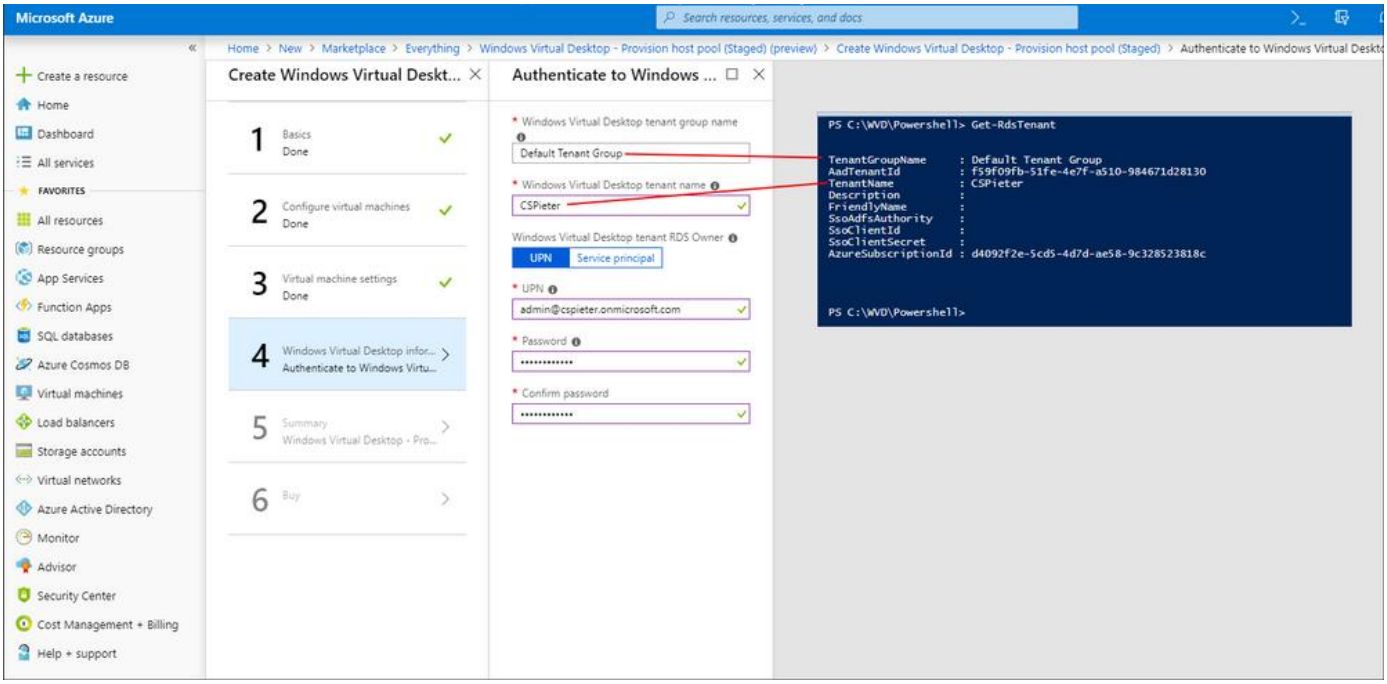


Step 4: Enter authentication details

Once you have configured your VM settings, you will need to enter details about your Windows Virtual Desktop tenant and Azure AD tenant. Unless otherwise directed, leave the **Windows Virtual Desktop tenant group name** as "Default Tenant Group." For the Windows Virtual Desktop tenant name, enter the name of the tenant you created earlier in this process.

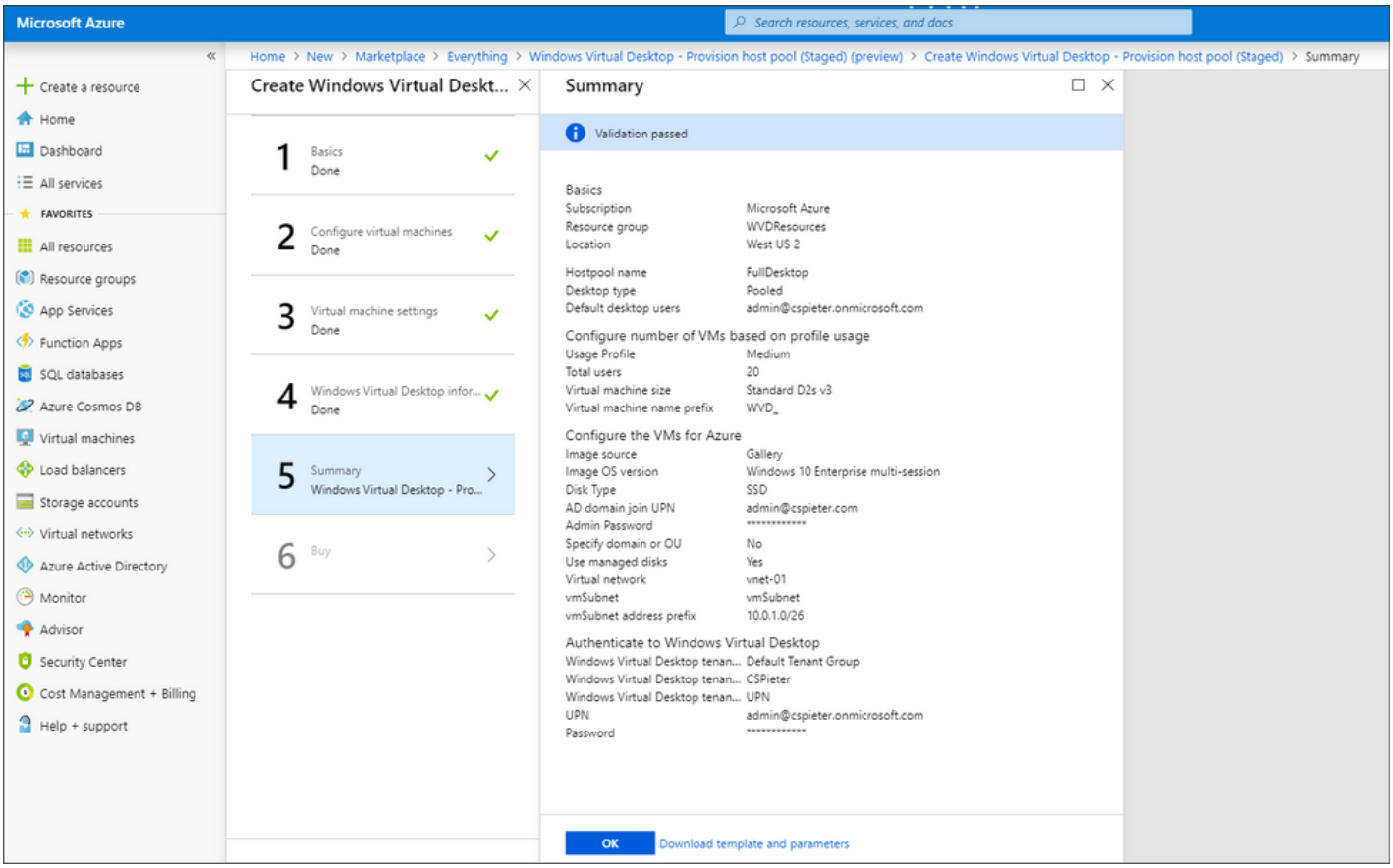
Note: If you are unsure what your Windows Virtual Desktop tenant name is, use the PowerShell command "Get-RdsTenant" to obtain it. (Note: Cmdlet does not exist natively.)

Enter valid credentials for your Azure AD environment (UPN and password).



Step 5: Check the summary

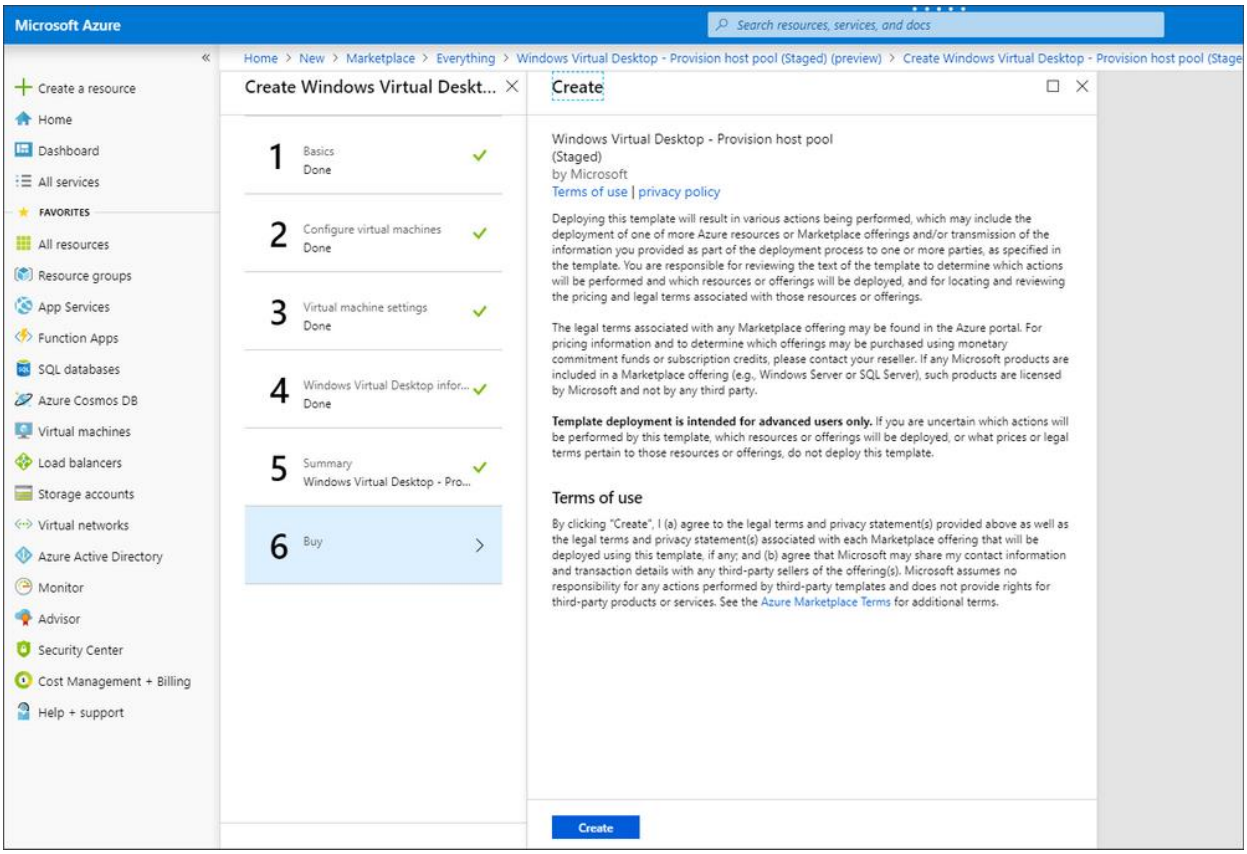
Check the summary windows to see your setup passed validation, then click **OK**.



Step 6: Finalize the creation of your host pool

Hit **Create**, sit back, and relax. Wait for the deployment to finish. The process takes roughly 20 minutes.

> **! Note**
>
> This process can take 40 minutes or more to complete. Watch the Notifications Bell for status updates.

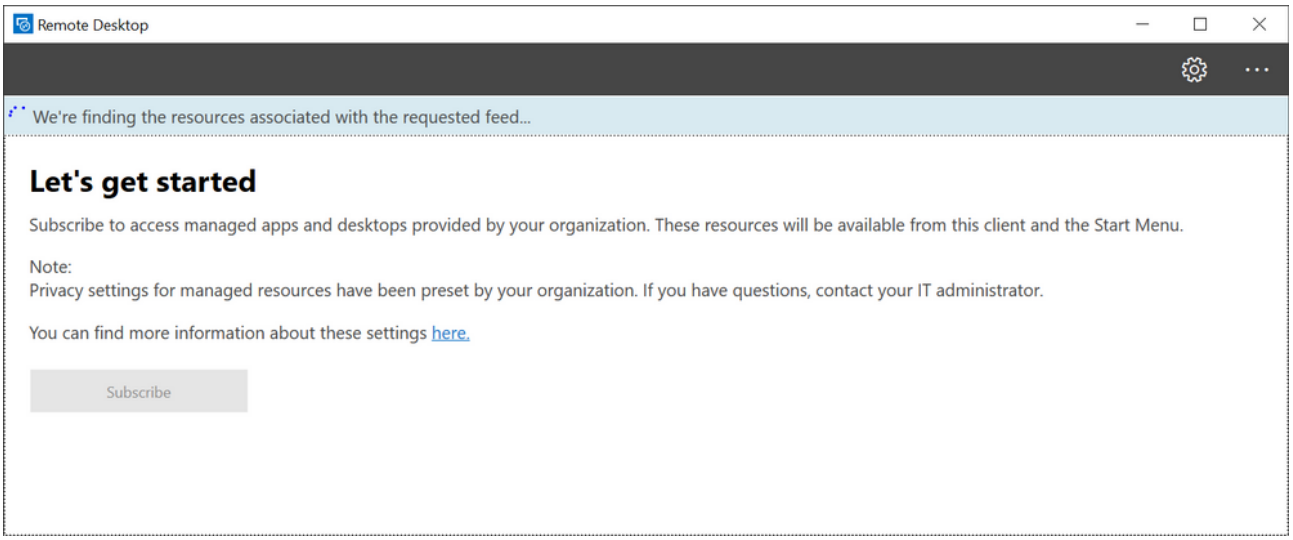


Test if a user can access a full desktop session

Once you have created your Windows Virtual Desktop host pool, you can download the client for Android or Windows, or use the HTML5 client. (The Microsoft Remote Desktop Beta for iOS can be tested using TestFlight.) Here's how to test with Windows or the HTML5 client.

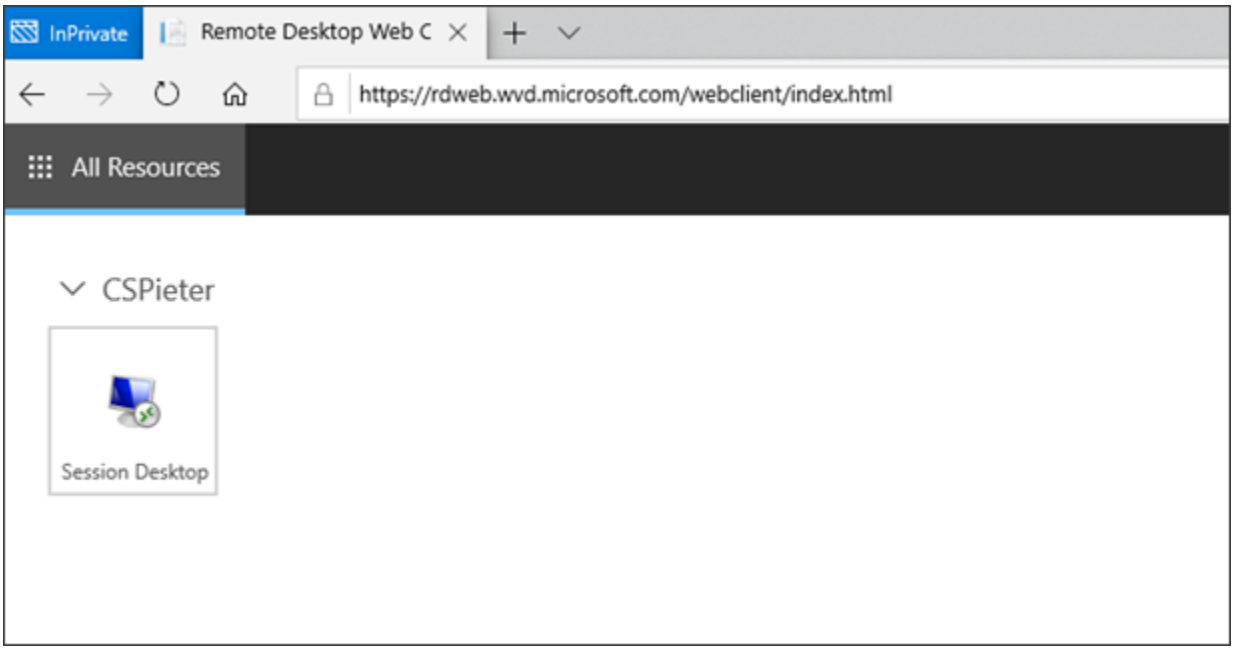
Test with the Windows client

Download the latest Windows Remote Desktop client and subscribe to the feed using the following URL: **https://rdweb.wvd.microsoft.com.** Once subscribed, you will find the virtualized apps and desktops in the Start menu. You'll also notice that it's possible to enable conditional access and/or MFA for users when subscribing to a feed.



Test with the HTML5 client

Launch a browser in privacy or incognito mode and visit http://aka.ms/wvdweb to access the HTML5 client. Authenticate using the login information to which you assigned a full desktop session.



! Note

In environments where Active Directory Federation Services (AD FS) is implemented for single sign-on (SSO), the user won't be prompted for credentials when connecting to the VM, providing a seamless sign-on experience. (**Using the Domain Controller in Azure you will need to login twice, but it helps to keep cost down.)**

If you are successful, you should be able to view the desktop:

What's next?

Once you have completed your setup of Windows Virtual Desktop, you can assign other users to your host pool using the following PowerShell command, replacing <WVDTENANTNAME> with the name of your tenant, <HOSTPOOLNAME> with the name of your host pool, and leveraging the appropriate user principal name:

```
Add-RdsAppGroupUser <WVDTENANTNAME> <HOSTPOOLNAME> "Desktop Application Group" -UserPrincipalName
USER@TENANT.onmicrosoft.com
```

In our environment, this is what the command looks like:

```
Add-RdsAppGroupUser AdminXX FullDesktop "Desktop Application Group" -UserPrincipalName userXX@sila
bs01.onmicrosoft.com
```

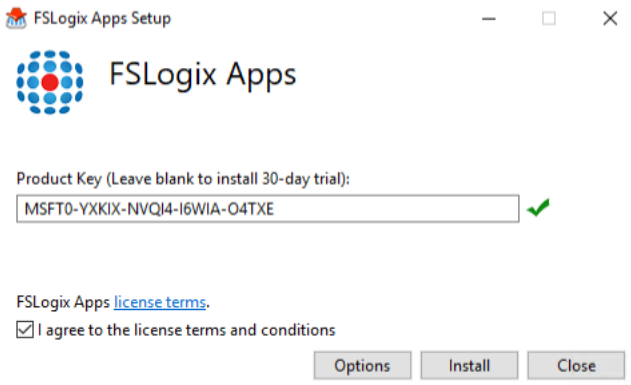
You can also move on to more advanced tasks, such as:

- Setting up a host pool dedicated for RemoteApps instead of full desktops.
- Installing FSLogix profile containers so that you can benefit from the profile solution that makes Office365 ProPlus work, just like on a local PC and/or laptop.

To explore other scenarios and learn more about Windows Virtual Desktop, please see the Windows Virtual Desktop documentation on Docs and watch What is Windows Virtual Desktop? on Microsoft Mechanics. I also encourage you to join the Windows Virtual Desktop community on Tech Community to connect with the Windows Virtual Desktop team and your fellow public preview participants.

FSLogix Agent Installation

1. Connect to the virtual machine with the credentials you provided when creating the virtual machine.
2. Launch an internet browser and navigate to this link to download the FSLogix agent. As part of the Windows Virtual Desktop public preview, you'll get a license key to activate the FSLogix software. The key is the LicenseKey.txt file included in the FSLogix agent .zip file.
3. Navigate to either \\Win32\Release or \\X64\Release in the .zip file and run **FSLogixAppsSetup** to install the FSLogix agent.
4. Type in the license key that can be found in installation files



5. Navigate to **Program Files** > **FSLogix** > **Apps** to confirm the agent installed.
6. From the start menu, run **RegEdit** as an administrator. Navigate to **Computer\HKEY_LOCAL_MACHINE\software\FSLogix\Profiles**
7. Create the following registry keys

Name	Type	Data/Value
Enabled	DWORD	1
VHDLocations	Multi-String Value	"Network path for file share"
VolumeType	String	VHDX
SizeInMBs	DWORD	"integer for size of profile" for exemple 3000
IsDynamic	DWORD	1
LockedRetryCount	DWORD	1
LockedRetryInterval	DWORD	0
RoamSearch	DWORD	2
DeleteLocalProfileWhenVHDShouldApply	DWORD	1
ProfileType	DWORD	3

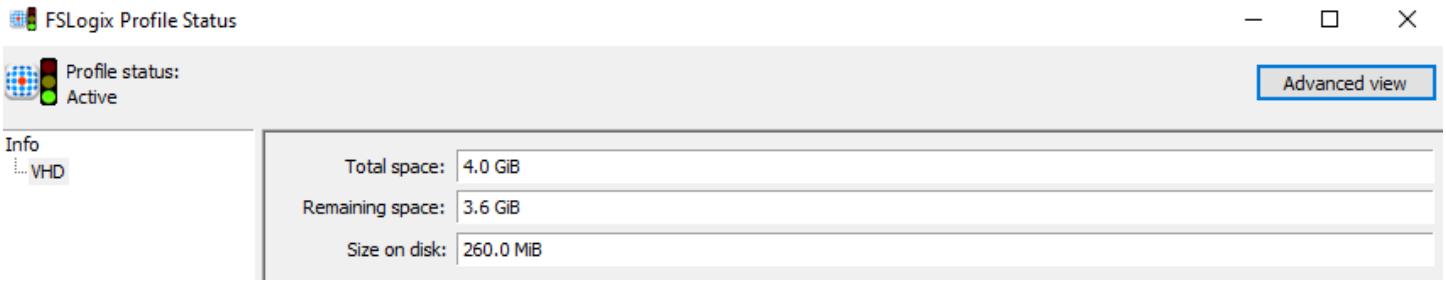
For VHDLocations, use the share folder previously created on the DC.

If you want to know what the settings are for go check this web page :
https://docs.fslogix.com/display/20170529/FSLogix+Profiles+Configuration+Settings

You can get a status if FSLogix agent works by launching FRXTray that in: “Program Files\FSLogix\Apps”

Then click on the traffic light that is in the tray:

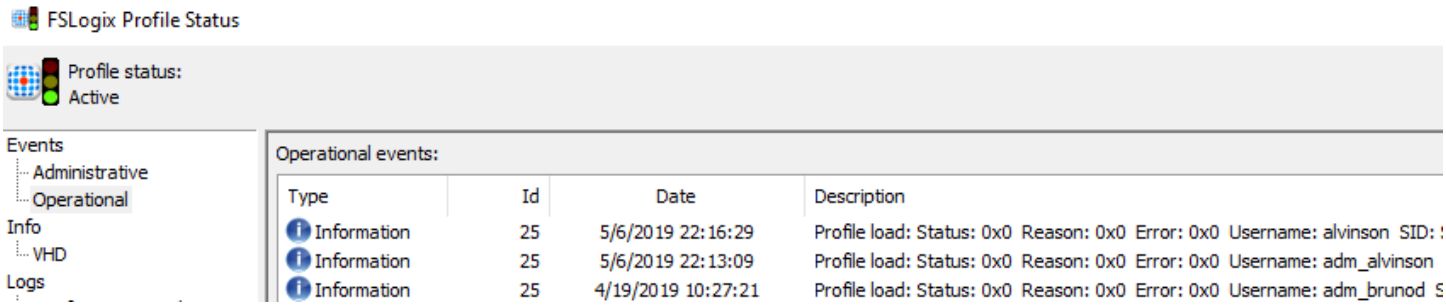
First of all, if everything work fine, traffic light should be green



Click on “Advanced view”

Go to “Operational” and look if you have no issue in the logs.

Here is an example. If issue, just go through the logs.



Install the OneDrive Client in Per Machine Mode

! Note

OneDrive Client Per Machine updated to match the online documentation 6/13/2019

https://docs.microsoft.com/en-us/onedrive/per-machine-installation

By default, the OneDrive sync client installs per user, meaning OneDrive.exe needs to be installed for each user account on the PC under the %localappdata% folder. With the new per-machine installation option, you can install OneDrive under the "Program Files (x86)" directory, meaning all profiles on the computer will use the same OneDrive.exe binary. Other than where the sync client is installed, the behavior is the same

This guide will show you how to install OneDrive on a Windows 10 Multi Session VM running inside the Windows Virtual Desktop Service

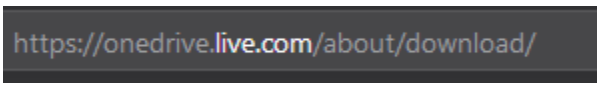
You can find instructions on OneDrive Per Machine Installation here https://docs.microsoft.com/en-us/onedrive/per-machine-installation

GUIDE

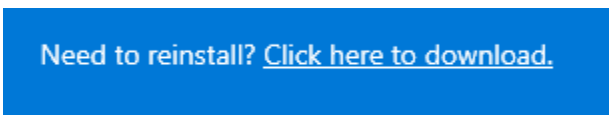
Connect to your Windows 10 Multisession Client using the web browser or an RDP Client

Open the Web Browser on your Windows 10 Multi Session machine

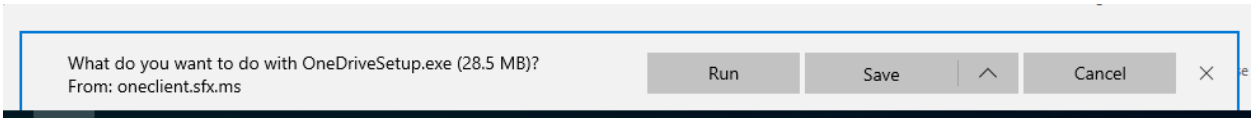
Once at your desktop open a browser on the Win 10 multis session VM and navigate to https://onedrive.live.com/about/download/ and hit enter. This is where you can download the OneDrive Client you need to install



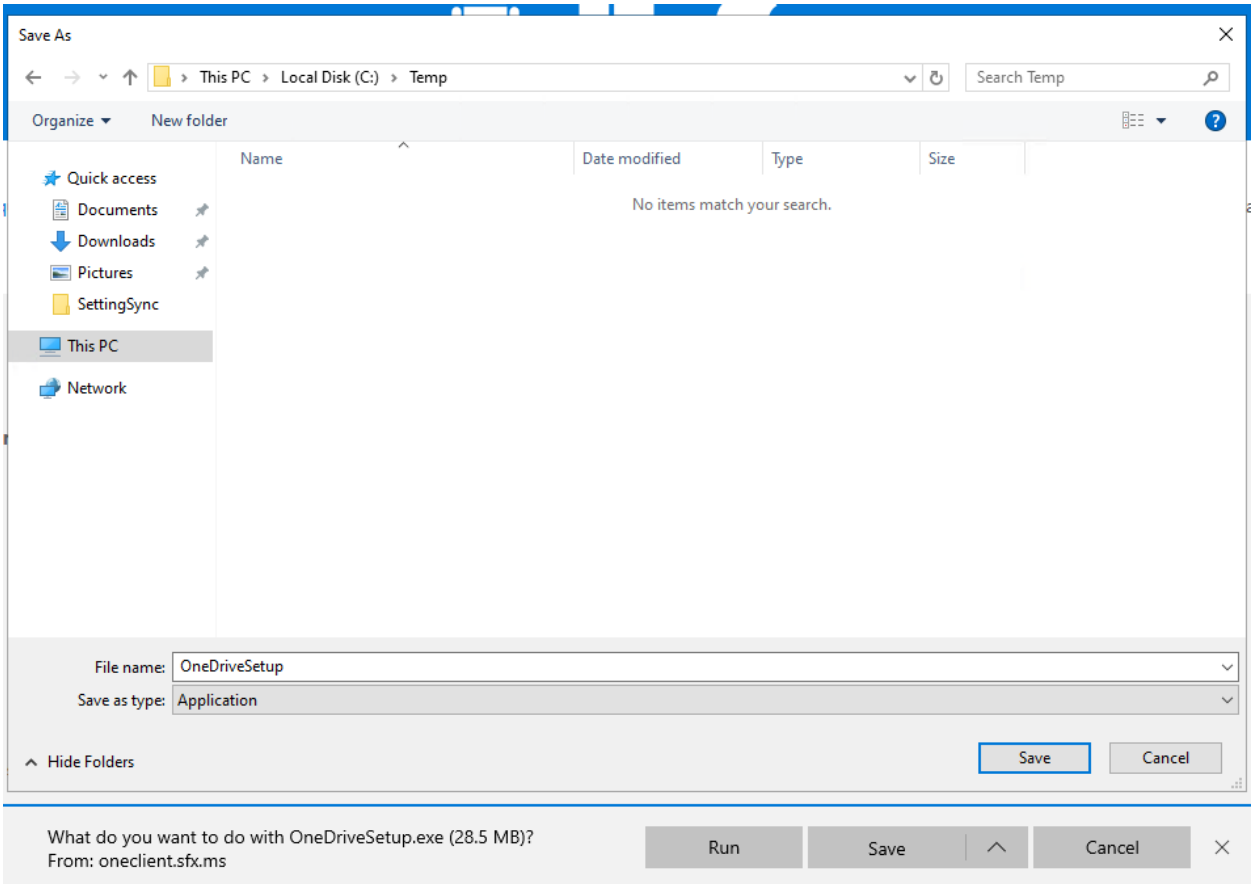
Close to the Icon that says, "Start OneDrive" is a hyperlink that says, "Click here to download". Click on it to start the download of the OneDrive client

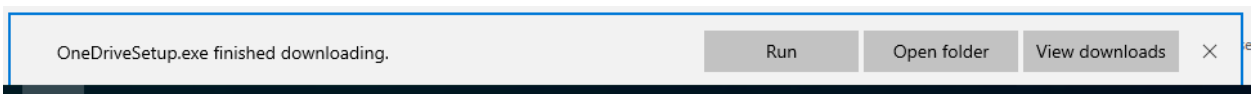


This should kick off a download which you can see at the bottom of your browser

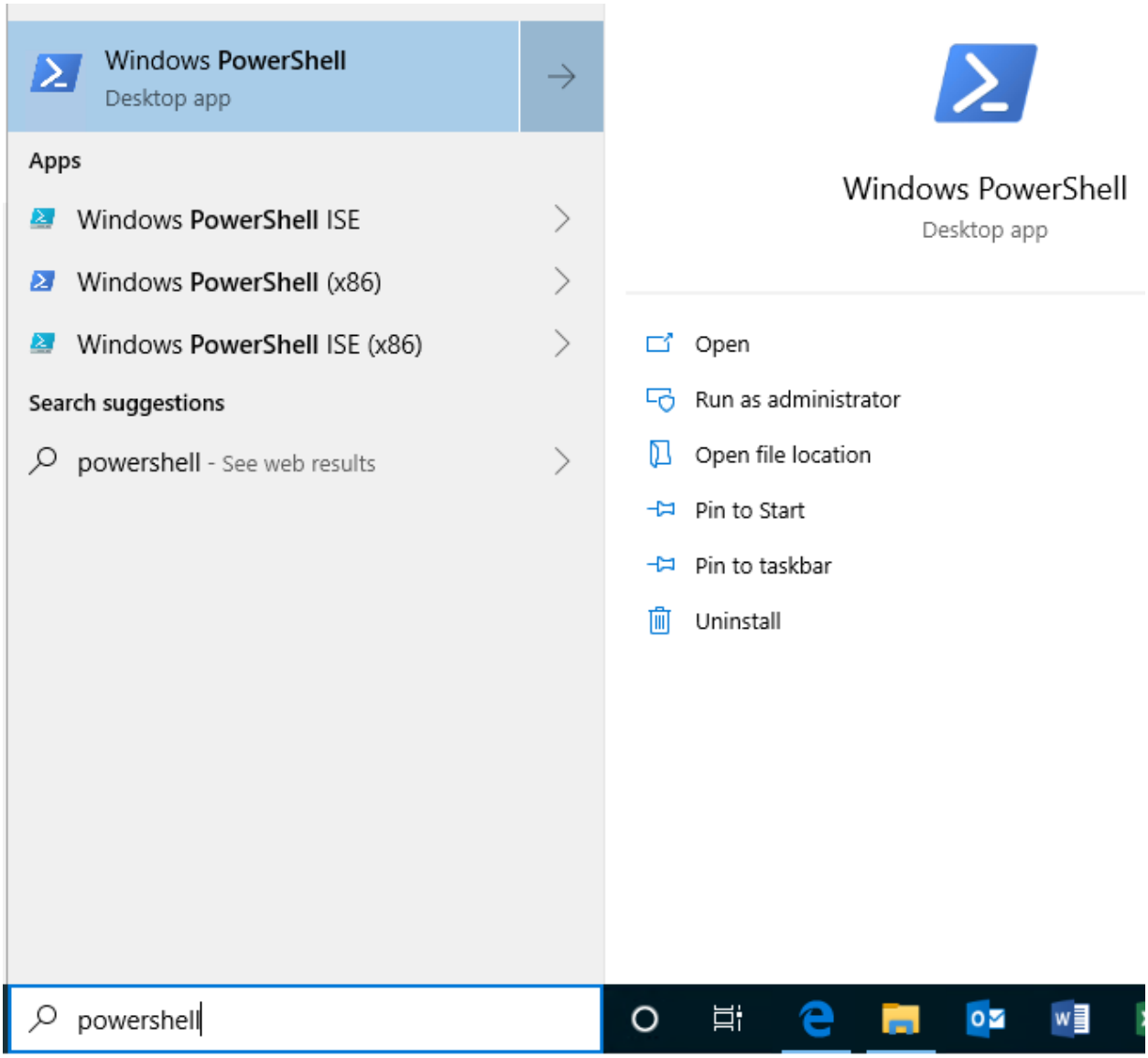


Click on the arrow next to Save and Click on "Save as" and choose a location where you want to save it. Note down the location



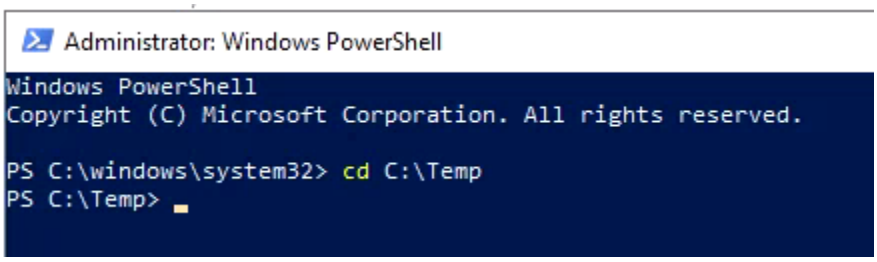


When you see the download complete open an elevated PowerShell window by going to "Start" typing in PowerShell and then choosing "Run as Administrator"



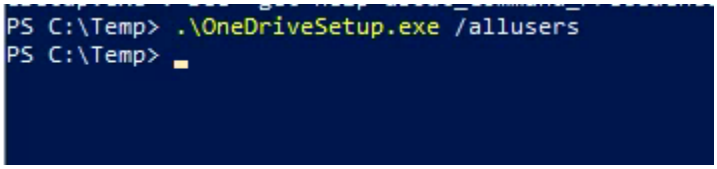
Change the location to where you saved the OneDrive Setup executable using the syntax:

CD <filepath>

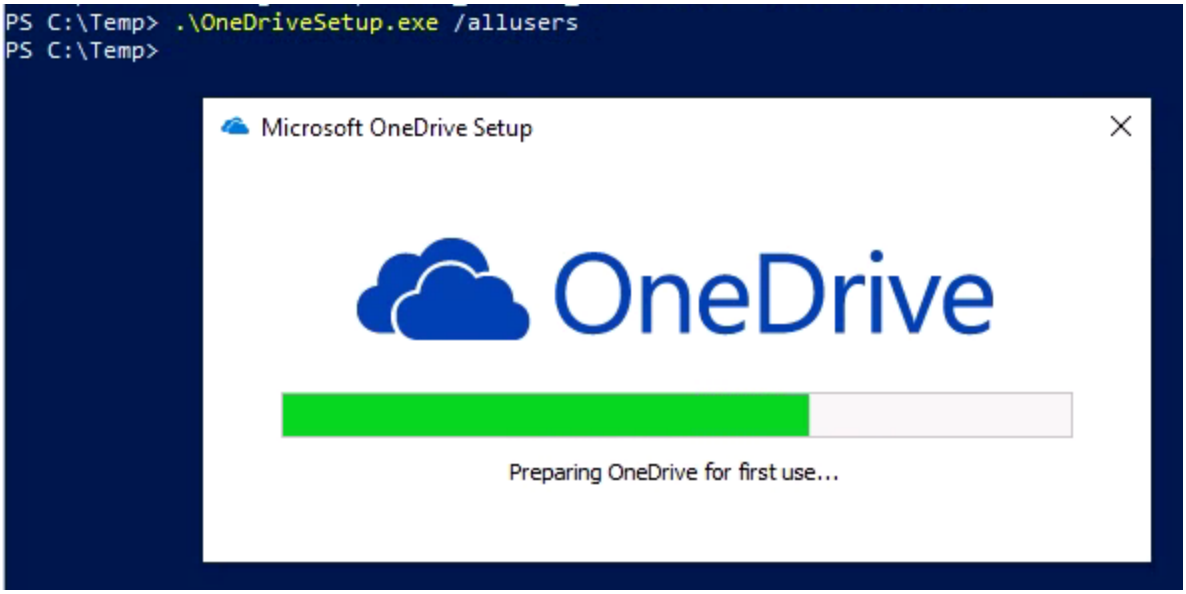


To install OneDrive per machine we need to install it with the "All Users" command. Type in this command and hit enter

.\OneDriveSetup.exe /allusers



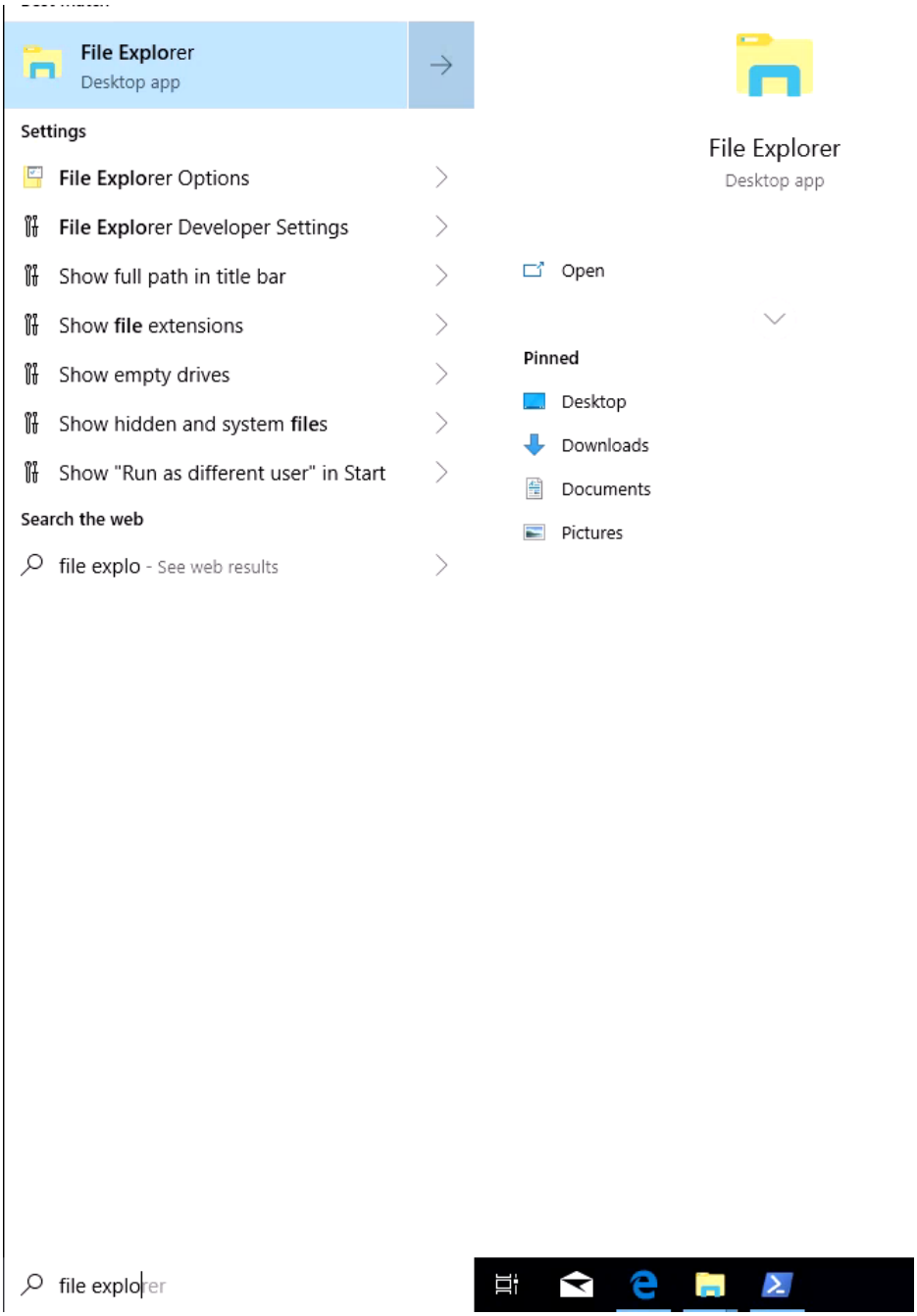
This might result in a UAC prompt which you should accept. You'll see a Dialog box telling you OneDrive is being prepared. I did this on a machine where no one had logged into OneDrive so your results may vary if you've logged into OneDrive before



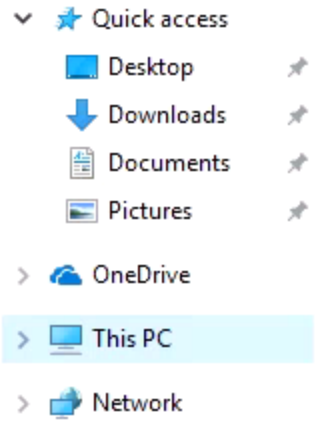
When setup completes, OneDrive should be installed. In my setup I had not logged in to OneDrive before, so I had to do the initial setup. If you were already logged into OneDrive on this machine then those accounts should be added on the computer automatically.

If you already had accounts added to this computer and you don't need to add any you are done. If you need to setup a new account on this machine, then keep reading

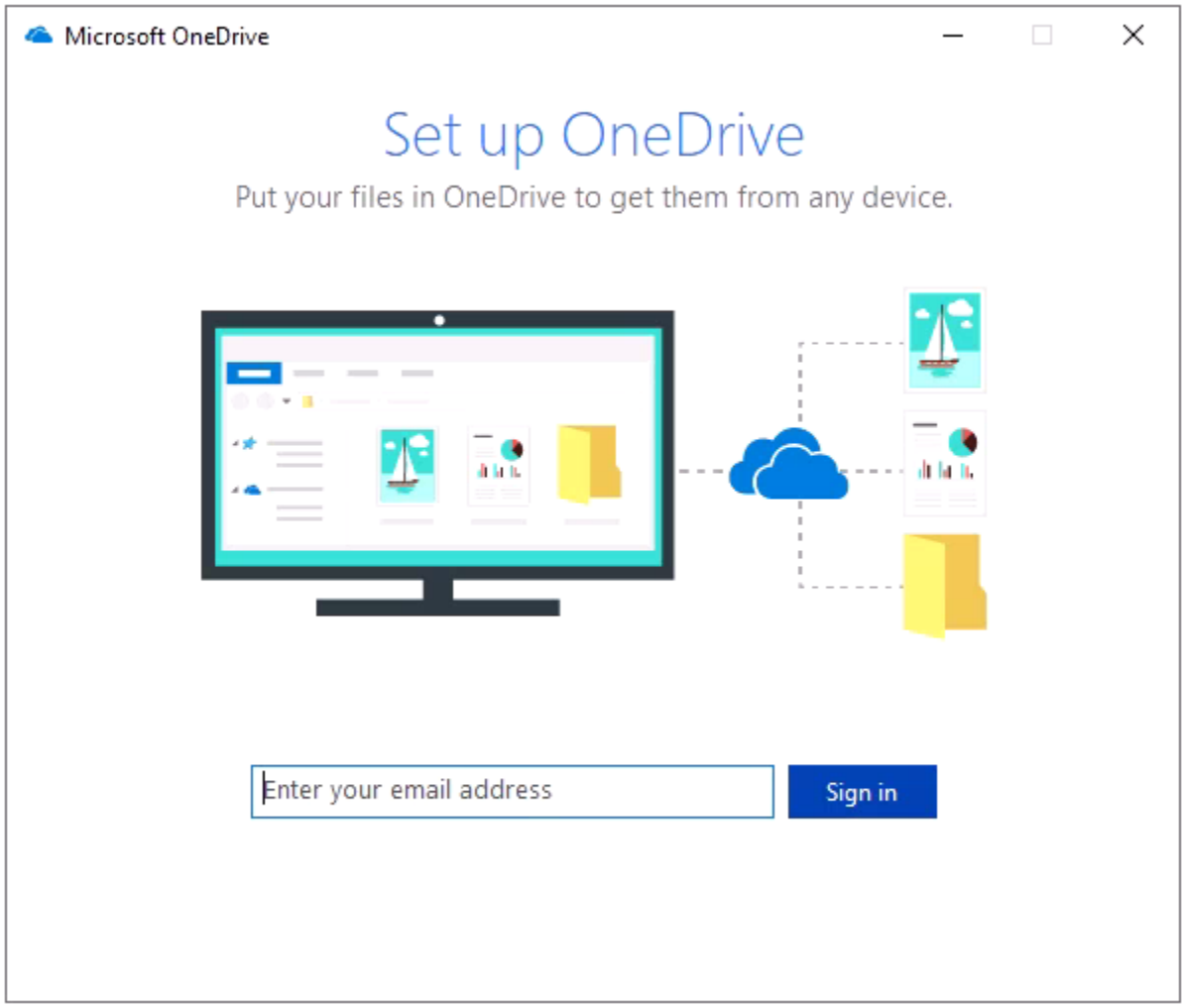
Open File Explorer by clicking on start and typing in "File Explorer" in the search bar and then hit enter



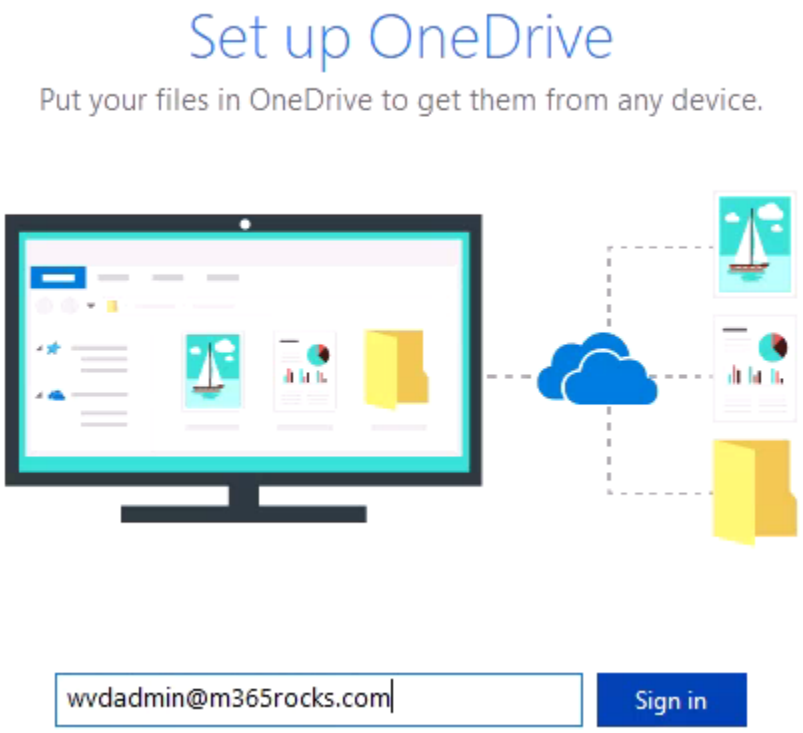
In File Explorer the OneDrive Sync Client will show up under the Quick Access Icon on the left hand side of the window

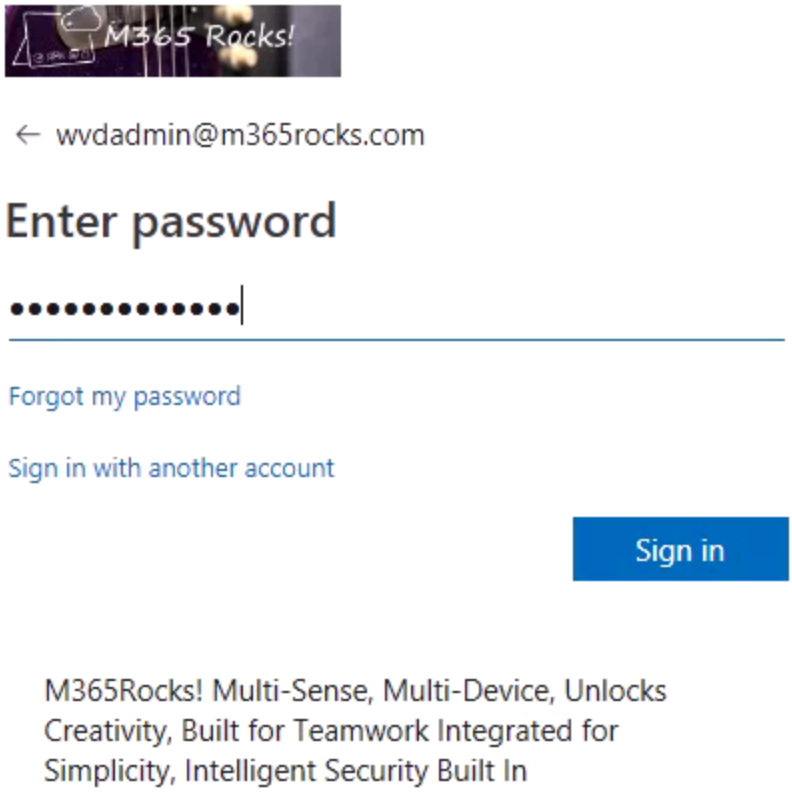


If you did not have any accounts signed into OneDrive, click the OneDrive icon and it will prompt you to sign in

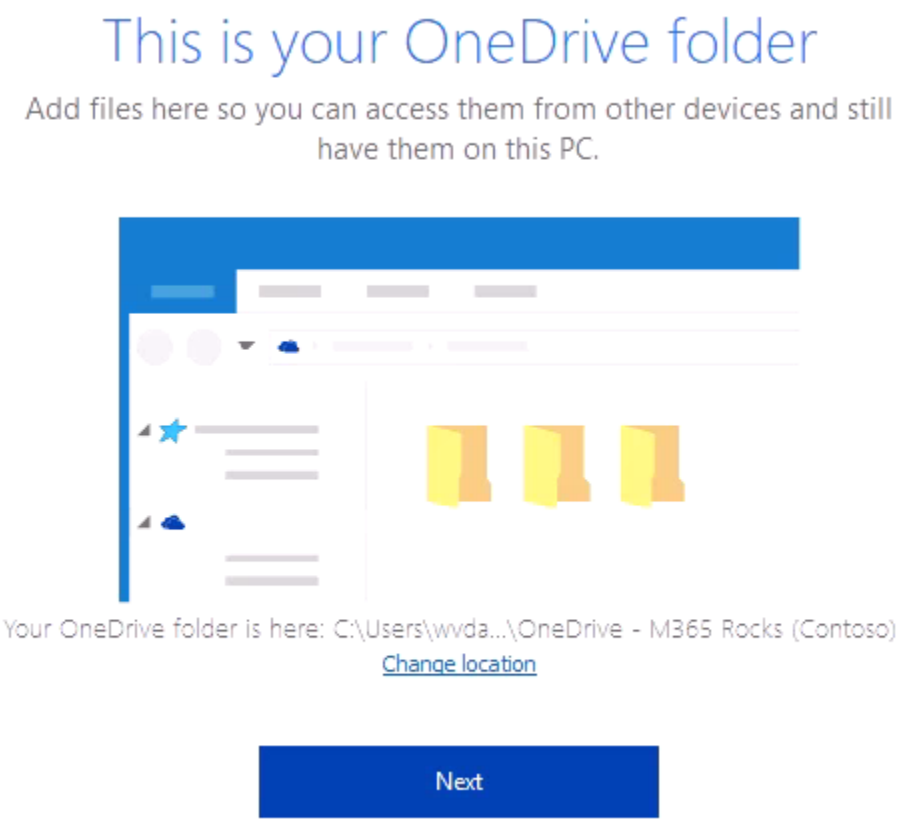


Enter your username and password and click sign in

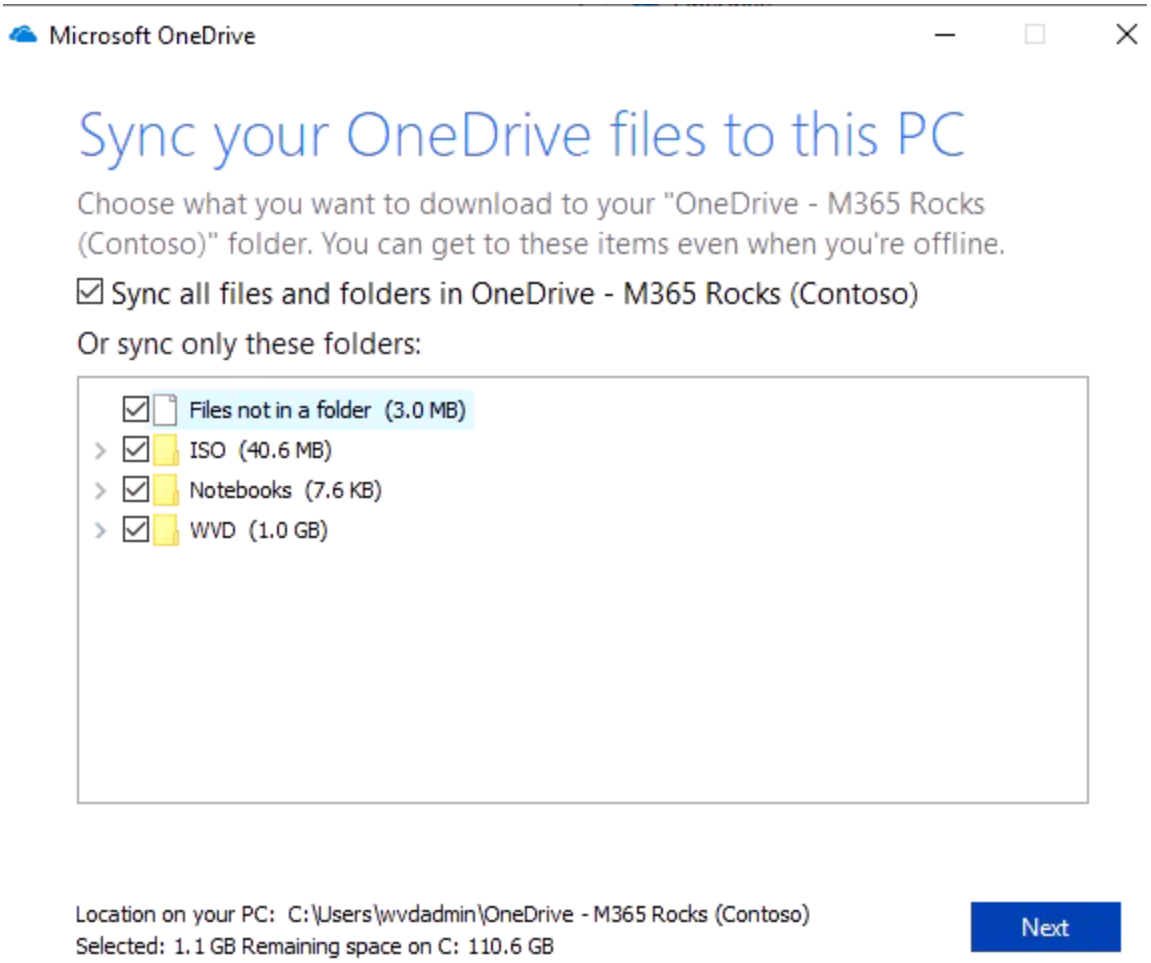




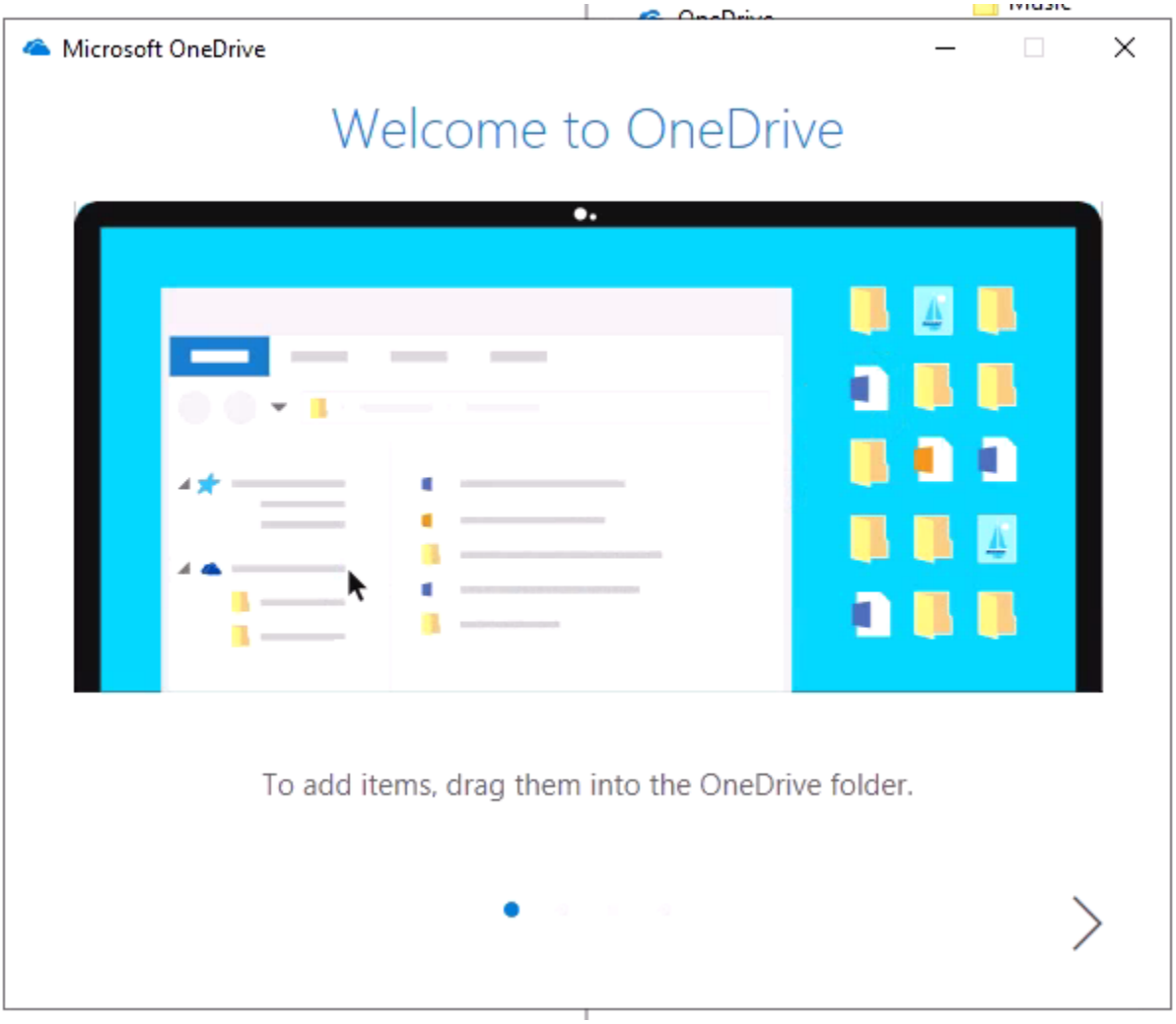
If all goes well and setup completes you'll see this



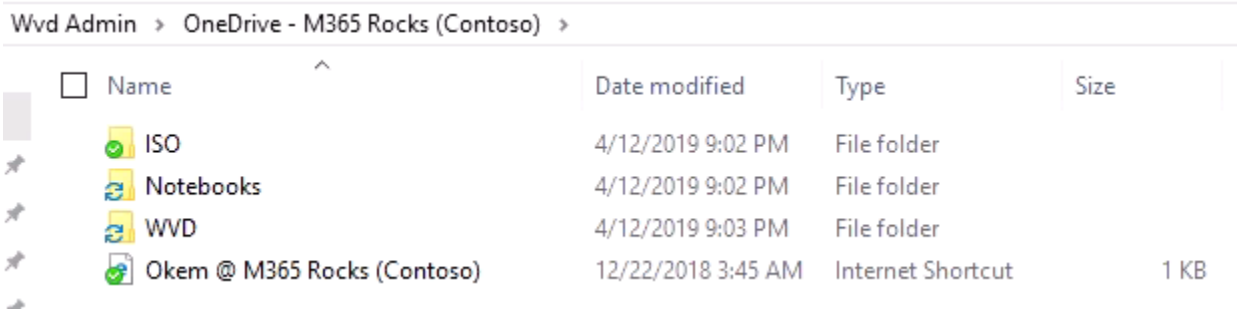
Click Next and choose the files you want to Sync and click next

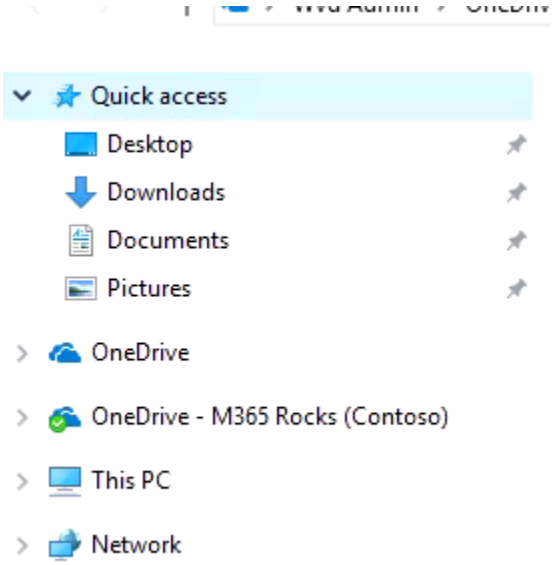


Close the "Welcome to One Drive" Dialog Box



You should see your files start syncing





You have now finished setting up OneDrive per Machine mode on your Windows 10 Multi Session VM

Install Office Pro Plus

Login in your Host pool VM.

Go to : https://docs.microsoft.com/en-us/azure/virtual-desktop/set-up-customize-master-image

Follow the steps in following chapters:

- "Install Office in shared computer activation mode"
- Disable Automatic Updates
- Configure session timeout policies
- Set up time zone redirection
- Disable Storage Sense

You are now ready to demonstrate the benefits of Microsoft 365 in Virtual environment.

Information: If you plan to create several VM in your host pool or you want to recreate them, our recommendation is to create a VHD template. The webpage that you previously followed also describe exactly how to create an Azure VHD template that can then be used during Host Pool deployment

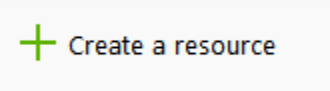
Automatically turning on and turning off your VM's Azure

So now that you've set up your tenants with its different component, we need a way to make sure we turn off the VM's daily and then turn them on so we don't go over the $150 we are allotted monthly. To turn on and turn off your VM's you need to create an agent that does this for you. I chose to go an Azure solution called "Start/Stop VM's". Like its name implies it allows you to do the following

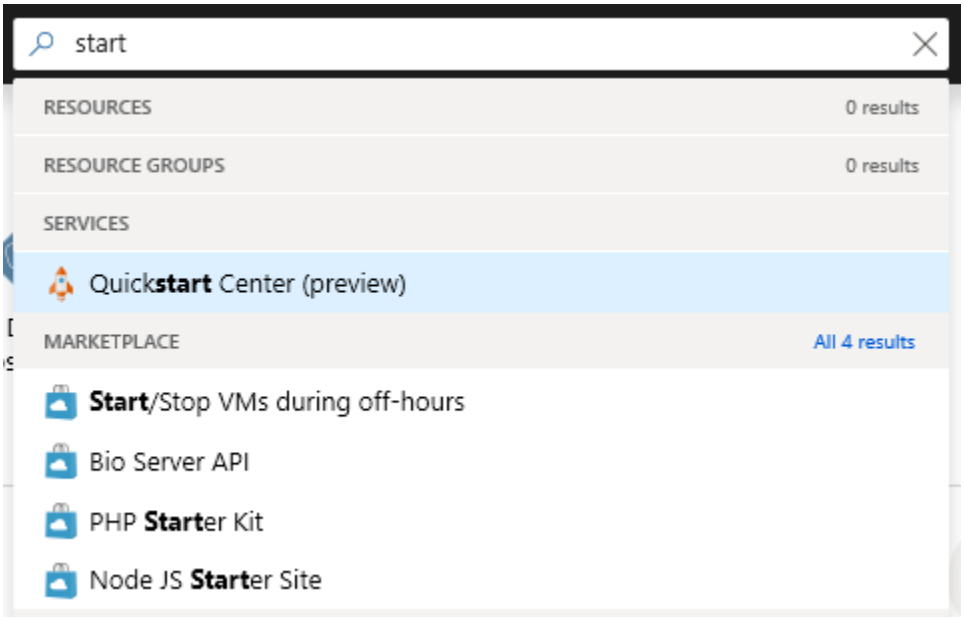
- Schedule VMs to start and stop.
- Schedule VMs to start and stop in ascending order by using Azure Tags (not supported for classic VMs).
- Auto stop VMs based on low CPU usage.

You need an Automation Account and a Log Analytics workspace to make this work. I walk you through how to create both these things from scratch. If you want to read more about the "Start/Stop VM" solution you can read it here

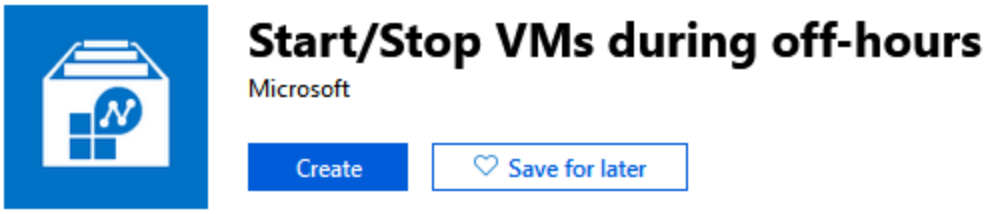
1. Log into your Visual Studio Azure Subscription using portal.azure.com
2. On the left-hand side of the screen click on "Create a resource"



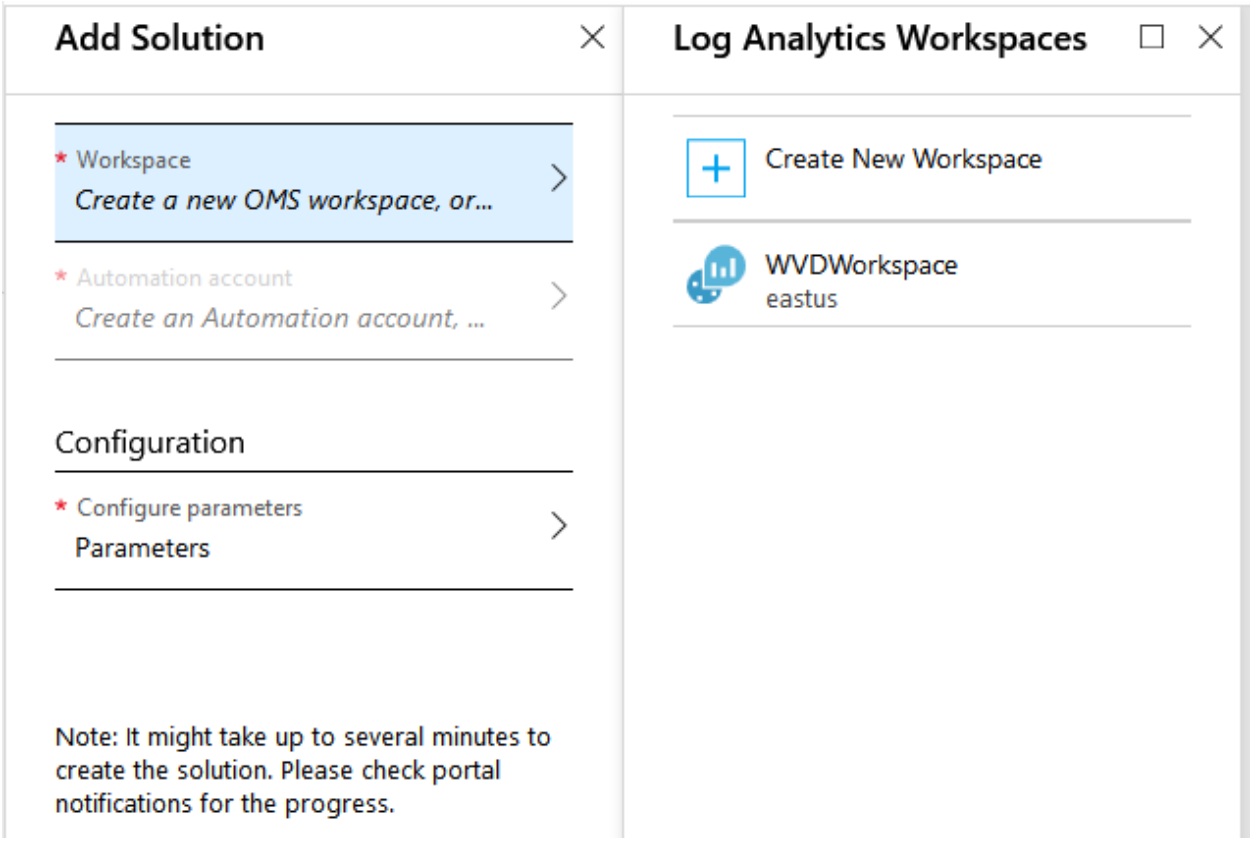
3. Type "Start/Stop" in the search bar



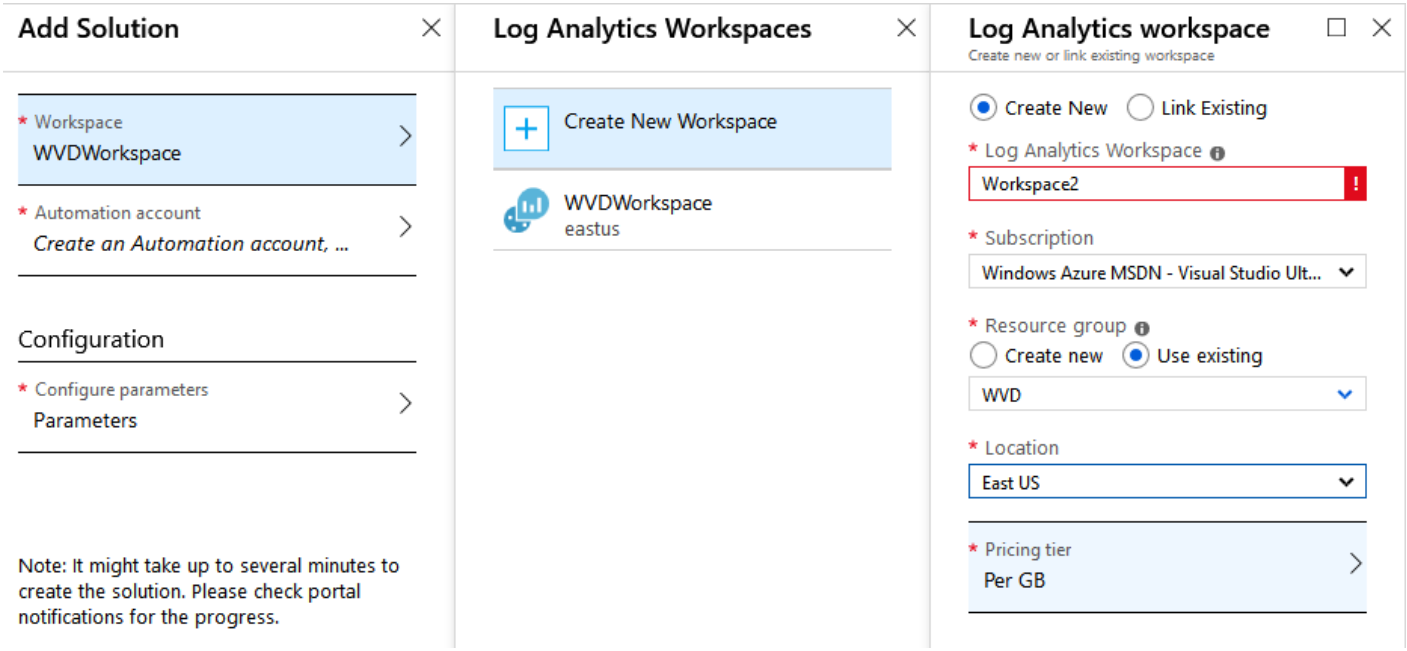
4. Click on Start/Stop VMs during off-hours
5. Click on Create



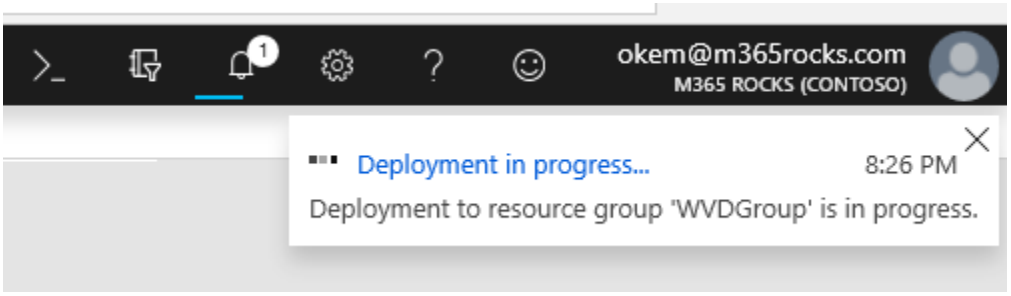
6. Under "Add Solution" click on the "Workspace" blade



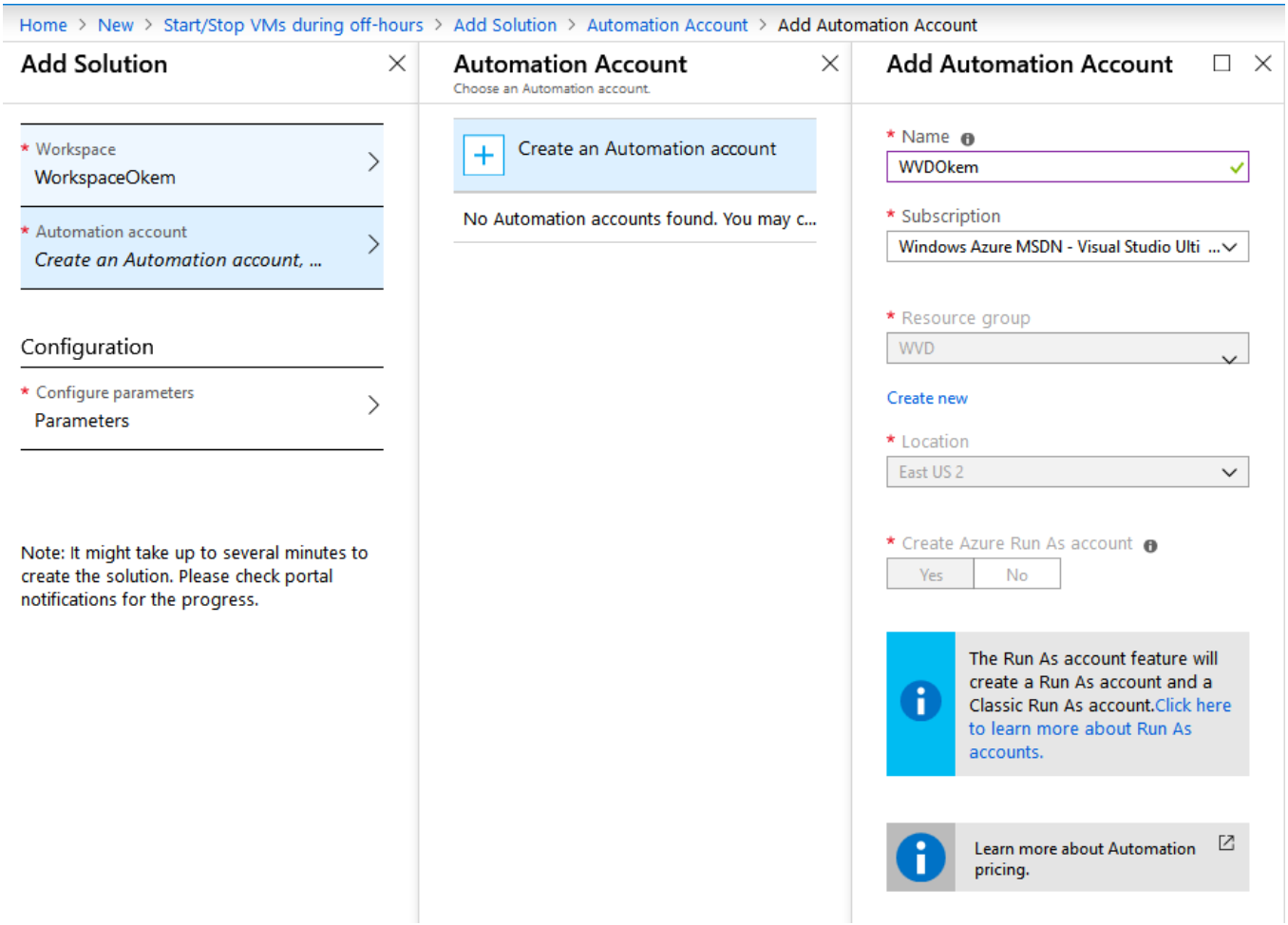
7. You can create a new workspace or use an old one. I chose to create a new workspace, so I clicked "Create New Workspace"
8. Choose a workspace which is located in the same region as the WVD tenant
9. Choose a unique name



10. Make sure your Visual Studio subscription is selected under "subscription"
11. Use the existing resource group where the WVD tenant is located
12. Put it in the same location as the WVD tenant
13. Leave the default pricing tier
14. Click Ok

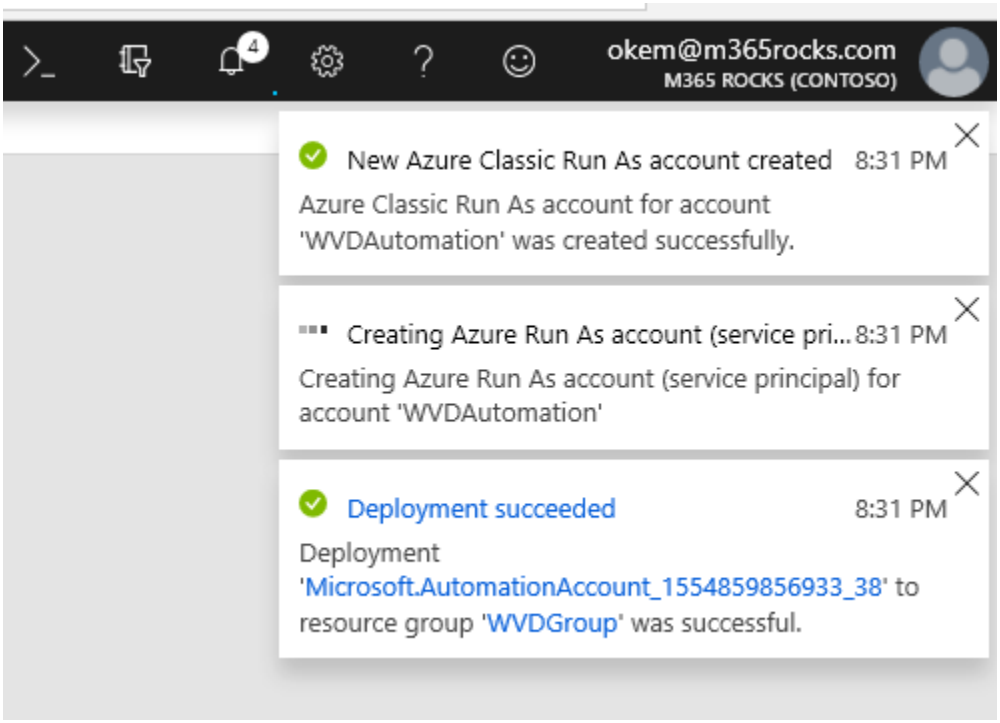


15. Under "Add Solution" click on "Automation Account"
16. You can choose an Automation Account you previously had but I recommend creating a new one. My instructions will show you how to create a new one

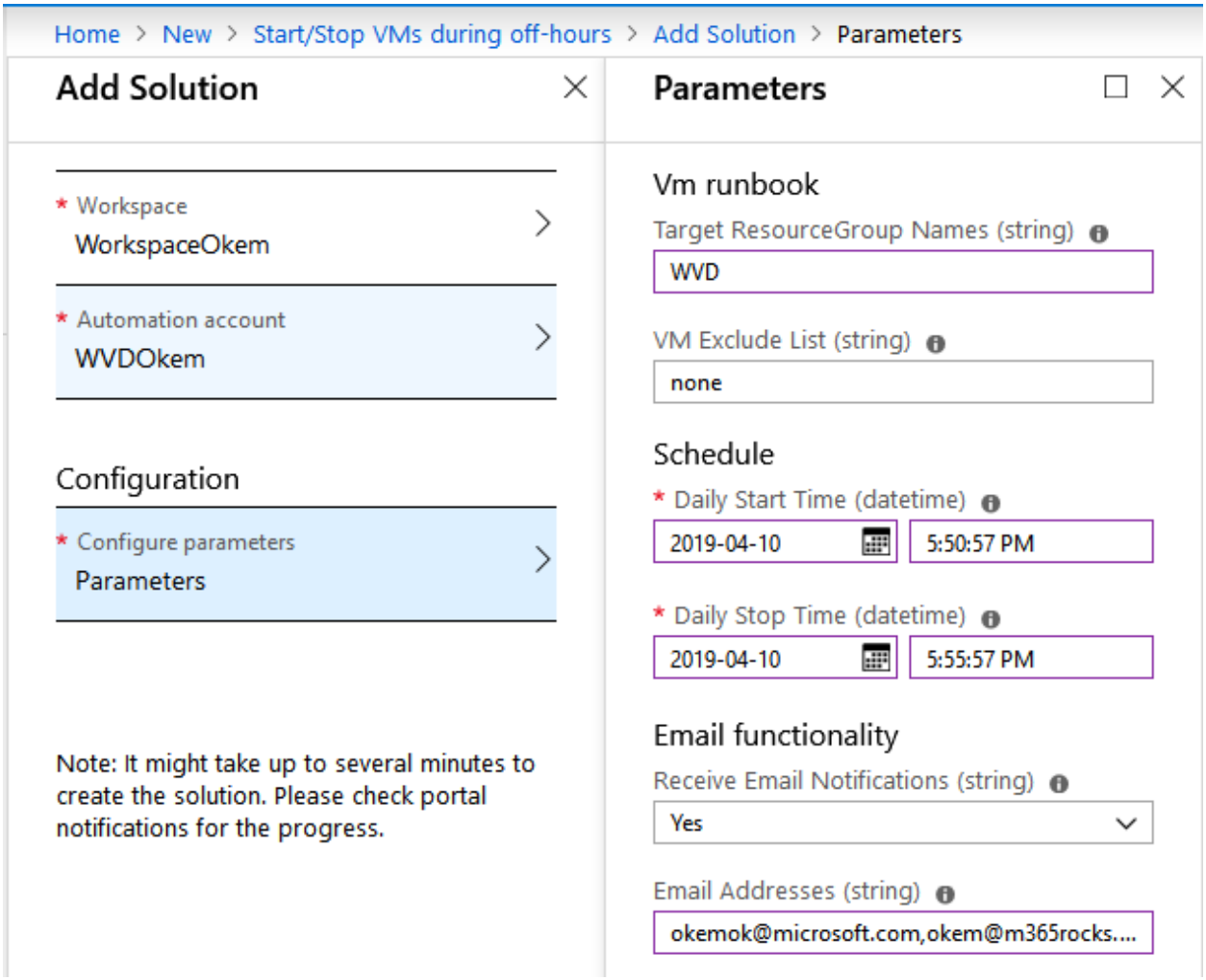


17. Click on "Create an Automation Account"
18. Select your Visual Studio Subscription
19. On the "Add Automation Account" blade enter a unique name, choose the same RG as the WVD tenant
20. Leave the "Create Azure Run as Account" in its default setting
21. Click Ok

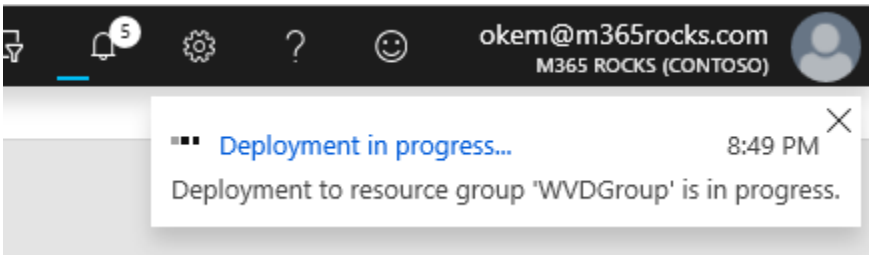
22. You should see your Run-As and your Automation account start deploying and complete



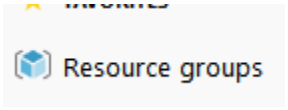
23. Click on the "configure parameters" under the Configuration section
24. Enter the name of the RG where the WVD Tenant is



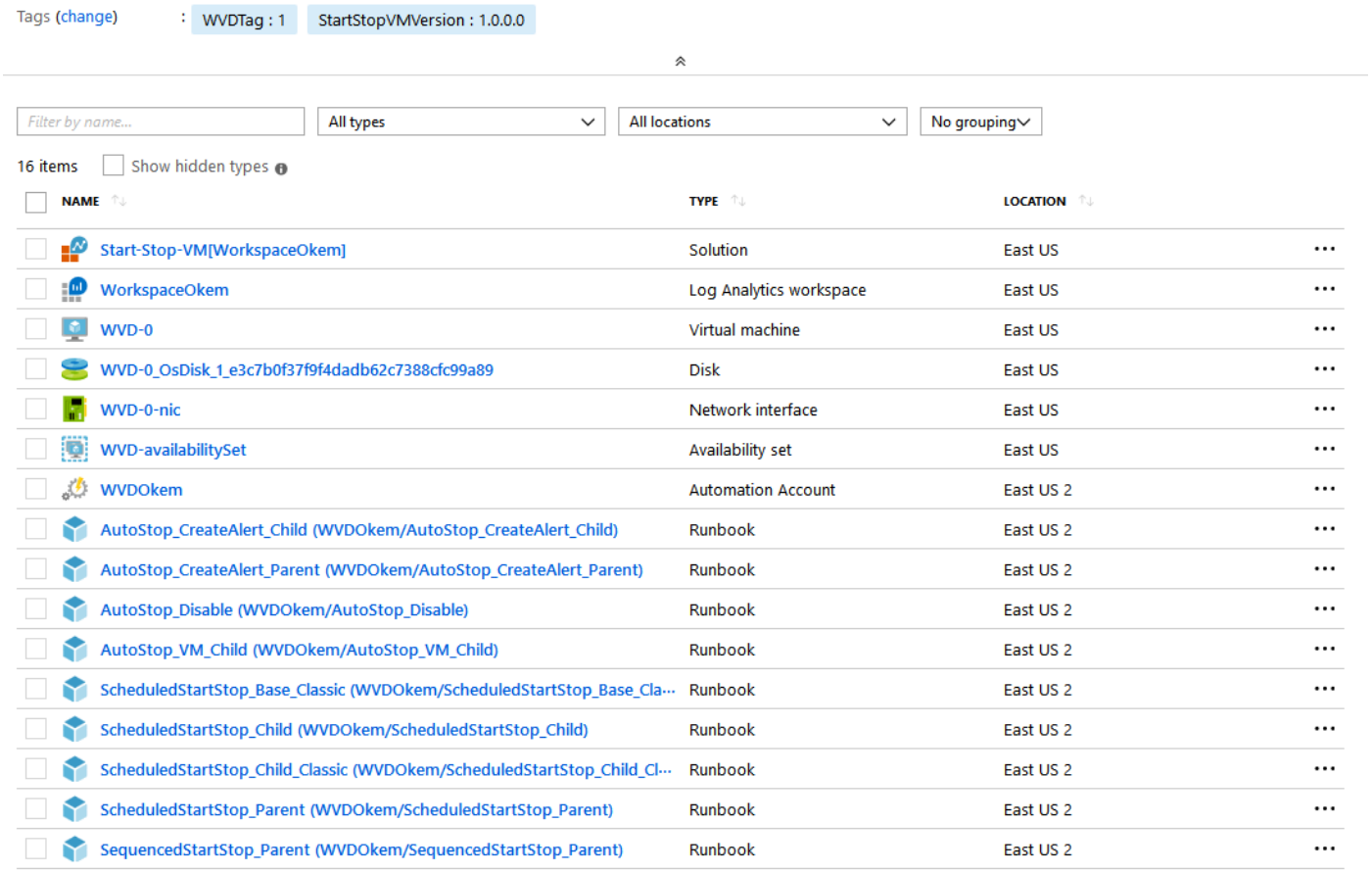
25. Choose a daily start time and daily stop time
26. Choose an email you want to get notifications on
27. Click OK
28. Click Create. You should see your deployment kickoff and complete



29. Navigate back to your resource group by clicking on Resource Group on the left-hand side of the screen.

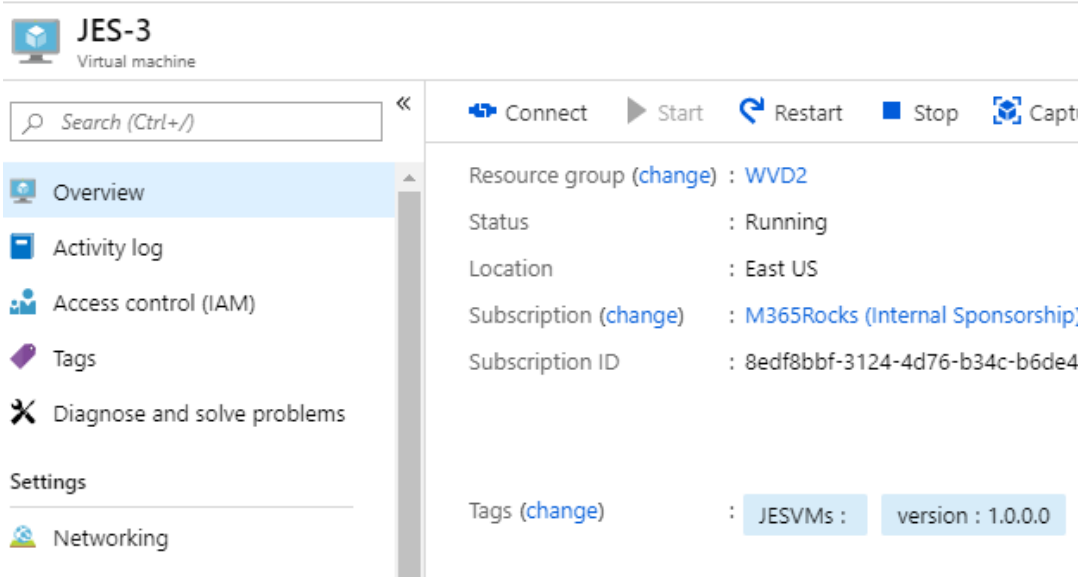


30. The Overview tab should be selected automatically, and you should see the contents of your resource group which should now contain the automation account and the runbook.

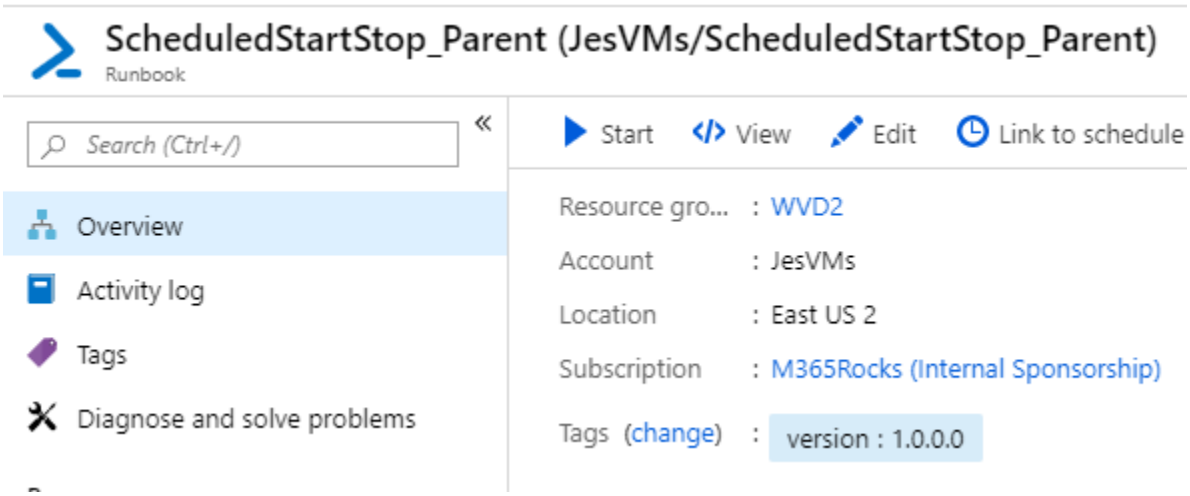


31. You can tag VM's to link them to certain actions. When you create the "Start/Stop VM" run command a tag automatically gets added to the VM's in the resource group and it is linked to the "Start/Stop VM" command.

VM in the Resource group the command will apply to showing the "version 1.0.0.0" tag automatically created



Runbook [part of the Start/Stop VM command] showing the "version 1.0.0.0" tag



If you have any other VM's you can click on them and go through the same process of adding a tag to the VM. When you complete that you are set to have your machines stop and start on the schedule you've created

Build Cheat Sheet

	Field	Value	Notes
Blade 1: Basics	Hostpool name		Example: FullDesktop
	Desktop Type		Recommended: Pooled
	Default desktop users		Add users synced from your local domain separated by a comma
	Subscription		Recommended: Visual Studio Enterprise
	Resource group		
	Location		(US) East US 2
Blade 2: Configure number of VMs based on profile usage	Total users		Recommended Small Number “2”
	Virtual machine size		Recommended: Standard B1ms or B2s
	Virtual machine name prefix		Should be unique to tenant, less than 10 characters and only letters or numbers: “WVD”
Blade 3: configure the VMs for Azure	Image Source		Recommend: Gallery
	Image OS		Windows 10 Enterprise multi-session + Office 365 ProPlus
	Disk type		Recommended: HDD
	AD Domain join UPN		Domain Admin account in the DC created for WVD
	Password		Note a password or hint
	Specify domain or OU		
	Domain to join		If used
	Virtual network		Virtual Network for DC and Host will need to be on the same network.
	Subnets		Subnet should match as well
Blade 4: WVD tenant information	Tenant group name		Important: Should be “Default Tenant Group”
	Tenant name		Use “Get-RdsTenant” to confirm
	Tenant admin authentication type		
	UPN		Azure AD Admin
	Password		Note a password or hint

Setting Deployment Context

	Field	Value	Notes
	$brokerurl	"https://rdbroker.wvd.microsoft.com"	
Azure Active Directory Directory ID	$aadTenantId		https://www.whatismytenantid.com/
Azure Subscription ID	$azureSubscriptionId		Azure Portal search for "Subscriptions."
Windows Virtual Desktop Tenant Name			This is set during the PowerShell script and is used when adding new users

Create a new PowerShell script, modifying the **bold** variables to reflect your tenant ID and subscription ID, and execute the following commands. When prompted, sign in using the admin account that was assigned to the TenantCreator role.

```
#Install PowerShell modules

Install-Module -Name Microsoft.RDInfra.RDPowerShell

Import-Module -Name Microsoft.RDInfra.RDPowerShell


# Setting Deployment context

$brokerurl = "https://rdbroker.wvd.microsoft.com"

$aadTenantId = < Azure Active Directory Directory ID >

$azureSubscriptionId = < Azure Subscription ID >

Add-RdsAccount -DeploymentUrl $brokerurl
```

Next, use the following command to create the Windows Virtual Desktop tenant.

```
New-RdsTenant -Name < Windows Virtual Desktop Tenant Name > -AadTenantId $aadTenantId -AzureSubscr
iptionId $azureSubscriptionId
```

Troubleshooting

If you have any other VM's you can click on them and go through the same process of adding a tag to the VM. When you complete that you are set to have your machines stop and start on the schedule you've created

1. Problem: When using a live.com account you get a message that does not exist in tenant 'MS Azure Cloud' and cannot access the application (Windows Virtual Desktop Client) in that tenant. The account needs to be added as an external user in the tenant first. Sign out and sign in again with a different Azure Active Directory user account.

   Answer: Create a second AAD Global Admin with the FQDN for the tenant that can also be used the Tenant Creator.

2. First login to the Windows Virtual Desktop you may see the message "Oops, we couldn't connect to "FullDesktop" We couldn't connect to the gateway because of an error. If this keeps happening, ask your admin or tech support for help."

   Normally this is just a timing issue. Even if the resources VM is available it may take another few minutes for the access to completely provision. Wait another 15 to 30 minutes before making any changes.

3. Error assigning a new user.

   Make sure the new user has been synced from the DC to AAD before adding to WVD host pool.

4. If you need console access to your host machine.

   Connect from your Azure DC using RDP to the Host. (Or if all else fails add a public IP to your Host machine for troubleshooting.)

5. If you get locked out of the WVD Host. Create/Reset a local admin user via the Azure portal for the Virtual Machine.

   Home → Virtual machines → WVD Host → Support + troubleshooting → Reset password

Manage with Microsoft Azure App

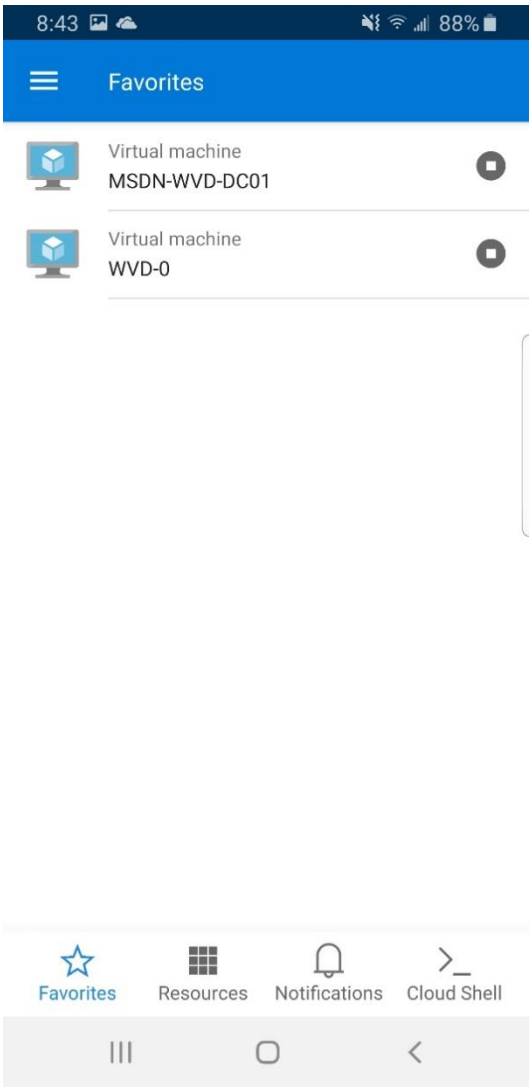
Showing our Customers Windows Virtual Desktop as a live demo can be made easier by using the Microsoft Azure app. This app allows you to turn on or off the VM's created in this document right from your phone.

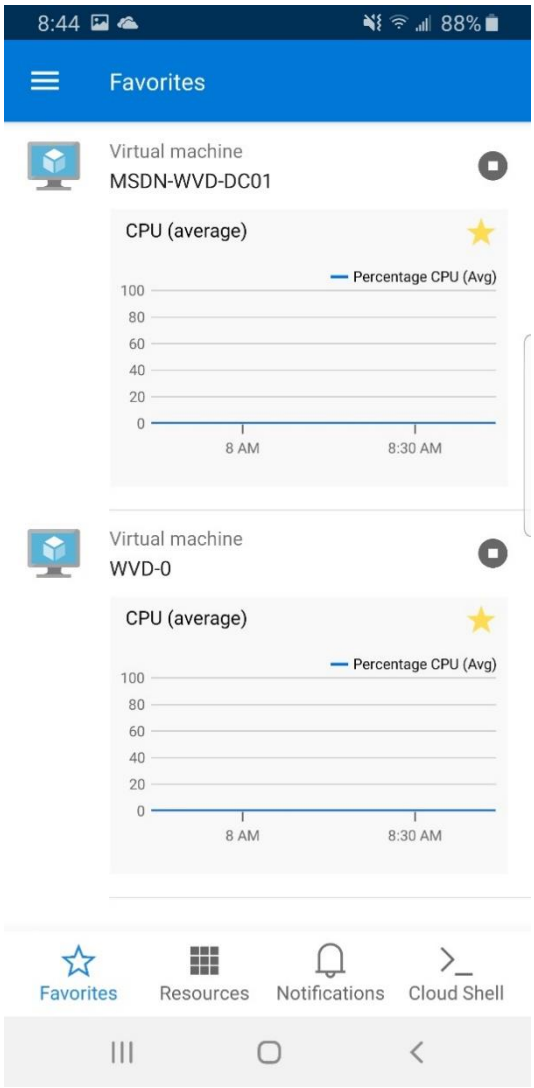
1. Download the App from the Play Store or Apple Store **Microsoft Azure**

2. Sign into the **Microsoft Azure** app with your Visual Studio admin account.

3. Under **Resources** select the **DC** and **WVD** desktop and mark as **Favorites**.

4. Finally select the **Favorites** tab to see the **VM's**. You can **start** and **stop** them as needed before beginning or at the end of your Windows Virtual Desktop demo. (Also useful for confirming the resources have been shut down and deallocated when not in use.)





(You can also add the CPU (average), Network (total), Disk bytes (total) or Disk Operations/sec (average) as a favorite item as well.)