A picture containing drawing

Description automatically generated

Windows Virtual Desktop (WVD) PoC Guide

Mar 2020

Contents

[Lab 1: WVD Implementation Steps 3](#_Toc36493618)

[Exercise 1: Grant Azure Active Directory permissions to the Windows Virtual Desktop service 3](#_Toc36493619)

[Exercise 2: Assign the Tenant Creator Application role to a user in your Azure Active Directory. 7](#_Toc36493620)

[Exercise 3: Download the WVD PowerShell Module 12](#_Toc36493621)

[Exercise 4: Create the WVD Tenant 13](#_Toc36493622)

[Exercise 5: Create service principals and role assignments by using PowerShell 14](#_Toc36493623)

[Prerequisites 15](#_Toc36493624)

[Task 1: Create a service principal in Azure Active Directory 15](#_Toc36493625)

[Task 2: View your credentials in PowerShell 16](#_Toc36493626)

[Task 3: Create a role assignment in Windows Virtual Desktop Preview 16](#_Toc36493627)

[Task 4: Sign in with the service principal 17](#_Toc36493628)

[Exercise 6: Create a host pool by using the Azure ARM Template 18](#_Toc36493629)

[Prerequisites 18](#_Toc36493630)

[Task 1: Deploying the ARM Template to create the Windows Virtual Desktop Host Pool. 18](#_Toc36493631)

[Task 2 (Optional): Assign additional users to the desktop application group 23](#_Toc36493632)

[Task 3**:** Validate the new Host Pools 24](#_Toc36493633)

[Exercise 7: Manage App Groups 25](#_Toc36493634)

[Exercise 8: End User Experience 29](#_Toc36493635)

[Lab 2: WVD Management 35](#_Toc36493636)

[Exercise 1: Configure the Load balancing Method 35](#_Toc36493637)

[Task 1: Breadth First 35](#_Toc36493638)

[Task 2: Depth First 36](#_Toc36493639)

[Task 3: Persistent Desktops 37](#_Toc36493640)

[Exercise 2: Set Friendly Names for Published Desktops 38](#_Toc36493641)

[Exercise 3: Deploy the Management UI 39](#_Toc36493642)

[Appendix 46](#_Toc36493643)

[Exercise 1: Create and configure an Azure Active Directory Domain Services instance 46](#_Toc36493644)

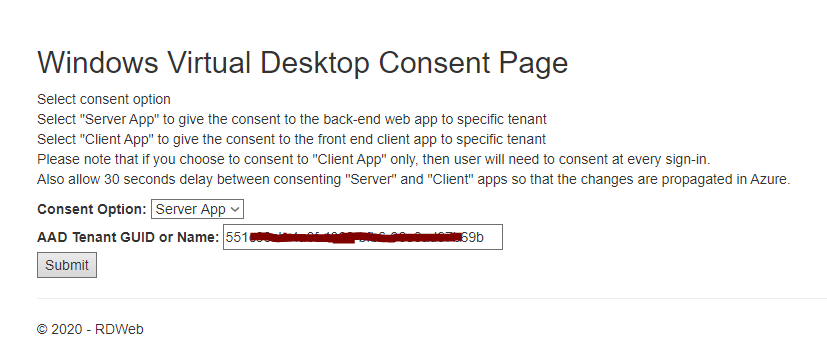
# Lab 1: WVD Implementation Steps

Creating a WVD (Windows Virtual Desktop) tenant is the first step towards building out your desktop virtualization solution. A tenant is a group of one or more host pools. Each host pool consists of multiple session hosts, running as virtual machines in Azure and registered to the Windows Virtual Desktop service. Each host pool also consists of one or more app groups that are used to publish remote desktop and remote application resources to users. With a tenant, you can build out host pools, create app groups, assign users, and make connections through the service.

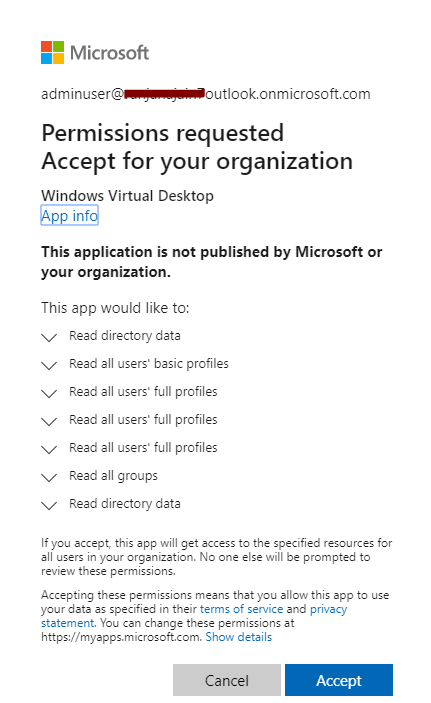
The subsequent sections will detail the step-step process to implement a working WVD solution in Azure.

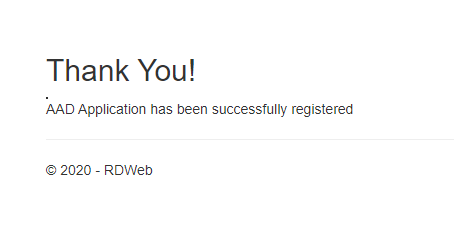
## Exercise 1: Grant Azure Active Directory permissions to the Windows Virtual Desktop service

1. Open a browser and connect to the [Windows Virtual Desktop consent page.](https://rdweb.wvd.microsoft.com/)
2. For **Consent Option** > **Server App**, enter the Azure Active Directory tenant name or Directory ID (from the Azure portal), then select **Submit**.
   * For Cloud Solution Provider customers, the ID is the customer's Microsoft ID from the Partner Portal.
   * For Enterprise customers, the ID is located under **Azure Active Directory** > **Properties** > **Directory ID**.

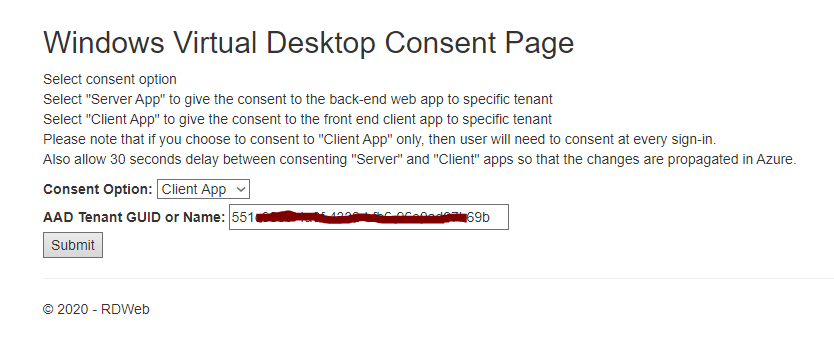


1. Sign in to the [Windows Virtual Desktop consent page.](https://rdweb.wvd.microsoft.com/) with a global administrator account. For example, if you were with the Contoso organization, your account might be admin@contoso.com or [admin@contoso.onmicrosoft.com](mailto:admin@contoso.onmicrosoft.com).

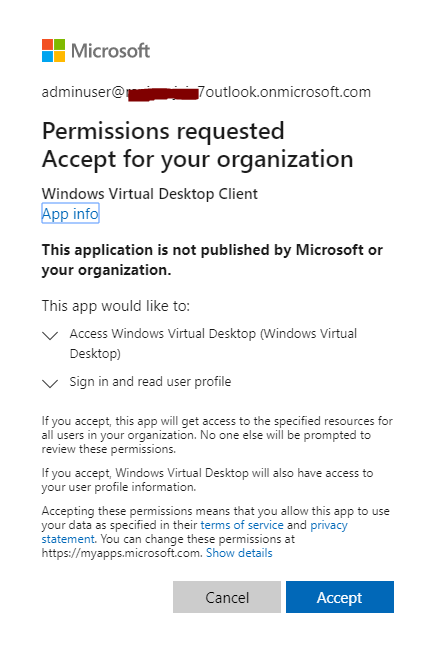


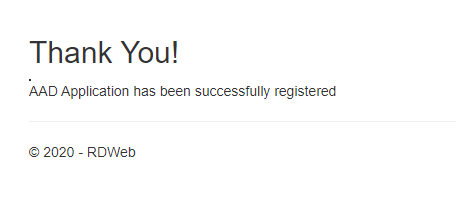


1. Select **Accept** > wait for one minute.
2. Navigate back to the [Windows Virtual Desktop consent page.](https://rdweb.wvd.microsoft.com/)
3. Go to **Consent Option** > **Client App**, enter the same Azure AD tenant name or Directory ID, then select **Submit**.



1. Sign into the Windows Virtual Desktop consent page as global administrator like you did back in step 3. Select **Accept**.



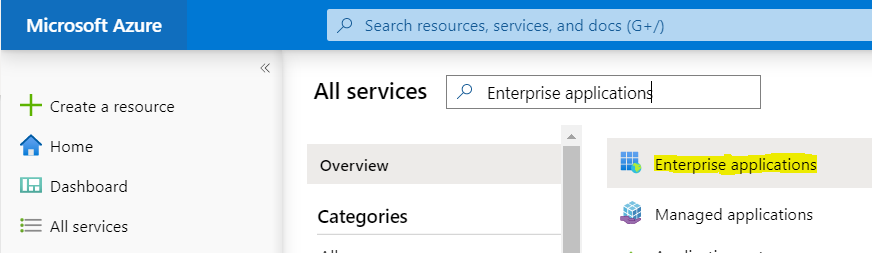


## Exercise 2: Assign the Tenant Creator Application role to a user in your Azure Active Directory.

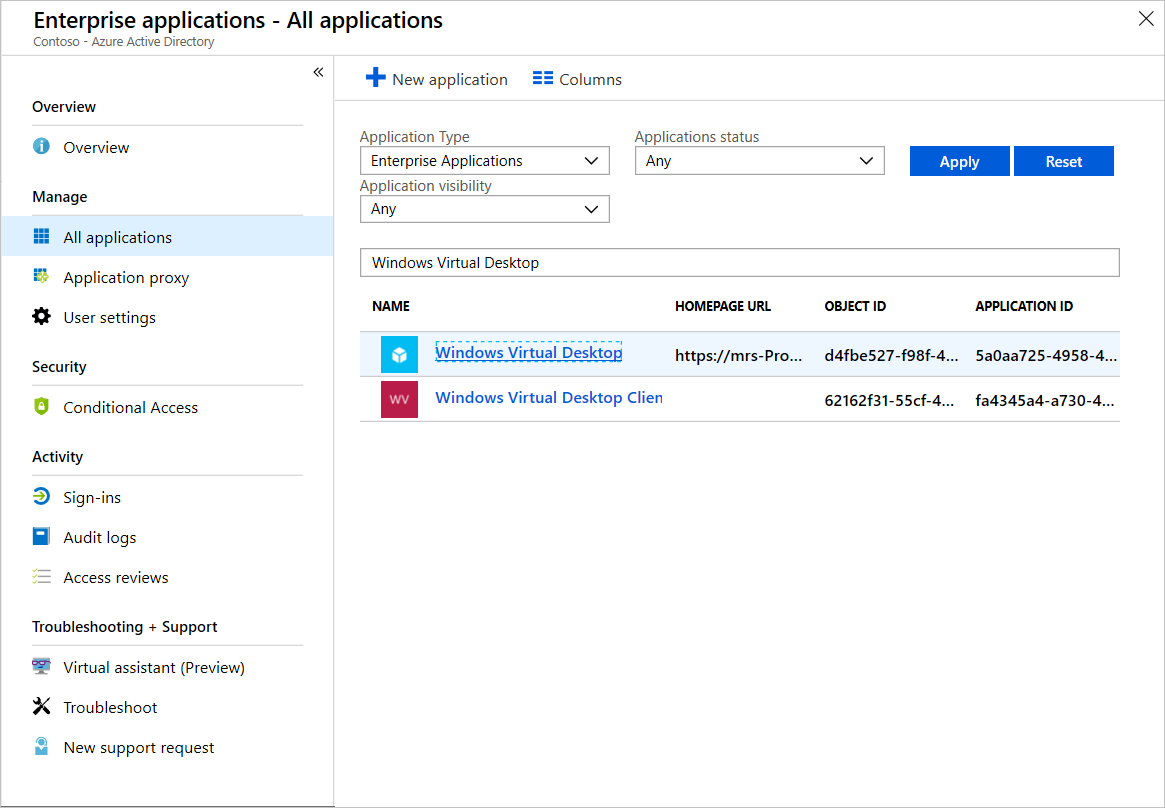
Assigning an Azure Active Directory user the TenantCreator application role allows that user to create a Windows Virtual Desktop tenant associated with the Azure Active Directory instance. You'll need to use your global administrator account to assign the TenantCreator role.

To assign the TenantCreator application role:

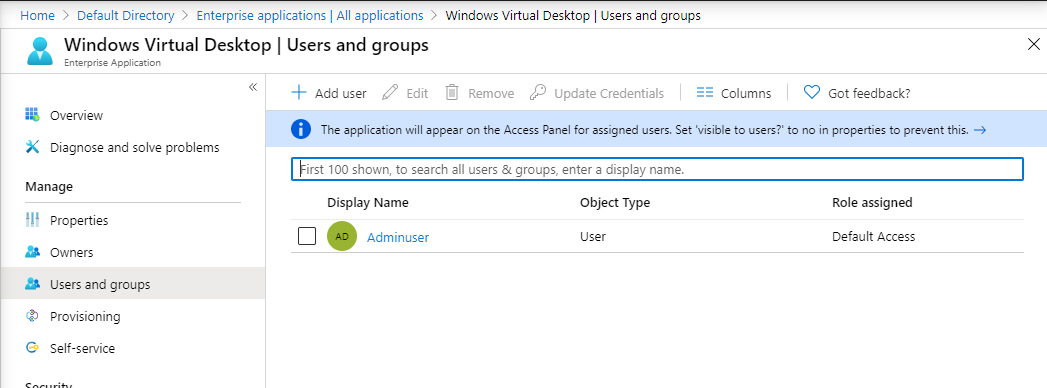
1. Go to the [Azure portal](https://portal.azure.com/) to manage the TenantCreator application role. Search for and select **Enterprise applications**.



1. Within **Enterprise applications**, search for **Windows Virtual Desktop**. You'll see the two applications that you provided consent for in the previous section. Of these two apps, select **Windows Virtual Desktop**.



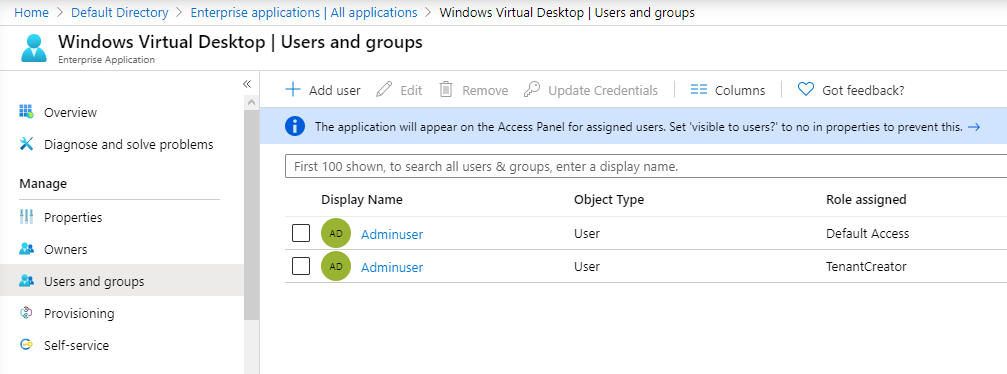
1. Select **Users and groups**. You might see that the administrator who granted consent to the application is already listed with the **Default Access** role assigned. This is not enough to create a Windows Virtual Desktop tenant. Continue following these instructions to add the **TenantCreator** role to a user.



1. Select **Add user**, and then select **Users and groups** in the **Add Assignment** tab.
2. Search for a user account that will create your Windows Virtual Desktop tenant. For simplicity, this can be the global administrator account.



1. Select the user account, choose the **Select** button, and then select **Assign**.
2. On the **Windows Virtual Desktop - Users and groups** page, verify that you see a new entry with the **TenantCreator** role assigned to the user who will create the Windows Virtual Desktop tenant.

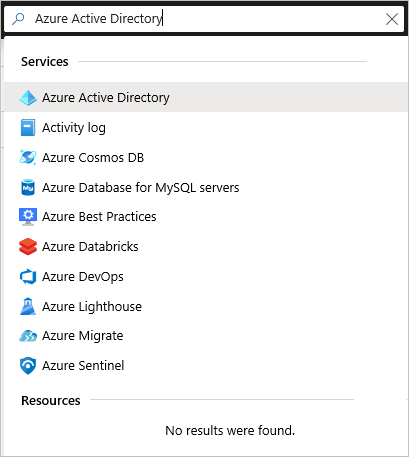


Before you continue on to create your Windows Virtual Desktop tenant, you need two pieces of information:

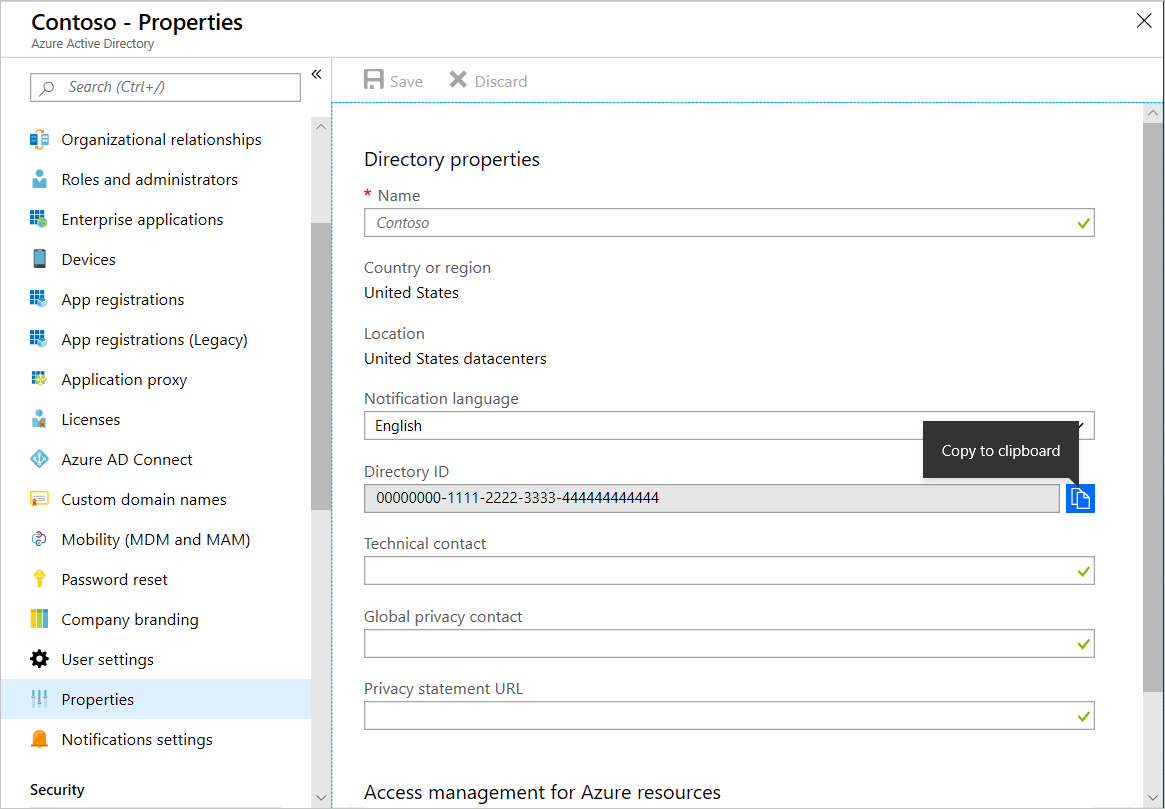
* Your Azure Active Directory tenant ID (or **Directory ID**)
* Your Azure subscription ID

To find your Azure Active Directory tenant ID (or **Directory ID**):

1. In the same [Azure portal](https://portal.azure.com/) session, search for and select **Azure Active Directory**.

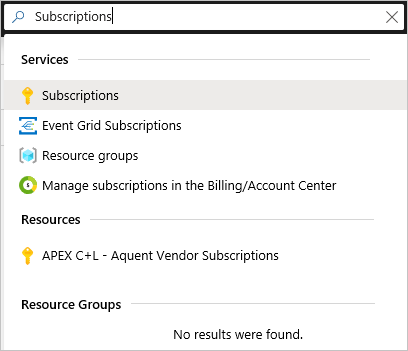


1. Scroll down until you find **Properties**, and then select it.
2. Look for **Directory ID**, and then select the clipboard icon. Paste it in a handy location so you can use it later as the **AadTenantId** value.

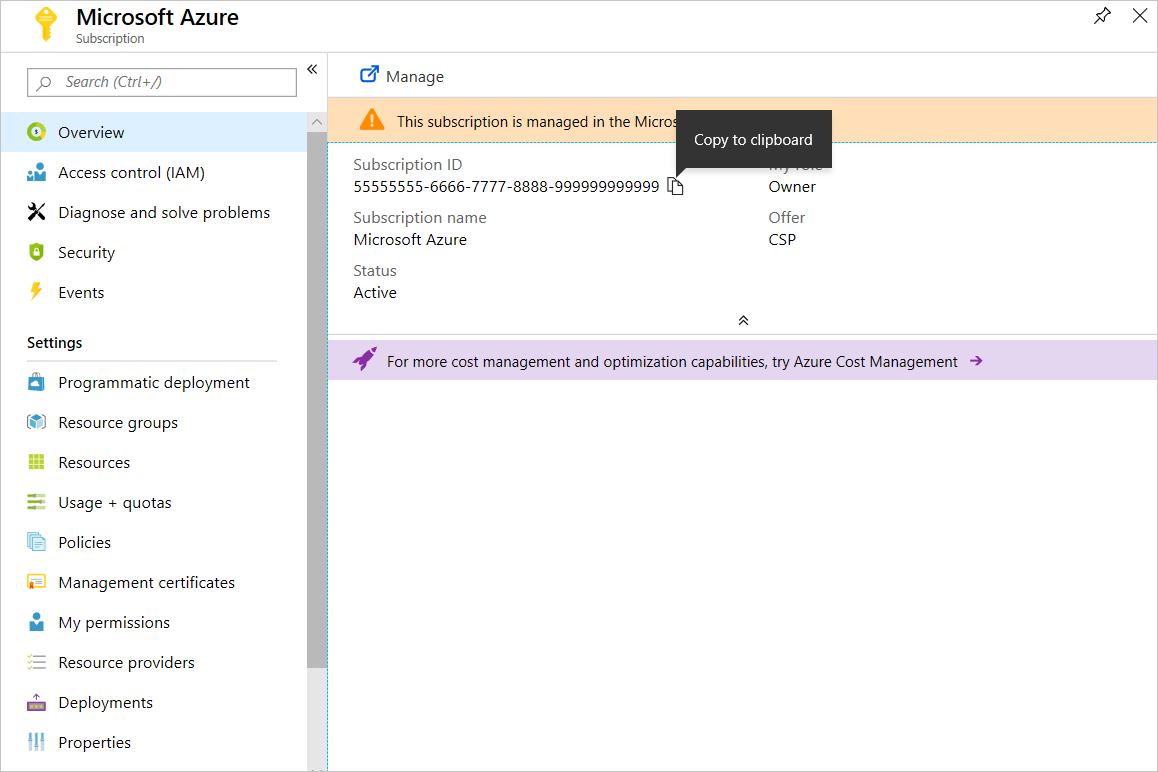


To find your Azure subscription ID:

1. In the same [Azure portal](https://portal.azure.com/) session, search for and select **Subscriptions**.

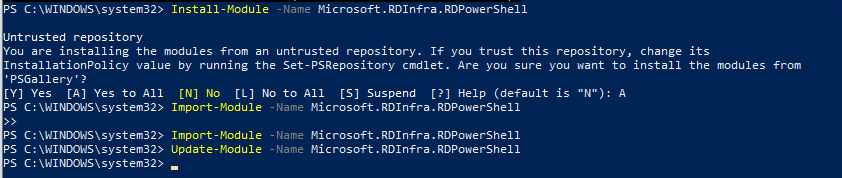


1. Select the Azure subscription you want to use to receive Windows Virtual Desktop service notifications.
2. Look for **Subscription ID**, and then hover over the value until a clipboard icon appears. Select the clipboard icon and paste it in a handy location so you can use it later as the **AzureSubscriptionId** value.



## Exercise 3: Download the WVD PowerShell Module

1. Install the [Windows Virtual Desktop module](https://docs.microsoft.com/en-us/powershell/windows-virtual-desktop/overview) on your computer.

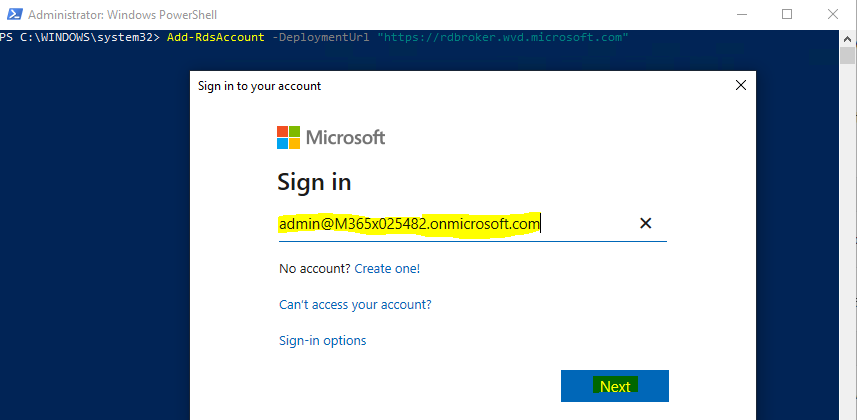


1. You can now run Windows Virtual Desktop cmdlets in your PowerShell window. If you close your PowerShell session, you'll have to import the module into your session again.

## Exercise 4: Create the WVD Tenant

1. Sign in to Windows Virtual Desktop by using the TenantCreator user account with this cmdlet:

Add-RdsAccount -DeploymentUrl <https://rdbroker.wvd.microsoft.com>

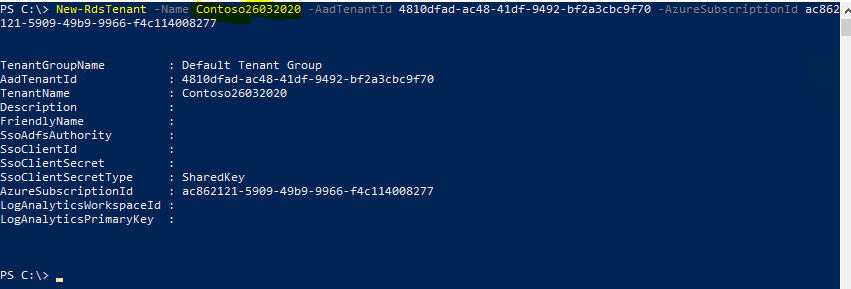


1. After that, create a new Windows Virtual Desktop tenant associated with the Azure Active Directory tenant:

New-RdsTenant -Name <TenantName> -AadTenantId <DirectoryID> -AzureSubscriptionId <SubscriptionID>

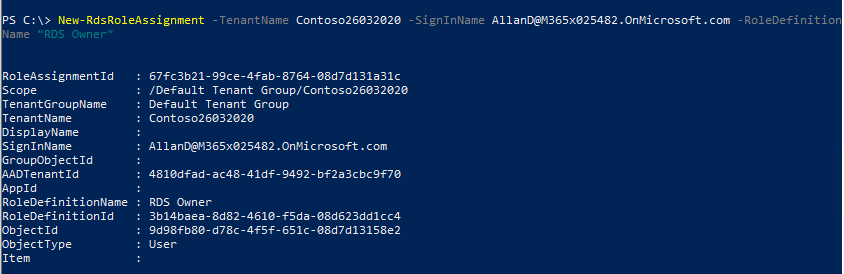
Example – Subsitute XXXXXX as it needs to be unique

New-RdsTenant -Name ContosoXXXXXX -AadTenantId 00000000-1111-2222-3333-444444444444 -AzureSubscriptionId 55555555-6666-7777-8888-999999999999



It's a good idea to assign administrative access to a second user in case you ever find yourself locked out of your account, or you go on vacation

New-RdsRoleAssignment -TenantName <TenantName> -SignInName <Upn> -RoleDefinitionName "RDS Owner"



## Exercise 5: Create service principals and role assignments by using PowerShell

Service principals are identities that you can create in Azure Active Directory to assign roles and permissions for a specific purpose. In Windows Virtual Desktop, you can create a service principal to:

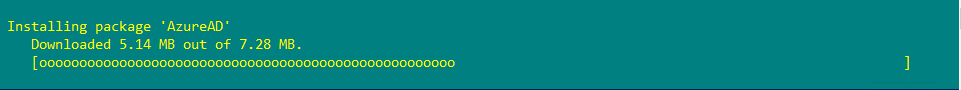
* Automate specific Windows Virtual Desktop management tasks.
* Use as credentials in place of MFA-required users when running any Azure Resource Manager template for Windows Virtual Desktop.

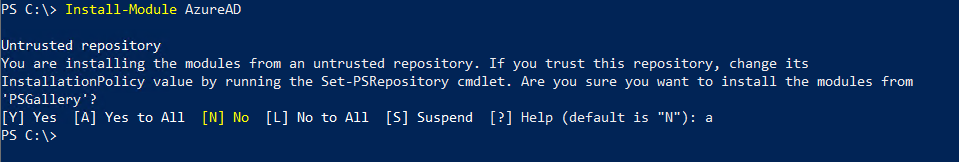
### Prerequisites

Before you can create service principals and role assignments, you need to do three things:

1. Install the AzureAD module. To install the module, run PowerShell as an administrator and run the following cmdlet:

Install-Module AzureAD





### Task 1: Create a service principal in Azure Active Directory

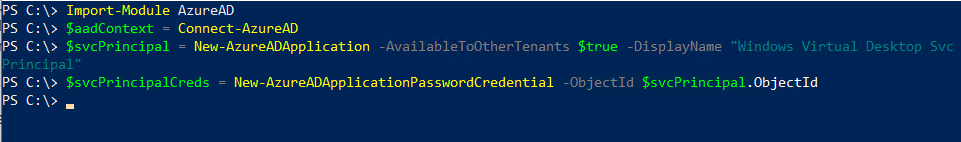
After you've fulfilled the prerequisites in your PowerShell session, run the following PowerShell cmdlets to create a multitenant service principal in Azure.

Import-Module AzureAD

$aadContext = Connect-AzureAD

$svcPrincipal = New-AzureADApplication -AvailableToOtherTenants $true -DisplayName "Windows Virtual Desktop Svc Principal"

$svcPrincipalCreds = New-AzureADApplicationPasswordCredential -ObjectId $svcPrincipal.ObjectId



### Task 2: View your credentials in PowerShell

Before you create the role assignment for your service principal, view your credentials and write them down for future reference. The password is especially important because you won't be able to retrieve it after you close this PowerShell session.

Here are the three credentials you should write down and the cmdlets you need to run to get them:

* Password:

$svcPrincipalCreds.Value

* Tenant ID:

$aadContext.TenantId.Guid

* Application ID:

$svcPrincipal.AppId



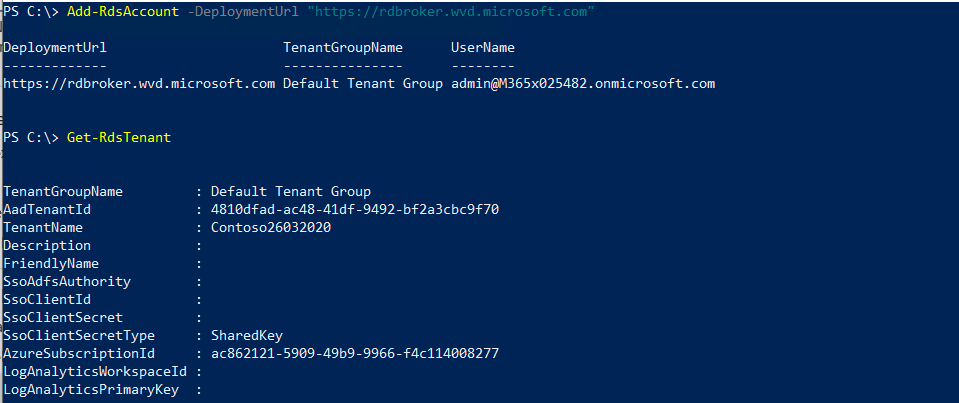
### Task 3: Create a role assignment in Windows Virtual Desktop Preview

Next, you need to create a role assignment so the service principal can sign in to Windows Virtual Desktop. Make sure to sign in with an account that has permissions to create role assignments.

Run the following PowerShell cmdlets to connect to Windows Virtual Desktop and display your tenants.

Add-RdsAccount -DeploymentUrl "https://rdbroker.wvd.microsoft.com"

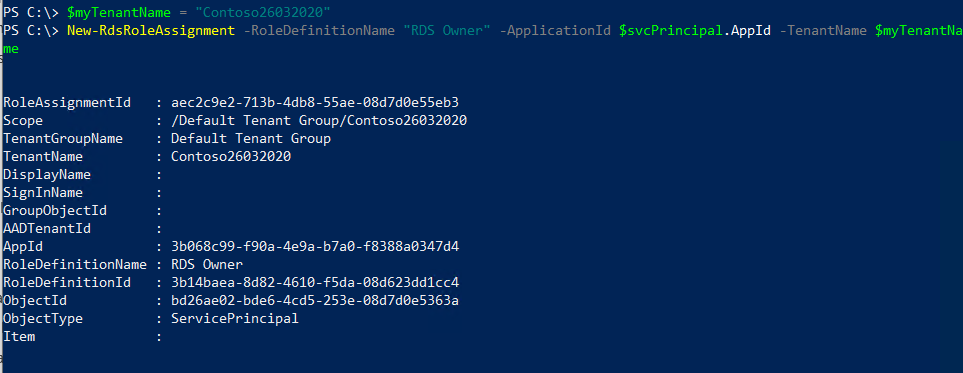
Get-RdsTenant



When you find the tenant name for the tenant you want to create a role assignment for, use that name in the following cmdlet:

$myTenantName = "<Windows Virtual Desktop Tenant Name>"

New-RdsRoleAssignment -RoleDefinitionName "RDS Owner" -ApplicationId $svcPrincipal.AppId -TenantName $myTenantName

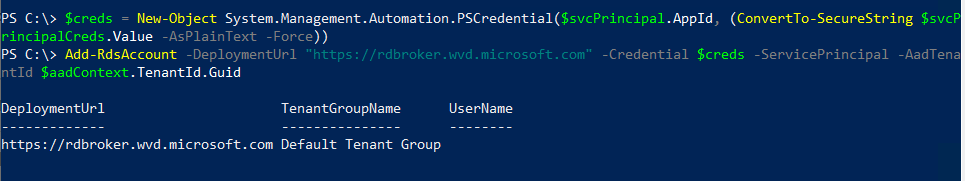


### Task 4: Sign in with the service principal

After you create a role assignment for the service principal, make sure the service principal can sign in to Windows Virtual Desktop by running the following cmdlet:

$creds = New-Object System.Management.Automation.PSCredential($svcPrincipal.AppId, (ConvertTo-SecureString $svcPrincipalCreds.Value -AsPlainText -Force))

Add-RdsAccount -DeploymentUrl "https://rdbroker.wvd.microsoft.com" -Credential $creds -ServicePrincipal -AadTenantId $aadContext.TenantId.Guid



## Exercise 6: Create a host pool by using the Azure ARM Template

Host pools are a collection of one or more identical virtual machines within Windows Virtual Desktop tenant environments. Each host pool can contain an app group that users can interact with as they would on a physical desktop.

### Prerequisites

* A tenant in Windows Virtual Desktop
* [Windows Virtual Desktop PowerShell module](https://docs.microsoft.com/en-us/powershell/windows-virtual-desktop/overview/) to be installed on VM/PC to configure the WVD tenant.
* Azure ADDS Instance. If you need to setup a new follow the [link](#_Create_and_configure)

Once the above requirements are met, then to Deploy the ARM Template, click on the below button

### Task 1: Deploying the ARM Template to create the Windows Virtual Desktop Host Pool.

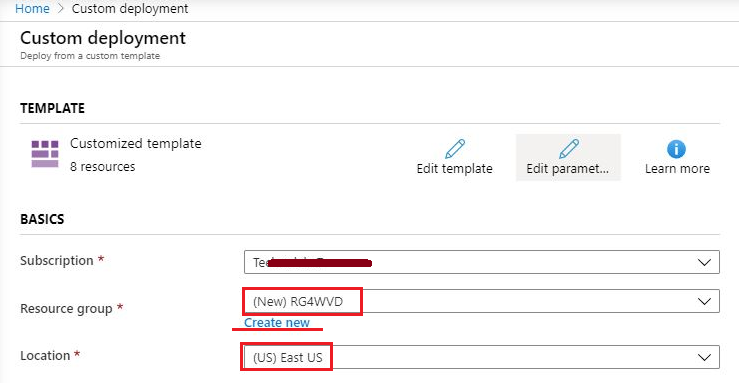
[](https://portal.azure.com/#create/Microsoft.Template/uri/https%3A%2F%2Fraw.githubusercontent.com%2FAzure%2FRDS-Templates%2Fmaster%2Fwvd-templates%2FCreate%20and%20provision%20WVD%20host%20pool%2FmainTemplate.json)

This template creates virtual machine(s) and registers them as session hosts to a new or existing Windows Virtual Desktop host pool. There are multiple sets of parameters you must enter to successfully deploy the template:

* VM image
* VM configuration
* Domain and network properties
* Authentication to Windows Virtual Desktop

Let first fill in the **Basics**

* Select the **Subscription** where you have you Azure ADDS configured
* Create a New Resource group – **RG4WVD**
* Select the **Location** where the Azure ADDS is configured.



**VM image**

When creating the virtual machines, you have three options:

* Azure Gallery image
* Custom VHD from blob storage
* Custom Azure Image resource from a resource group

Select Azure Gallery Image

By selecting Azure Gallery Image, you can select up-to-date images provided by Microsoft and other publishers. Enter or select values for the following parameters:

* Rdsh Image Source, select **Gallery**.
* Rdsh Gallery Image **SKU - windows-10-enterprise-multi-session-with-office-365-proplus**

**Ignore the following parameters:**

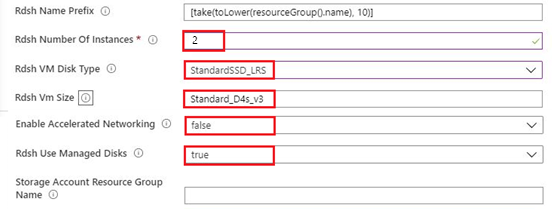
* Vm Image Vhd Uri
* Rdsh Custom Image Source Name
* Rdsh Custom Image Source Resource Group
* Rdsh Use Managed Disks
* Storage Account Resource Group Name



**VM configuration**

Enter the remaining configuration parameters for the virtual machines.

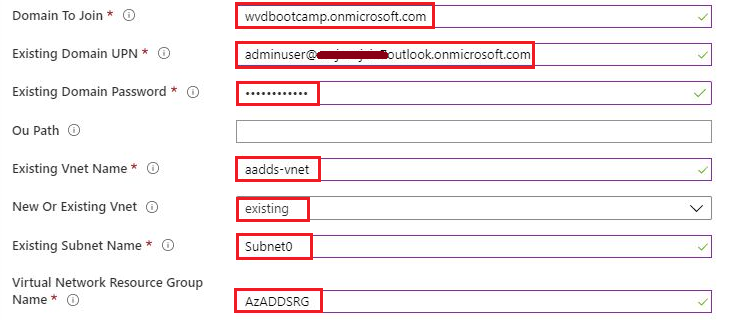
* Vm Size – **Standard\_D4s\_v3**
* Enable Accelerated Networking. - **False.**
* Rdsh Name Prefix – **default will use the Resource group name**
* Rdsh Number Of Instances - **2**
* Rdsh VM Disk Type – **StandardSSD\_LRS**
* Rdsh Use Managed Disks - **True**



**Domain and network properties**

Enter the following properties to connect the virtual machines to the appropriate network and join them to the appropriate domain (and organizational unit, if defined).

* Domain To Join – Name of your **Azure ADDS domain**.
* Existing Domain UPN – **UPN of account with Global Admin permissions on the Azure ADDS.**
* Existing Domain Password – **Password of the UPN account given above.**
* OU Path. If you do not have a specific organization unit for the virtual machines to join, **leave this parameter empty**.
* Existing Vnet Name – **aadds-vnet** [Azure ADDS vnet with DNS configured]
* Existing Subnet Name – **Subnet0** [Subnet with connectivity to Azure ADDS]
* Virtual Network Resource Group Name – **AzADDSRG** Resource group having the Azure ADDS and Vnet.



**Authentication to Windows Virtual Desktop**

Enter the following information to authenticate to Windows Virtual Desktop and register the new virtual machines as session hosts to a new or existing host pool.

* Rd Broker URL - **https://rdbroker.wvd.microsoft.com**
* Existing Tenant Group Name - leave this value as "**Default Tenant Group**".
* Existing Tenant Name - **Contoso29032020**
* Host Pool Name – Give a Name for Host Pool – **HostPool29032020**
* Tenant Admin Upn or Application Id - **UPN of account with RDS Owner or RDS Contributor permissions on WVD Tenant.**

[!WARNING] You cannot enter a UPN that requires MFA to successfully authenticate. If you do, this template will create the virtual machines but fail to register them to a host pool.

* Tenant Admin Password – **Password of the UPN used above**
* Is Service Principal – False
* Default Desktop Users - Make sure you don't use spaces between the commas example - user1@contoso.com,user2@contoso.com,user3@contoso.com.



Agree to the terms and condition and then click on **Purchase** to start the Deployment.

### Task 2 (Optional): Assign additional users to the desktop application group

After Azure ARM Template deployment finishes creating the pool, you can assign more users to the desktop application group. If you don't want to add more, skip this section.

To assign users to the desktop application group:

1. Open a PowerShell window.
2. Run the following command to sign in to the Windows Virtual Desktop environment:

Add-RdsAccount -DeploymentUrl "https://rdbroker.wvd.microsoft.com"

1. Add users to the desktop application group by using this command:

Add-RdsAppGroupUser -TenantName <tenantname> -HostPoolName <hostpoolname> -AppGroupName "Desktop Application Group" -UserPrincipalName <userupn>

Add-RdsAppGroupUser -TenantName $tenantName -HostPoolName $HostPoolName -AppGroupName “Desktop Application Group” -UserPrincipalName User@Contoso.com

Users you add to the desktop application group can sign in to Windows Virtual Desktop with supported Remote Desktop clients and see a resource for a session desktop.

Here are the current supported clients:

* [Remote Desktop client for Windows 7 and Windows 10](https://docs.microsoft.com/en-us/azure/virtual-desktop/connect-windows-7-and-10)
* [Windows Virtual Desktop web client](https://docs.microsoft.com/en-us/azure/virtual-desktop/connect-web)

### Task 3**:** Validate the new Host Pools

1. Now, we will validate this newly created host pool
2. Open PowerShell and first connect to the WVD tenant using below commands.

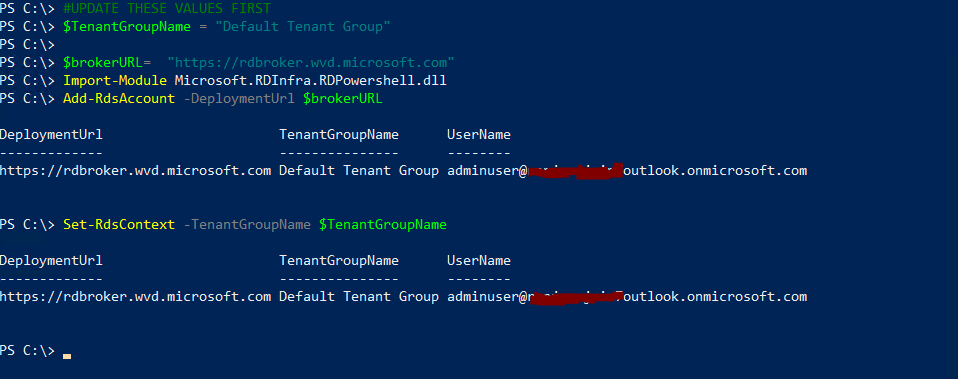
$TenantGroupName = “Default Tenant Group”

$brokerURL= “https://rdbroker.wvd.microsoft.com[”](https://rdbroker.wvd.microsoft.com/)

Import-Module Microsoft.RDInfra.RDPowershell.dll

Add-RdsAccount -DeploymentUrl $brokerURL

Set-RdsContext -TenantGroupName $TenantGroupName

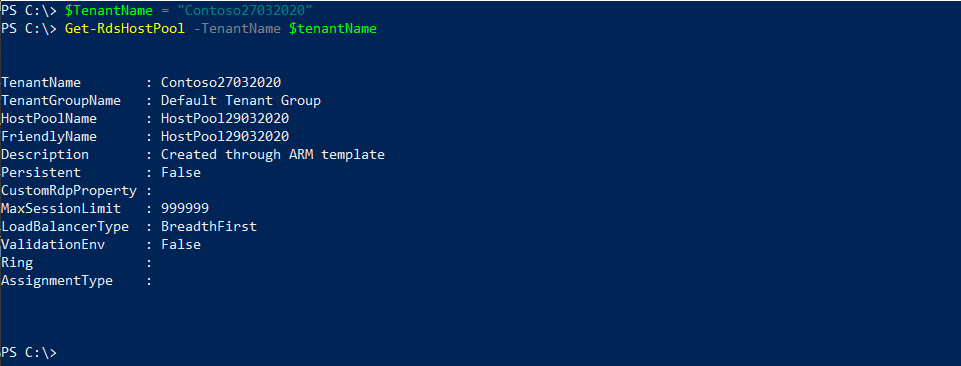


1. Check for the new Host Pool using below command

#UPDATE THESE VALUES FIRST

$TenantName = “Contoso27032020”

Get-RdsHostPool -TenantName $tenantName

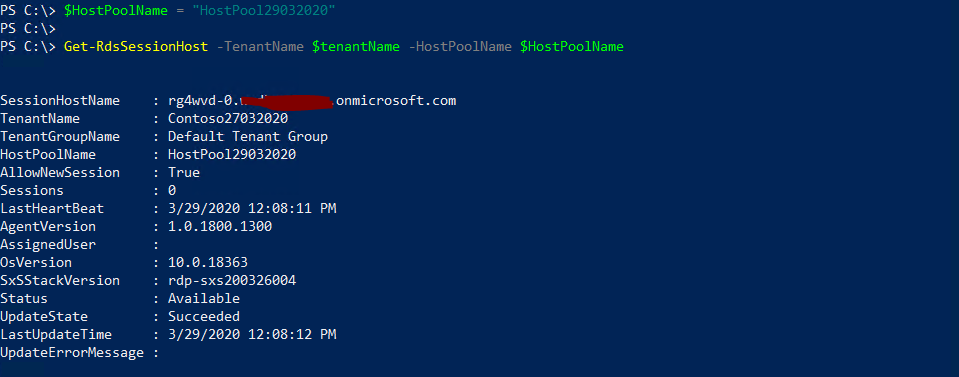


1. Check for the Session hosts in the Host Pool and ensure the status is **Available**

#UPDATE THESE VALUES FIRST

$HostPoolName = “HostPool29032020”

Get-RdsSessionHost -TenantName $tenantName -HostPoolName $HostPoolName

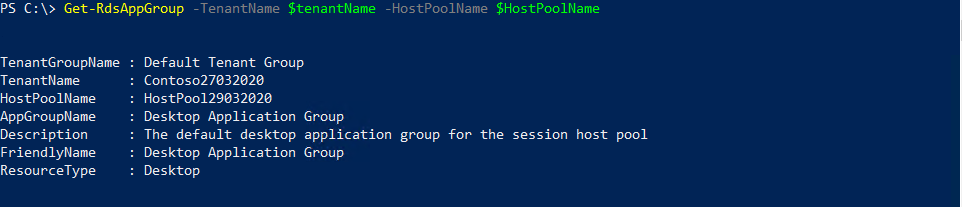


## Exercise 7: Manage App Groups

The default app group is automatically created for a new host pool that publishes the full desktop. In addition, you can create one or more application groups for the host pool. In this section, we will create a RemoteApp AppGroup and publish individual Start menu apps.

1. Check the Default Desktop Application Group is automatically created using below command

Get-RdsAppGroup -TenantName $tenantName -HostPoolName $HostPoolName

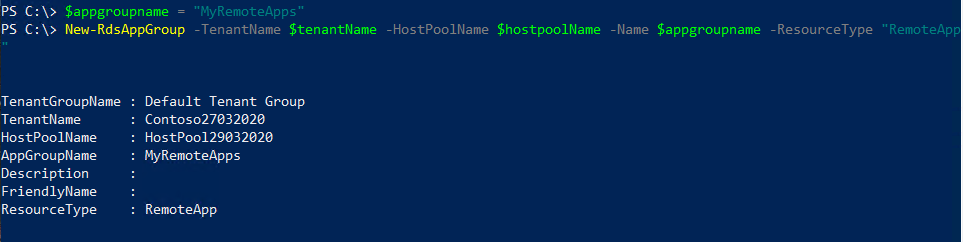


1. Now run the following PowerShell cmdlet to create a new empty RemoteApp group

#UPDATE THESE VALUES FIRST

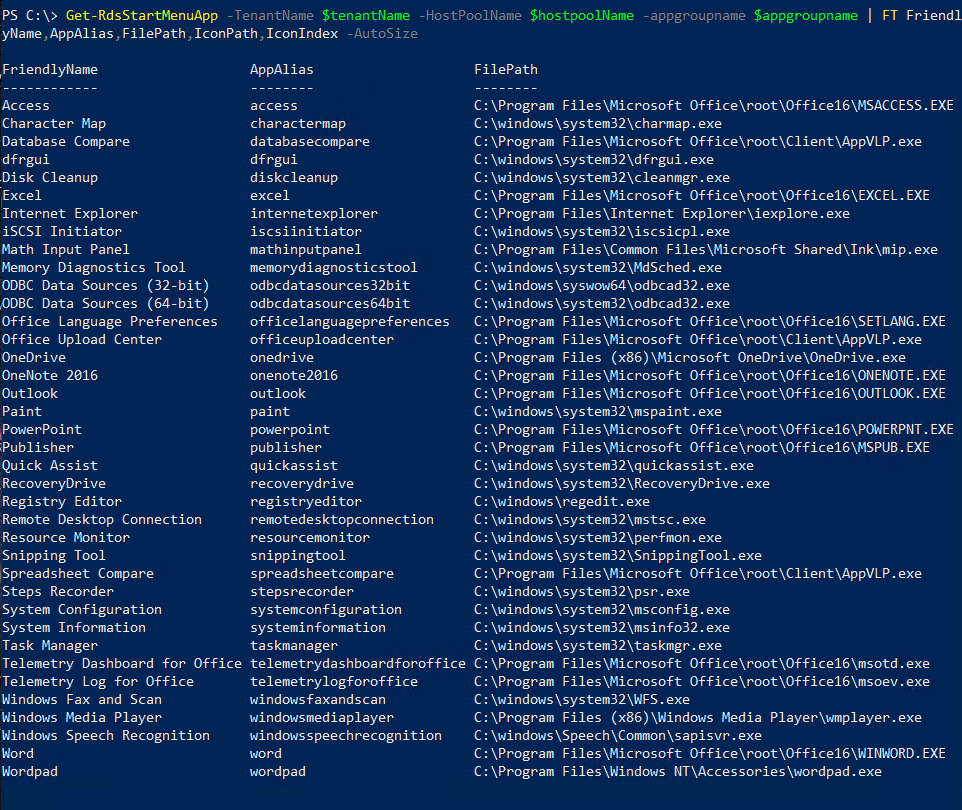
$appgroupname = “MyRemoteApps”

New-RdsAppGroup -TenantName $tenantName -HostPoolName $hostpoolName -Name $appgroupname -ResourceType “RemoteApp”



1. Run the following cmdlet to get a list of start menu apps on the host pool's virtual machine image. Write down the values for FilePath, IconPath, IconIndex, and other important information for the application you want to publish.

Get-RdsStartMenuApp -TenantName $tenantName -HostPoolName $hostpoolName -appgroupname $appgroupname | FT FriendlyName,AppAlias,FilePath,IconPath,IconIndex -AutoSize



1. Run the following cmdlet to publish a new RemoteApp to the application group and you will need the values from the above command to be used here.

#updates these variables with corresponding values form above command that you saved.

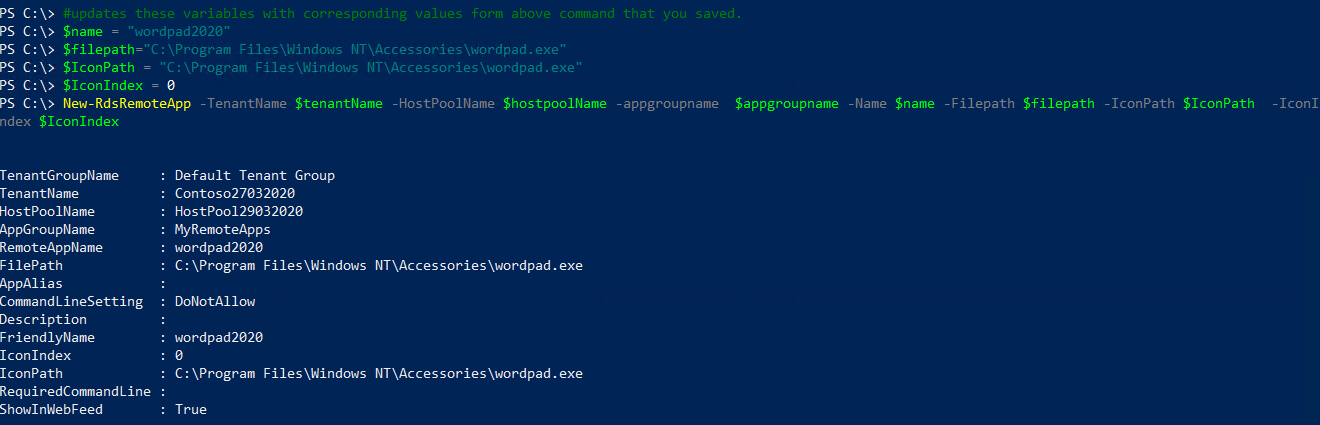
$name = “wordpad2020”

$filepath=“C:\Program Files\Windows NT\Accessories\wordpad.exe”

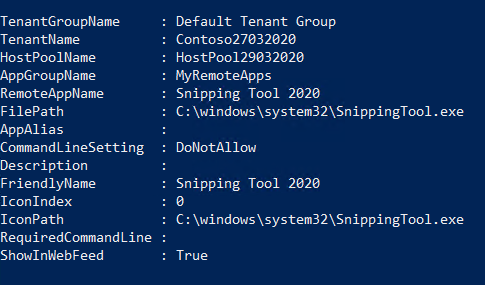
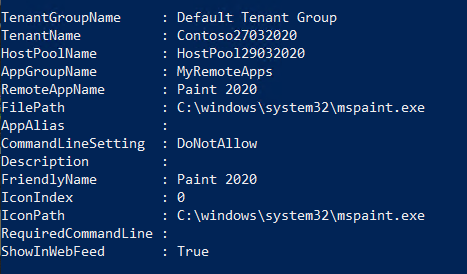
$IconPath = “C:\Program Files\Windows NT\Accessories\wordpad.exe”

$IconIndex = 0

New-RdsRemoteApp -TenantName $tenantName -HostPoolName $hostpoolName -appgroupname $appgroupname -Name $name -Filepath $filepath -IconPath $IconPath -IconIndex $IconIndex

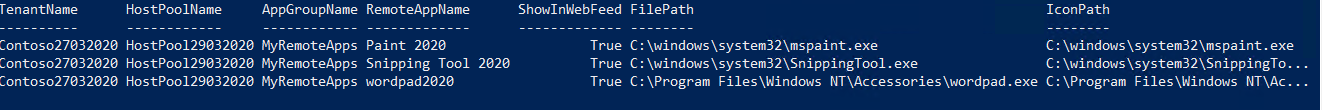


Now, update the variables and repeat the above commands for any other applications you want to publish. As an example, we could publish Paint & Snipping Tool in addition to WordPad.

1. To verify that the app was published, run the following cmdlet.

Get-RdsRemoteApp -TenantName $tenantName -HostPoolName $hostpoolName -AppGroupName $appgroupname | FT TenantName,HostPoolName,AppGroupName,RemoteAppName,ShowInWebFeed,FilePath,IconPath,IconIndex



1. Run the following cmdlet to grant users access to the RemoteApps in the app group

#UPDATE THESE VALUES FIRST

$appgroupname = “MyRemoteApps”

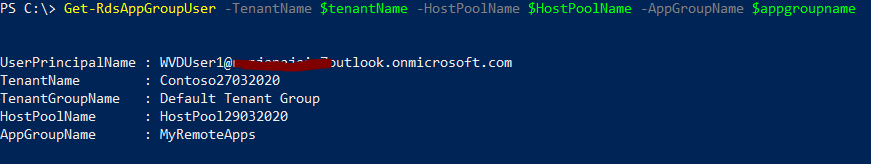
$upn = “rdsuser1@domain.com” #this should be the user that will access WVD resources from your domain

Add-RdsAppGroupUser -TenantName $tenantName -HostPoolName $HostPoolName -AppGroupName $appgroupname -UserPrincipalName $upn



#check the ACL has been applied using

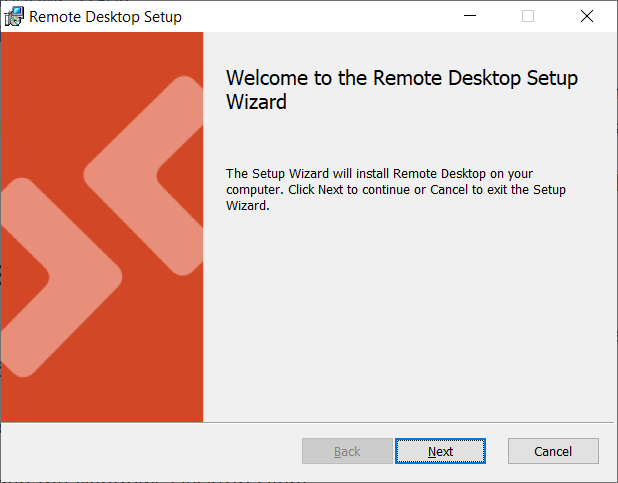
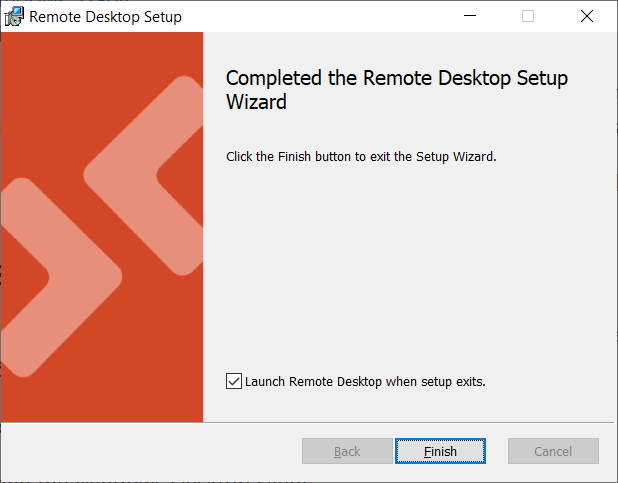
Get-RdsAppGroupUser -TenantName $tenantName -HostPoolName $HostPoolName -AppGroupName $appgroupname



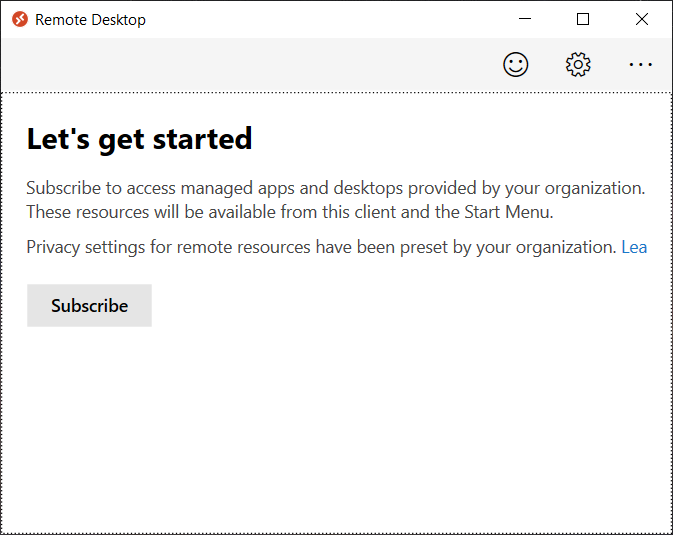
## Exercise 8: End User Experience

At this stage, your RemoteApps are deployed on the WVD session hosts for the end user profile management. A downloadable client is available that provides access to Windows Virtual Desktop resources from devices running Windows 7 and Windows 10 OR there is also a web client that can be used.

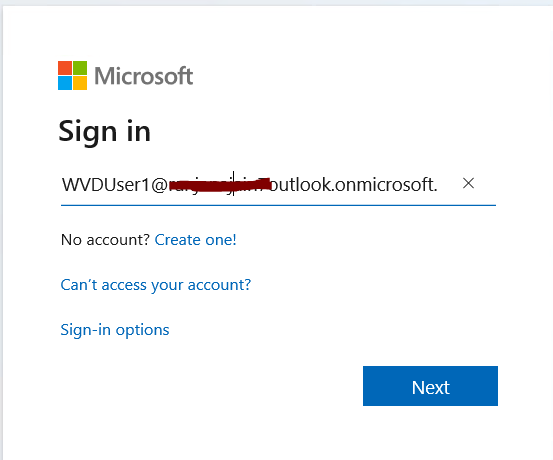
1. [Download the client](https://go.microsoft.com/fwlink/?linkid=2068602) and run the MSI to complete the installation. [Link](https://docs.microsoft.com/en-us/azure/virtual-desktop/connect-windows-7-and-10#install-the-windows-desktop-client)

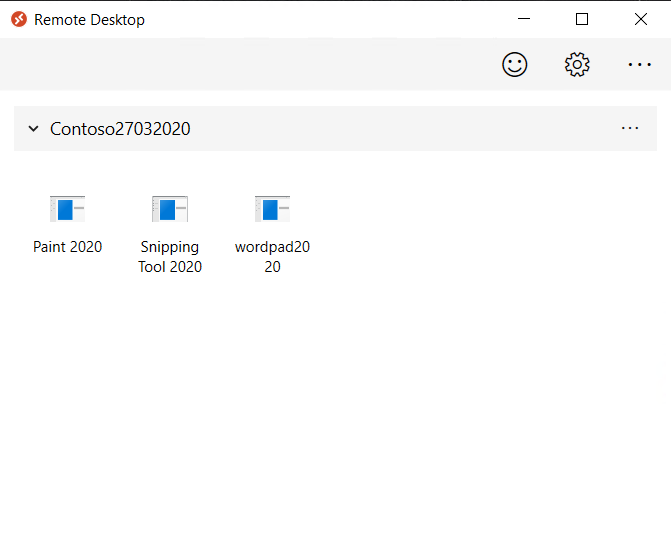
1. Start the client from the All Apps List, look for Remote Desktop.



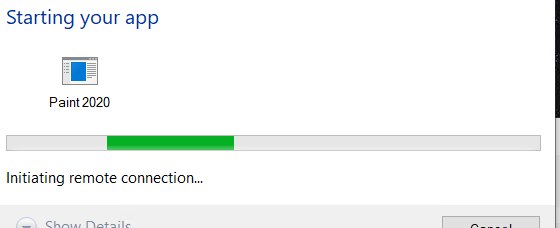
1. Click Subscribe and then sign in



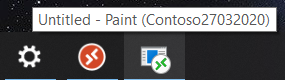
1. Once signed in you would be able to see the assigned resources.



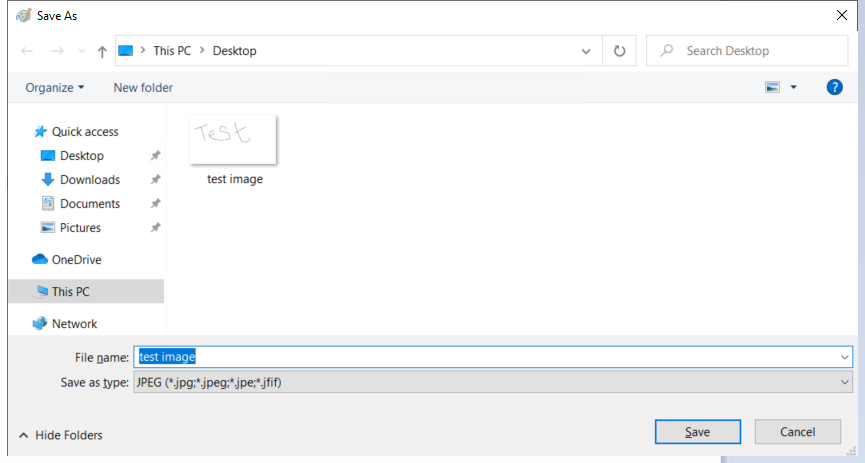
1. Please launch any of the resources (EX: Paint2020). *please be advised that the first launch may be slow as your user profile is being created.*



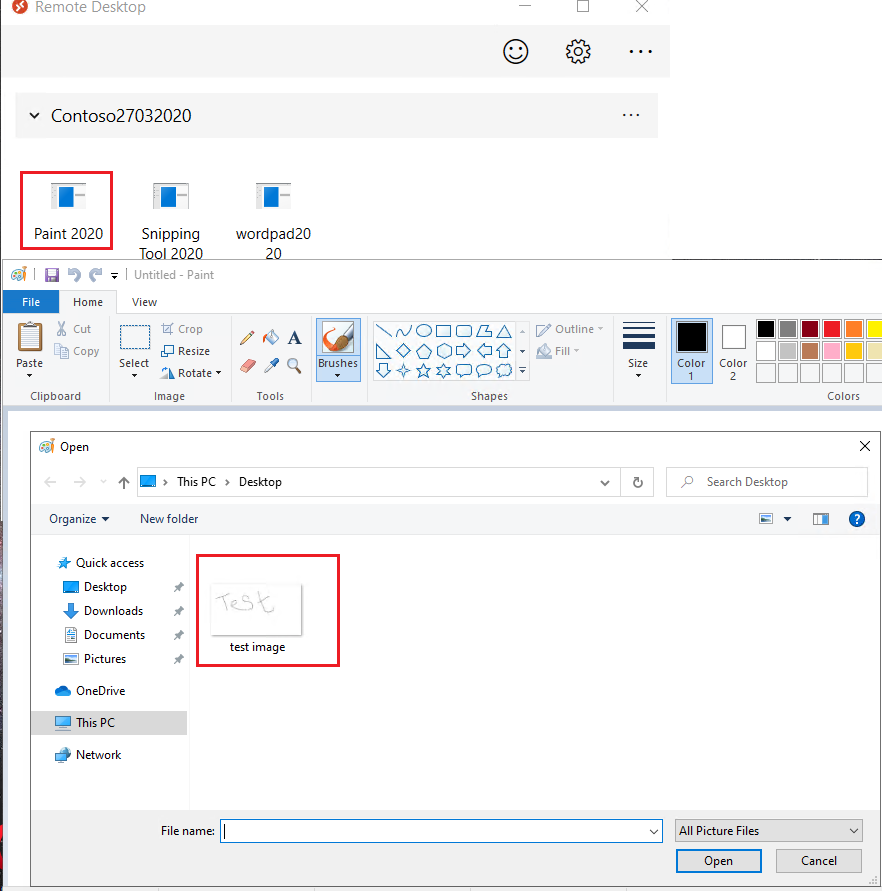
1. You may be required to re-authenticate, once launched, you can see the icon in your taskbar.



1. Now save your file > close Paint. This file will be saved on the Session host under the user’s profile.



1. Once again launch Paint from the WVD client > Ctrl +O > to see you document present.

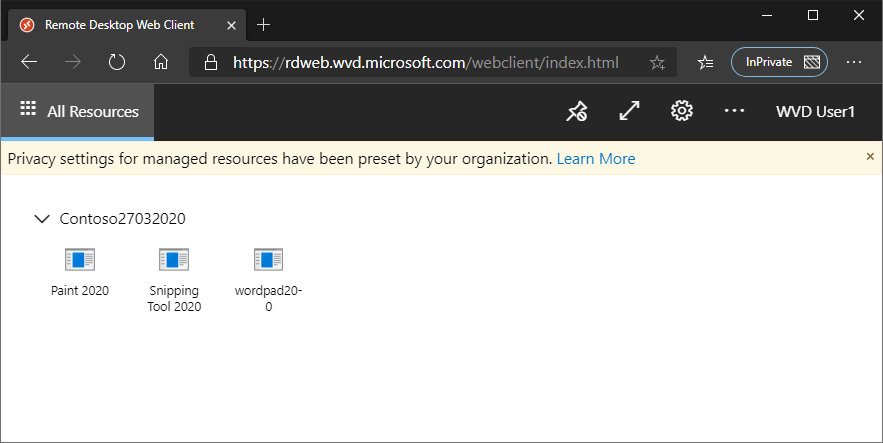


1. Alternatively, you can have a similar connection experience using a web browser by following the steps below.

NOTE: the browser must be HTML-5 compatible. Supported ones include latest versions of IE/Edge/Safari/Firefox/Chrome

o Going to [https://rdweb.wvd.microsoft.com](https://rdweb.wvd.microsoft.com/)  o Login with user domain credentials

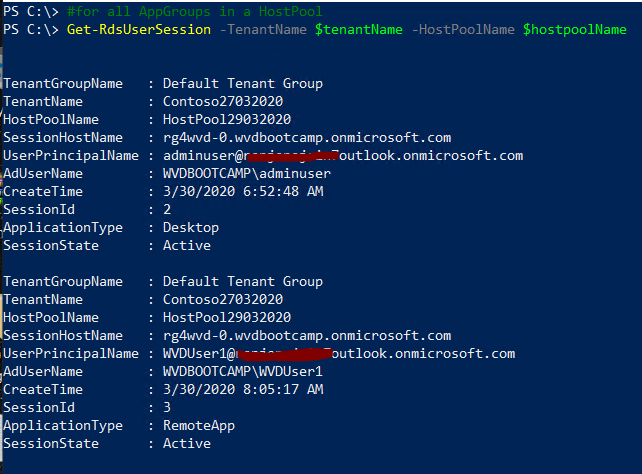
o Access Apps & Desktops



1. As an Admin, you can also validate the User Session data from the WVD end using either of the commands.

#for all AppGroups in a HostPool

Get-RdsUserSession -TenantName $tenantName -HostPoolName $hostpoolName



# Lab 2: WVD Management

## Exercise 1: Configure the Load balancing Method

Explanation about the different LB methods is found at [Configure the Windows Virtual Desktop load-balancing method](https://docs.microsoft.com/en-us/azure/virtual-desktop/configure-host-pool-load-balancing) . Below are a couple of screenshots that confirm how the session allocation across the session hosts changes with the Load balancing configuration.

**Configure breadth-first load balancing**

Set-RdsHostPool -TenantName $tenantName -HostPoolName $HostPoolName -BreadthFirstLoadBalancer -MaxSessionLimit 10

**Configure depth-first load balancing**

Set-RdsHostPool -TenantName $tenantName -HostPoolName $HostPoolName -DepthFirstLoadBalancer -MaxSessionLimit 10

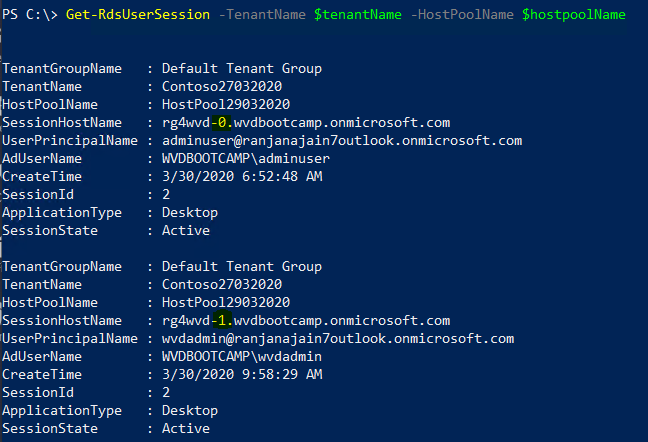
### Task 1: Breadth First

Session Allocation using Breadth-First for a HostPool with 2 VM’s. RDSUser1 & RDSUser2 are scattered across the VM’s for better using experience.

Get-RdsHostPool -TenantName $tenantName

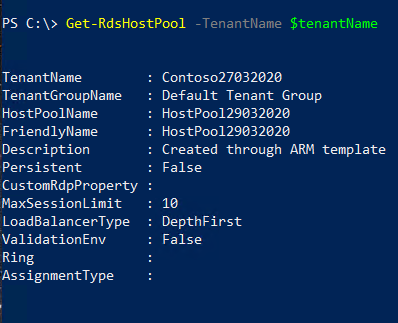


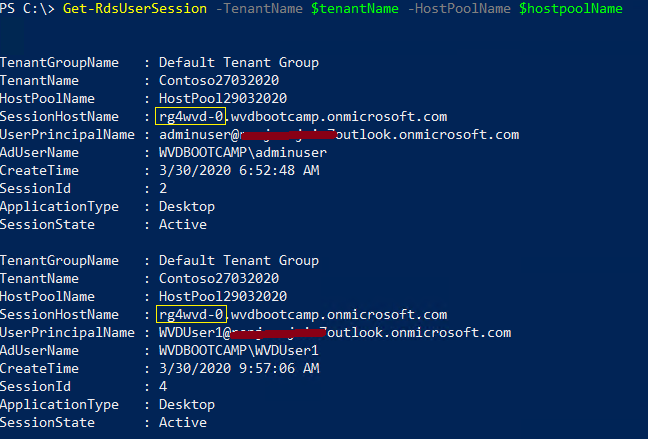
Get-RdsUserSession -TenantName $tenantName -HostPoolName $hostpoolName



### Task 2: Depth First

Session Allocation using Depth-First for a HostPool with 2 VM’s. RDSUser1 & RDSUser2 are logged onto the same VM till the HostPool session limit threshold is met.

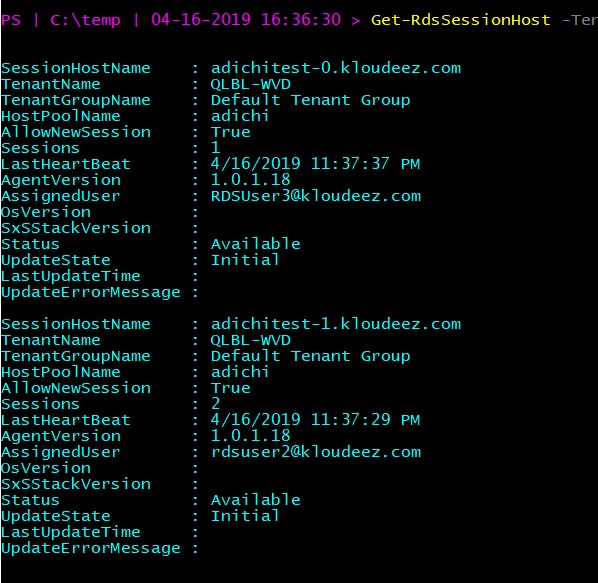
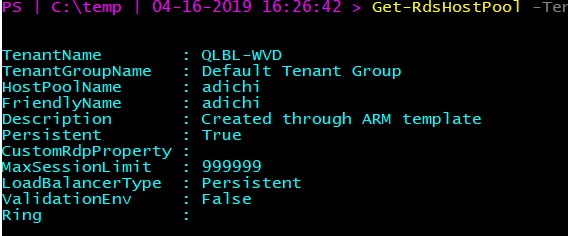




### Task 3: Persistent Desktops

Persistent desktops can only be created at deployment time. A typical use case scenario would be a VDI like environment. Here, users are auto-assigned an available session host during the first logon and any subsequent logins are directed to the same VM.

Unlike multi user session, persistence follows a 1:1 mapping between users & session hosts. For Example: if the HostPool has 5 VM’s, they will be assigned to the first 5 users and the 6’th user will get an error that enough resources (VMs) are unavailable.

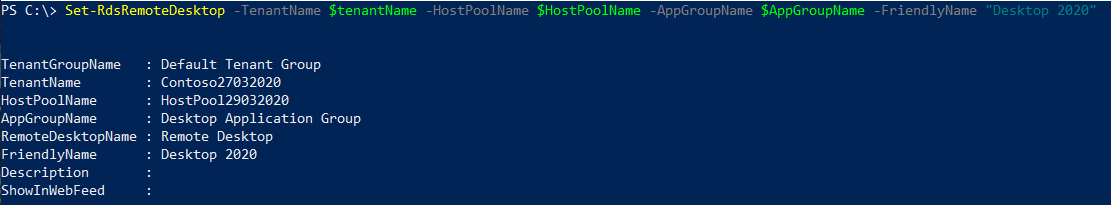


## Exercise 2: Set Friendly Names for Published Desktops

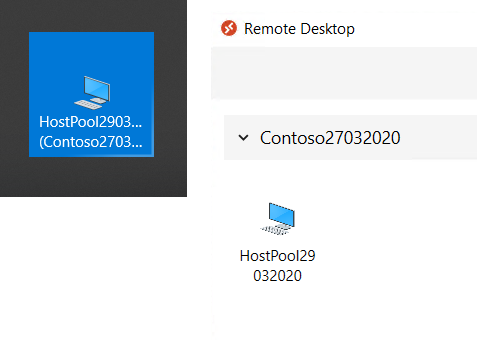
Using the below command, you can set friendly names to uniquely identify multiple desktops published to a user.

#Update the respective $variables first and then execute command

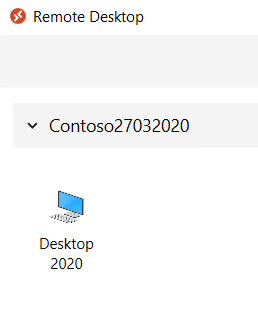
Set-RdsRemoteDesktop -TenantName $tenantName -HostPoolName $HostPoolName -AppGroupName $AppGroupName -FriendlyName "Desktop 2020"



***Before***



***After***



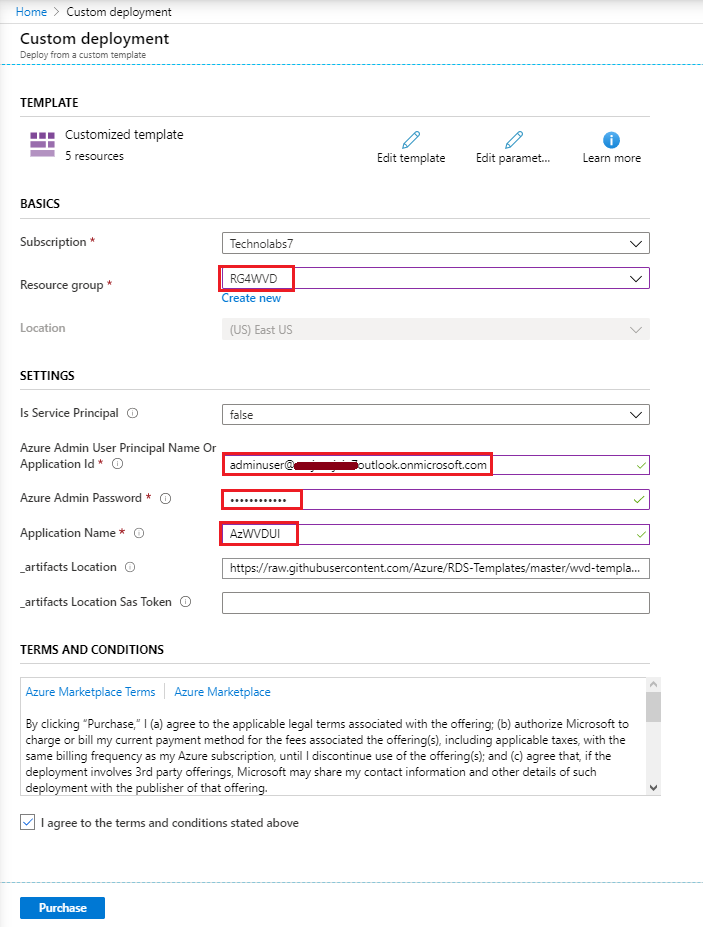
## Exercise 3: Deploy the Management UI

Follow the steps below to deploy the WVD management UI and be able to manage some aspects of WVD like provisioning, modifications etc.

1. Click the button below to deploy:

[Deploy to Azure](https://portal.azure.com/#create/Microsoft.Template/uri/https%3A%2F%2Fraw.githubusercontent.com%2FAzure%2FRDS-Templates%2Fmaster%2Fwvd-templates%2Fwvd-management-ux%2Fdeploy%2FmainTemplate.json)

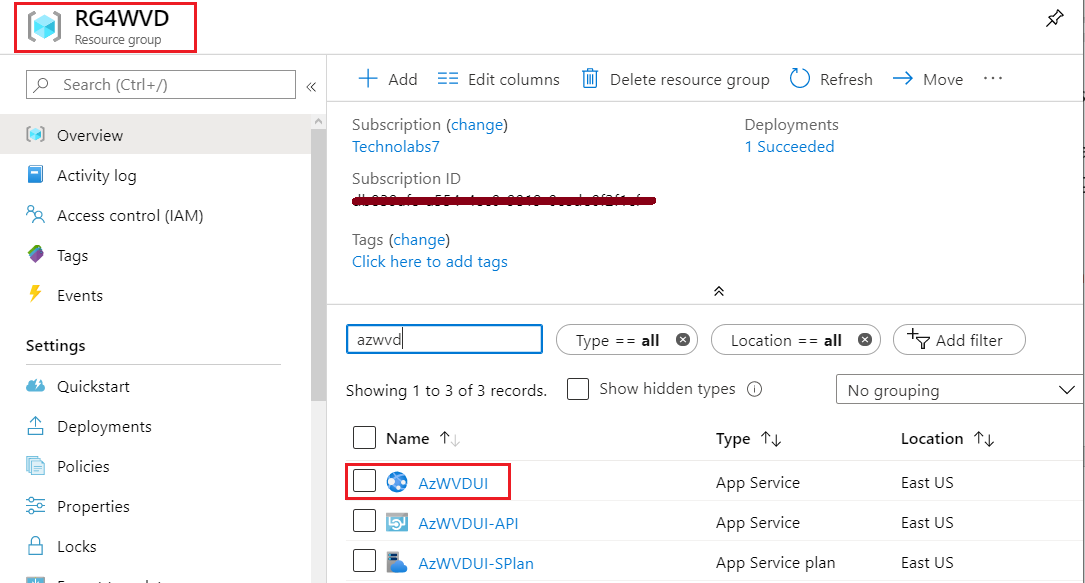
1. You may be required to sign in again.
2. ON the custom deployment page, provide the following info on the service:
   * + Resource Group Name: Same as the WVD Host Pool – **RG4WVD**
     + Azure Login ID/ password – please give an admin that **does not have MFA enforced**.
     + Input the APP name to something unique in your subscription (e.g. **AzWVDUI**xxxx)



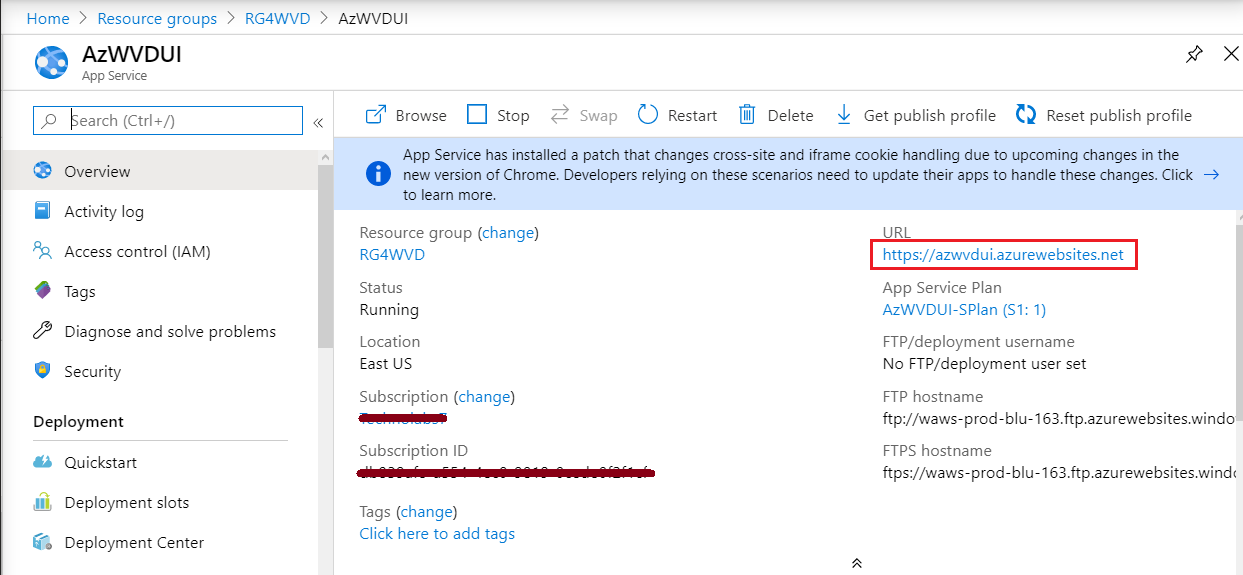
This will create a resource group and will have 2 app services along with 1 app service plan:

1. To launch the UX, click on app service with name you provided in the template

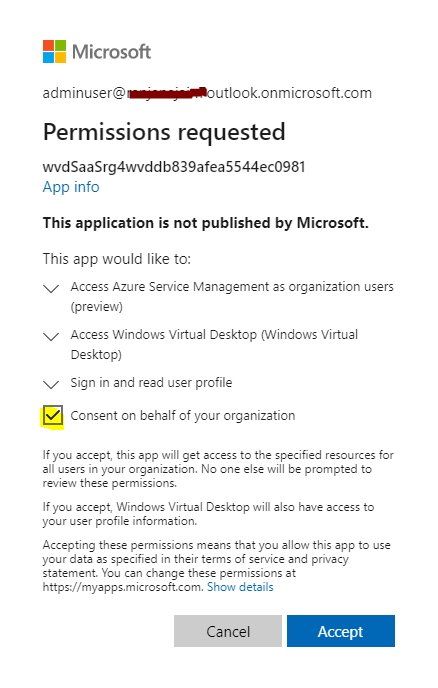
(e.g. **AzWVDUI**xxxx) and navigate to the URL that shows up on the top right side



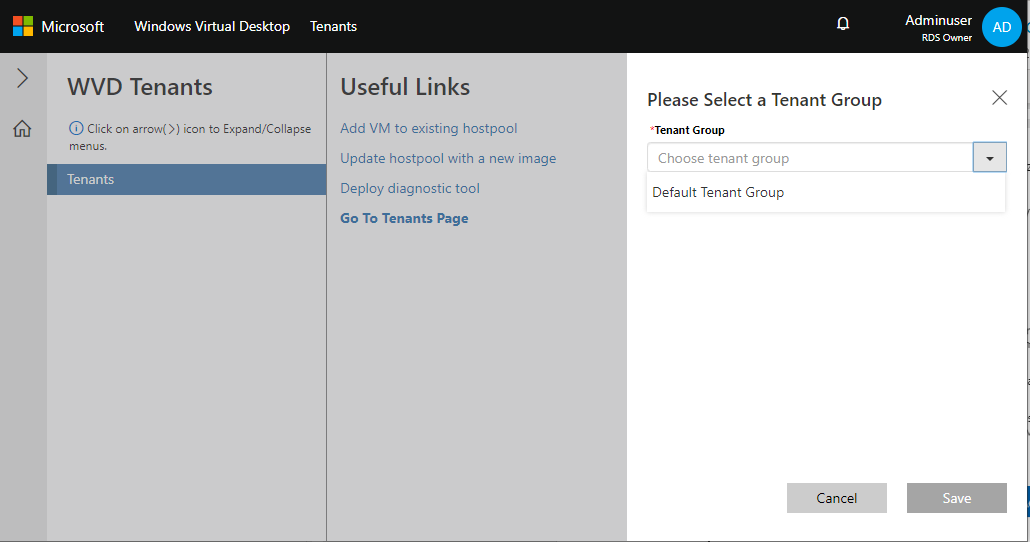
Click on the App Service. Then click on the URL to launch the Application UX.



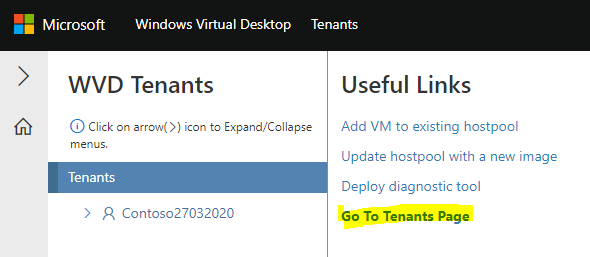
You may be required to sign in and then Consent on behalf of your organization, select **Accept**.



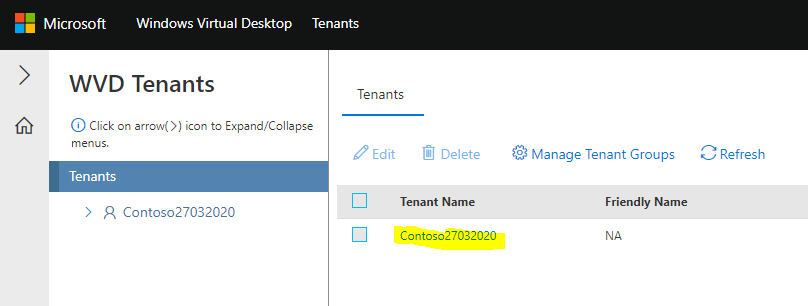
Select the Default Tenant Group and click save.



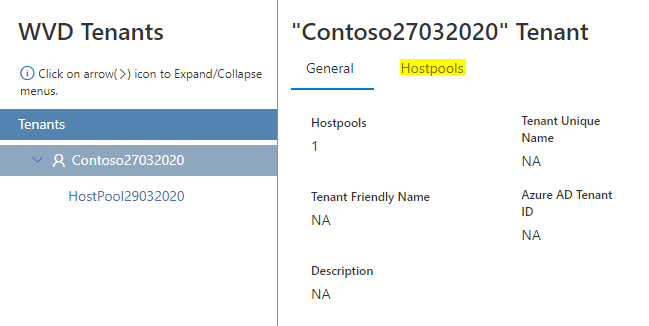
From the useful links click on **Go to Tenant Page**



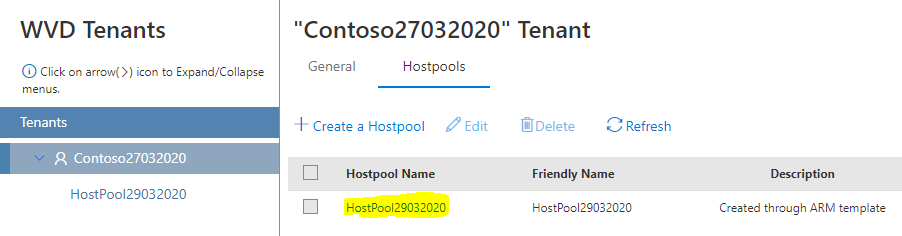
Select the Tenant – **Contoso27032020**



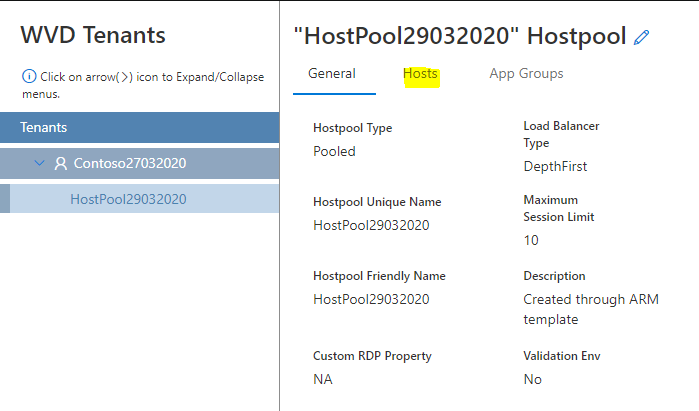
Select the **Hostpools**



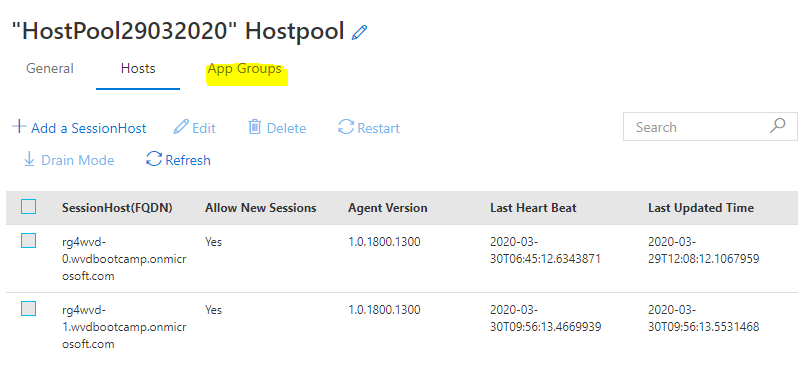
Select the Hostpool – **HostPool29032020**



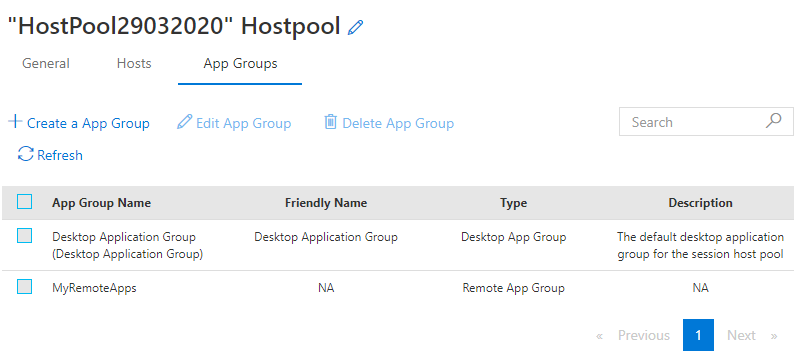
Here you can see the General details about the Hostpool. Click on **Hosts** to see the **VMs** in this HostPool.



Here we can see both the VMs details, which we had seen earlier only through powershell, now click on the **App Groups**



Here we can see the Default App Group and Custom App Group we created.



This complete the demo for the Management UX.

# Appendix

# Exercise 1: Create and configure an Azure Active Directory Domain Services instance

Azure Active Directory Domain Services (Azure AD DS) provides managed domain services such as domain join, group policy, LDAP, Kerberos/NTLM authentication that is fully compatible with Windows Server Active Directory.

**Prerequisites**

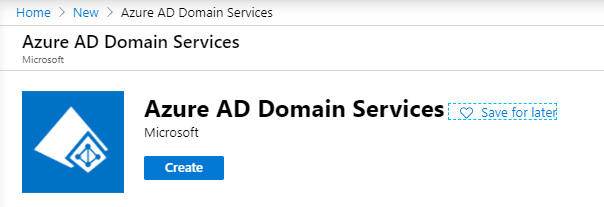
* An active Azure subscription.
* An Azure Active Directory tenant associated with your subscription, either synchronized with an on-premises directory or a cloud-only directory.
* You need global administrator privileges in your Azure AD tenant to enable Azure AD DS.
* You need Contributor privileges in your Azure subscription to create the required Azure AD DS resources.

**Sign in to the Azure portal –** <Https://Portal.Azure.com>

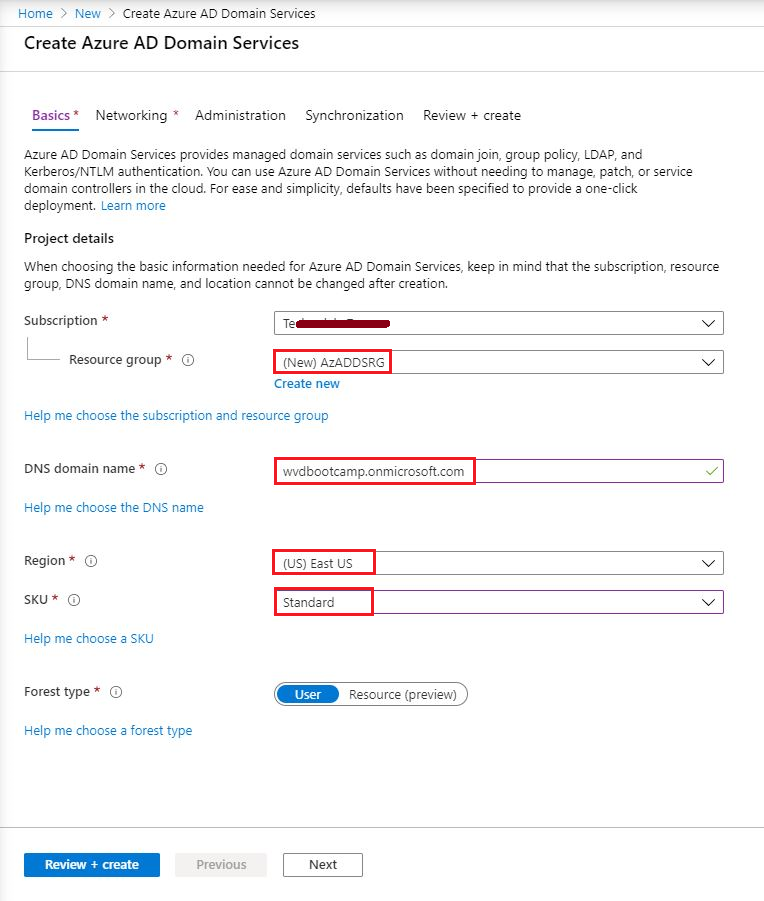
**Create an instance**

To launch the Enable Azure AD Domain Services wizard, complete the following steps:

* On the Azure portal menu or from the Home page, select **Create a resource**.
* Enter **Domain Services** into the search bar, then choose **Azure AD Domain Services** from the search suggestions.
* On the Azure AD Domain Services page, select **Create**.



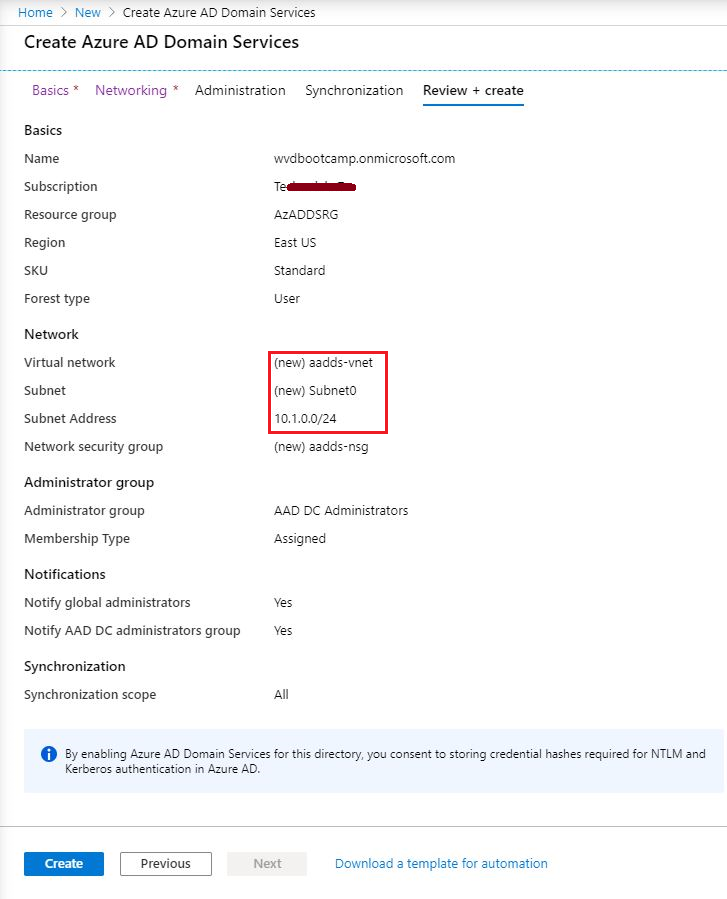
* Select the **Azure Subscription** in which you would like to create the managed domain.
* Choose to Create new or select an existing resource group. - **AzADDSRG**
* DNS Domain Name – **wvdbootcamp.onmicrosoft.com**
  + When you create an Azure AD DS instance, you specify a DNS name. There are some considerations when you choose this DNS name:
    - **Built-in domain name**: By default, the built-in domain name of the directory is used
    - **Custom domain names:** The most common approach is to specify a custom domain name
    - **Non-routable domain suffixes:** We generally recommend that you avoid a non-routable domain name suffix
* Choose the Azure **Location** in which the managed domain should be created – **East US**
* Select the **Standard** SKU.
* By default, an Azure AD DS managed domain is created as a **User forest**.
* Select **Next**, to configure the Networking.



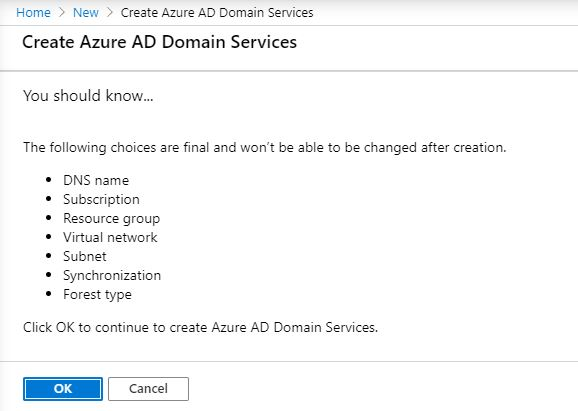
To quickly create an Azure AD DS managed domain, you can select **Review + create** to accept additional default configuration options. The following defaults are configured when you choose this create option:

* Creates a virtual network named aadds-vnet that uses the IP address range of 10.0.1.0/24.
* Creates a subnet named aadds-subnet using the IP address range of 10.0.1.0/24.
* Synchronizes All users from Azure AD into the Azure AD DS managed domain.
* Select Review + create to accept these default configuration options.

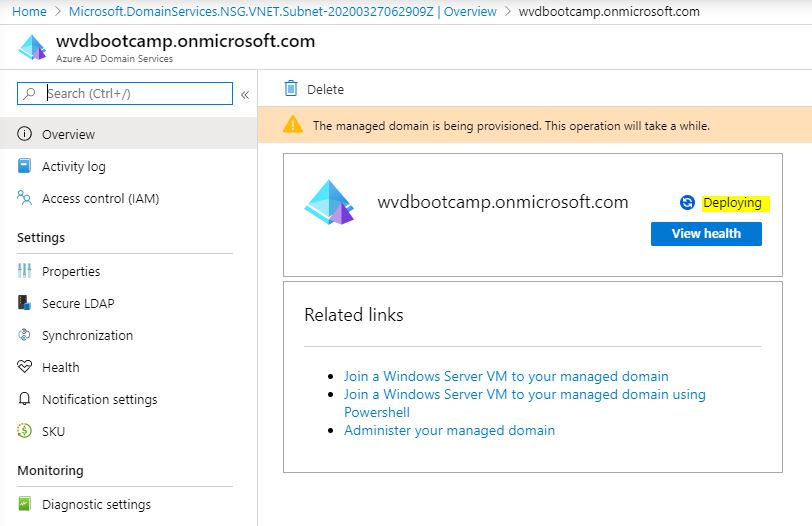
Click **Create** to deploy the managed domain.



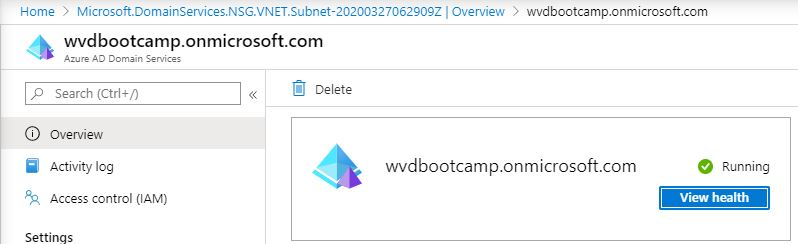
It will confirm once again, click ok to create the Azure ADDS.



Select your resource group, **AzADDSRG**, then choose your Azure AD DS instance from the list of Azure resources. The **Overview** tab shows that the managed domain is currently Deploying.



When the managed domain is fully provisioned, the **Overview** tab shows the domain status as Running.

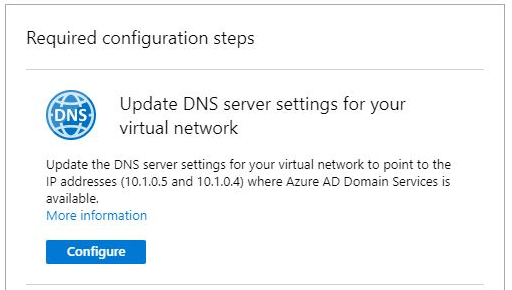


**Update DNS settings for the Azure virtual network**

With Azure AD DS successfully deployed, now configure the virtual network to allow other connected VMs and applications to use the managed domain.

The Overview tab for your managed domain shows some Required configuration steps. The first configuration step is to update DNS server settings for your virtual network. Once the DNS settings are correctly configured, this step is no longer shown.

To update the DNS server settings for the virtual network, select the **Configure** button. The DNS settings are automatically configured for your virtual network.





**Enable user accounts for Azure AD DS**

For cloud-only user accounts, users must change their passwords before they can use Azure AD DS. This password change process causes the password hashes for Kerberos and NTLM authentication to be generated and stored in Azure AD. You can either expire the passwords for all users in the tenant who need to use Azure AD DS, which forces a password change on next sign-in, or instruct them to manually change their passwords. For this tutorial, let's manually change a user password.