Windows Virtual Desktop (WVD) Proof of Concept (PoC): Technical Guide



**MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.**

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation. Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, our provision of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property. The descriptions of other companies’ products in this document, if any, are provided only as a convenience to you. Any such references should not be considered an endorsement or support by Microsoft. Microsoft cannot guarantee their accuracy, and the products may change over time. Also, the descriptions are intended as brief highlights to aid understanding, rather than as thorough coverage. For authoritative descriptions of these products, please consult their respective manufacturers. © 2016 Microsoft Corporation. All rights reserved. Any use or distribution of these materials without express authorization of Microsoft Corp. is strictly prohibited. Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Contents

[1. Introduction 5](#_Toc9346400)

[2. Target Audience 5](#_Toc9346401)

[3. Pre-Requisites/Requirements 5](#_Toc9346402)

[4. Planning & Design 6](#_Toc9346403)

[4.1. General Best Practices 6](#_Toc9346408)

[4.2. Azure networking 7](#_Toc9346409)

[4.3. Identity & Access Management 9](#_Toc9346410)

[4.4. Scale Out File Server (SOFS) with Storage Spaces Direct (S2D) 12](#_Toc9346411)

[4.5. WVD Session Host VM & Storage SKU Guidance 18](#_Toc9346412)

[5. WVD PoC Implementation Steps 21](#_Toc9346413)

[5.1. Grant Azure Active Directory permissions to the Windows Virtual Desktop service 21](#_Toc9346427)

[5.2. Assign the Tenant Creator Application role to a user in your Azure Active Directory. 21](#_Toc9346428)

[5.3. Download the WVD PowerShell Module 22](#_Toc9346429)

[5.4. Create the WVD Tenant 22](#_Toc9346430)

[5.5. Deploy the WVD Host Pools 24](#_Toc9346431)

[5.5.1. Deploy VMs with Client Operating System 24](#_Toc9346432)

[5.5.2. Deploy VMs with Server Operating System 24](#_Toc9346433)

[5.6. Validate the new Host Pools 25](#_Toc9346434)

[5.7. Manage App Groups 26](#_Toc9346435)

[5.8. O365 Optimization using FSLogix 30](#_Toc9346436)

[5.8.1. Install FSLogix on Session Hosts 30](#_Toc9346446)

[5.8.2. Configure FSLogix GPO Settings 30](#_Toc9346447)

[5.9. End User Experience 30](#_Toc9346448)

[5.9.1. Validate FSlogix Profile containers 34](#_Toc9346449)

[6. WVD Management 35](#_Toc9346470)

[6.1. Configure the Load balancing Method 35](#_Toc9346471)

[6.1.1. Breadth First 35](#_Toc9346472)

[6.1.2. Depth First 36](#_Toc9346473)

[6.1.3. Persistent Desktops 36](#_Toc9346474)

[6.2. Set Friendly Names for Published Desktops 38](#_Toc9346475)

[6.3. Deploy the Management UI 39](#_Toc9346476)

[6.4. Check Diagnostic data 40](#_Toc9346477)

[7. Appendix 40](#_Toc9346478)

[7.1. Deploy HostPool using Azure Portal (Marketplace) 40](#_Toc9346481)

[7.2. Deploy HostPool using ARM template 46](#_Toc9346482)

[7.3. Deploy HostPool using modified ARM template 46](#_Toc9346483)

# Introduction

The ***primary intent of this guide is to illustrate the implementation steps for a Windows Virtual Desktop (WVD) PoC (proof of concept) solution in Azure***. It is intended to be used by Customer & Partners to help familiarize themselves with the processes, methodologies and tools required to implement& manage a WVD solution in Azure.

# Target Audience

This document is ***Level 400+ technical guide*** primarily intended for Azure Specialists, Cloud Solution Architects, Migration experts, System Administrators & anyone else who are going to be hands-on in executing the technical steps to implementing the solution. It is assumed that the audience has deep insights into their on-premise workload architectures, storage & networking capabilities along with the interdependencies across multiple services/components involved like Active Directory, RDS deployments, Microsoft Azure and its core services (compute, storage & Network).

Please note that this document will primarily focus on the detailed process & methodologies and is NOT a primer for the technologies afore mentioned.

# Pre-Requisites/Requirements

Before getting started, **all** items listed below **must** be checked/validated to ensure the most basic requirements are in place to proceed with executing the remaining steps in this guide. ***For any reason, if you do NOT meet all requirements, then either get the required access or get in touch with a team/person who can help you achieve the same.***

* An Azure tenant (Ex: yourdomain.onmicrosoft.com) environment along with at least 1 active subscription.
  + If you are a customer, then reach out to your CSP partner who can provide you with the required tenant information and access
  + If you are the CSP partner, then you can get the customer details by logging onto the [Microsoft Partner Portal](https://partner.microsoft.com/en-US/) > dashboard > customers . Here you can see the domain under the column Primary Domain Name
* Ensure that the user who will provision & configure WVD must have “Global Admin” rights to the Azure tenant they are a part of.
  + Based on the operating model, some customers might not have this enabled so contact your CSP-Partner who can help with the same.
* Ensure that the user who will provision & configure WVD must have at least “Contributor” rights to the Azure subscription
  + Based on the operating model, some customers might not have this enabled so contact your CSP-Partner who can help with the same.
* **All** [**WVD requirements**](https://docs.microsoft.com/en-us/azure/virtual-desktop/overview#requirements) **must be satisfied.**
* Knowledge and comfortability in managing Azure services like:
  + Azure Networking (VNET/Subnets/NIC/NSGs)
  + Azure Active Directory (AAD), Azure Active Directory Domain Services (AAD-DS) & Azure AD Connect
  + Azure Compute (VMs/Availability Sets)
  + Azure Storage (disks/storage accounts)
  + Azure Networking (VNET/Subnets/NIC/NSGs)
  + Ability to work with command line implementation using PowerShell, Azure Modules.
  + The ability to manage ARM templates and deploy azure resources with it.
* Optionally knowledge of [FSlogix Profile Containers](https://docs.fslogix.com/)

# Planning & Design

**The idea of this section is to plan and design for your WVD infrastructure in a way that once the PoC is complete and successfully validated, the existing environment can be extended upon to serve as the production infrastructure.**



## General Best Practices

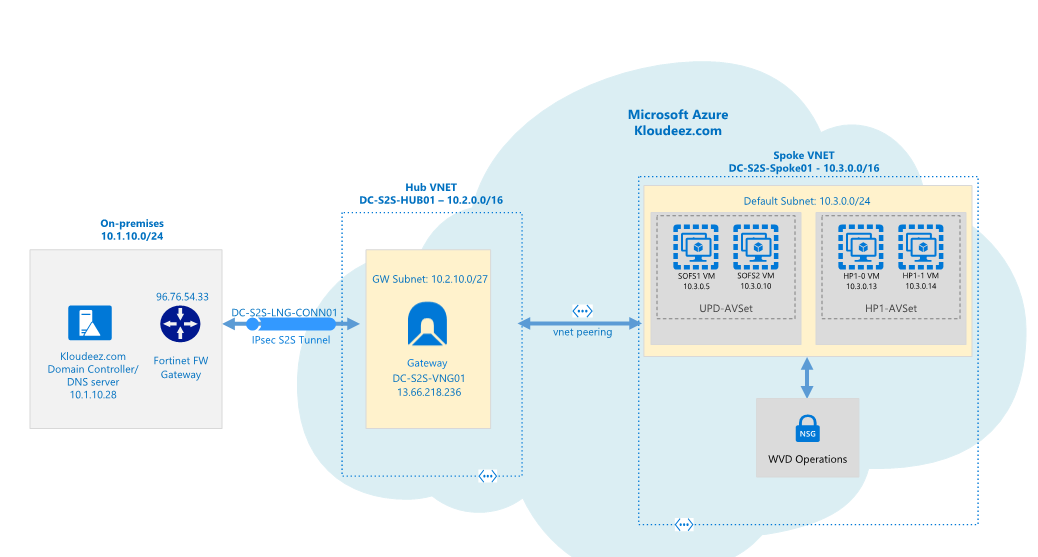
Since everyone’s business and technical requirements vary across the board, it is always a good idea to familiarize yourselves with the standard best practices across the different Azure technologies & services.

* Standard naming Conventions
  + Skip this section if you are already following a standard naming convention for resources on-prem and in Azure. If not, please follow the guidance in the [link](file:///C:/Users/risanc/AppData/Local/Microsoft/Windows/INetCache/Content.Outlook/X66Q8Y6H/•%09https:/docs.microsoft.com/en-us/azure/architecture/best-practices/naming-conventions) to maintain a consistent naming convention across your resources
* [Azure security best practices and patterns](https://docs.microsoft.com/en-us/azure/security/security-best-practices-and-patterns)
* Azure Active Directory Hybrid Identity [best practices](https://docs.microsoft.com/en-us/azure/active-directory/hybrid/plan-hybrid-identity-design-considerations-overview)
  + [Azure identity management and access control security best practices](https://docs.microsoft.com/en-us/azure/security/azure-security-identity-management-best-practices)
* Azure Networking & security [Best Practices](https://docs.microsoft.com/en-us/azure/security/azure-security-network-security-best-practices)
  + [Implementing a secure hybrid network architecture in Azure](https://docs.microsoft.com/en-us/azure/guidance/guidance-iaas-ra-secure-vnet-hybrid)
* Azure Storage security [overview](https://docs.microsoft.com/en-us/azure/security/security-storage-overview)
* [Best practices for Azure VM security](https://docs.microsoft.com/en-us/azure/security/azure-security-best-practices-vms)

## Azure networking

The recommendation is to design your Azure Networking using a [Hub-Spoke topology](https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/hybrid-networking/hub-spoke). Consider the HUB like a DMZ deployed with your Virtual network Gateways and other security/edge appliances like Firewalls Etc. while the Spoke will act as the backend zone where your session hosts servers are deployed to and is peered with the HUB.

Below is the architecture diagram that outlines the Azure Networking plan that was deployed for the sake of this migration guide.



WVD Network Architecture

The sections below will briefly summarize the components deployed as a part of the Azure networking plan. ***It is \*highly recommended\* that your networking team is consulted during this phase for an optimal implementation.***

* 1. **Azure Virtual Networks (VNET)**

Like discussed earlier we are going to create 2 VNETs in a HUB and Spoke model using the below details.

* + 1. **HUB** 
       - VNET Name: DC-S2S-Hub01
       - CIDR: 10.2.0.0/16 *or anything else which does not overlap/conflict with any existing networks*
       - Create a subnet called “GatewaySubnet” (this is cannot change and will host the Virtual Network Gateway)
       - Based on your requirements, choose an Azure virtual Network Gateway using the specifications from [Gateway SKU](https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways#gwsku) and deploy it to the “GatewaySubnet”
    2. **Spoke** 
       - VNET Name: DC-S2S-Spoke01
       - CIDR: 10.3.0.0/16 *or anything else which does not overlap/conflict with any existing networks.*
       - Create a subnet called “Default” (or make it specific based on how you want to isolate & manage servers)
    3. **VNET Peering**
* Configure Peering across the HUB & Spoke VNETs so that resources in networks Hub (10.2.0.0/16) & Spoke (10.3.0.0/16) can communicate with each other.
  + 1. **S2S Connectivity between on-premises & Azure.**

**This is ONLY required if you have an on-premises environment that you want to sync/extend into azure or have any service dependencies. Please consult your networking team to understand and implement steps in this section.**

* Based on your bandwidth, latency & security requirements first choose between the connectivity model.
  + S2S or Express Route. *[For the sake of this document, we will be using S2S IPSEC tunnel]*
* Follow the instructions below to build an S2S-IPSEC tunnel using the on-premises edge networking device.
  + Read through the [vpn-gateway, Bandwidth requirements](https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways) to finalize your requirements first
  + [Create the VPN Gateway](https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-tutorial-create-gateway-powershell)
    - *Like shown in the above diagram, an Azure virtual network gateway called “DC-S2S-VNG01” has been deployed to the HUB VNET and a static publicIP address 13.66.218.236 assigned to it.*
  + Use the instructions at [Build an S2S IPSEC tunnel with Azure](https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-tutorial-vpnconnection-powershell) and complete the connectivity to azure.
    - *From the architecture diagram, a connection “DC-S2S-LNG-CONN01” back to the on-premises device (96.76.54.33) has been created in Azure.*
  + Update the VNET with your on-premises DNS servers using the instructions at [Change DNS servers](https://docs.microsoft.com/en-us/azure/virtual-network/manage-virtual-network#change-dns-servers)
    - *For both the HUB & SPOKE VNETs the DNS servers has been updated to 10.1.10.28(on-prem). IF you are planning to deploy additional Domain Controllers in Azure, please remember to add those as well once ready.*
  + Now you should be able to launch a VM in the Spoke VNET > domain join and access it like a local resource.

## Identity & Access Management

Please ensure that the Active Directory Requirements mentioned at [**WVD requirements**](https://docs.microsoft.com/en-us/azure/virtual-desktop/overview#requirements) are completed only after which steps in the below section can be accomplished.

This section will cover a multitude of areas starting for provisioning AD security groups & organizing users, creating GPO objects, extending your Identity into Azure ETC. It is highly recommended to work with your AD team for this section.

* 1. **Create Test Users and AD Security Groups**

For the sake of implementing a WVD PoC, we will be creating some test users’ objects & AD Security groups that can be used to validate WVD functionality without disrupting everyday operations.

1. Let’s start by creating some test users that will later be used to grant access to remote desktops & apps.
2. Log onto the domain controller > open PowerShell and run the below command

#update values first

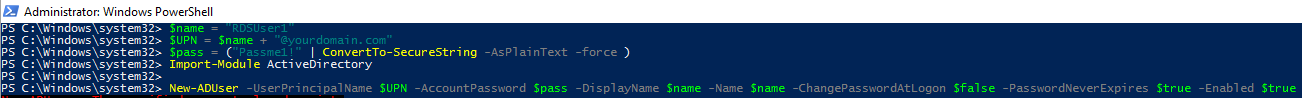
$name = "RDSUser1"

$UPN = $name + "@yourdomain.com"

$pass = ("Passme1!" | ConvertTo-SecureString -AsPlainText -force )

Import-Module ActiveDirectory

New-ADUser -UserPrincipalName $UPN -AccountPassword $pass -DisplayName $name -Name $name -ChangePasswordAtLogon $false -PasswordNeverExpires $true -Enabled $true



Update the values and Repeat the command for as many users you like to test with.

1. Now let’s create the Security group(s) that will be required to manage resources and grant access at different stages in the subsequent sections. Below is a list of the security groups we need and why.

| SecurityGroupName | Description |
| --- | --- |
| AccessFSLogix | Will contains the Session Host computer Objects that need to access the SOFS/S2D cluster to manage user profile containers |
| RDS-RemoteAppUsers | Contains users that need access to RemoteApps hosted using WVD |
| RDS-PooledDesktopUsers | Contains users that need access to RemoteDesktop(Pooled) hosted using WVD |

Execute below commands on the domain controller in PowerShell

#update values using the first

$SecurityGroupName = "value from the SecurityGroupName column"

$Description = ""value from the Description column"

New-ADGroup -Name $SecurityGroupName -SamAccountName $SecurityGroupName -GroupCategory Security -GroupScope Global - -Description $Description



1. Now let’s add the test users to the RemoteApp & PooledDesktop Security Groups. *FYI, I created a total of 4 test users and will be adding 2 users to each group.*

Execute below commands on the domain controller in PowerShell

#update the value and add users to RemoteApp group

$Identity = “RDS-RemoteAppUsers”

Add-ADGroupMember -Identity $Identity -Members @("rdsuser1","rdsuser2")

#adding users to RemoteDesktop group

$Identity = “RDS-PooledDesktopUsers”

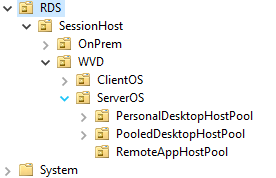
Add-ADGroupMember -Identity $Identity -Members @("rdsuser3","rdsuser4")

* 1. **Active Directory Organization Unit (OU) structure for WVD session hosts.**

It is strongly recommended to consult your in-house GPO expert for this section. The below guidance is subjective, and every enterprise should already have an established process/guidelines to manage their AD computer objects. Consider the below information as a mere FYI to help understand the steps to setup an OU structure.

Since we are introducing new servers into the existing environment and would most likely manage them using GPOs (Group Policy Objects), it is important to plan for the same. Settings like RDS licensing would already be managed using GPOs and since FSlogix for profile management is being introduced, the below guidance was used to organize WVD session hosts into a specific OU structure where FSLogix settings can be centrally controlled for the WVD sessions hosts across the different HostPools.

1. *On your domain controller, open ADUC (dsa.msc)*
2. *Expand the domain and get to the RDS OU (consider this the main OU where all your on-prem RDS computer objects are stored)*
   * *Under RDS, create a sub OU called Session Host (or Likewise) to manage common settings for all session hosts (on-prem & WVD)*
   * *Under Session Host, create a sub OU called WVD (or Likewise) to manage the WVD session hosts*
   * *Under WVD, create a sub OU called “RemoteAppHostPool” (the idea is to store all session hosts that server remote apps relative to the purpose of your HostPool in WVD)*



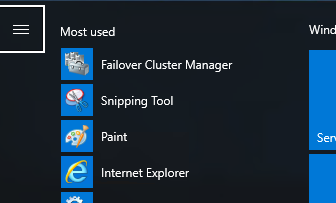
* + *Once the WVD session hosts are provisioned in Azure at a later section, steps are provided to move servers into the respective OUs.*

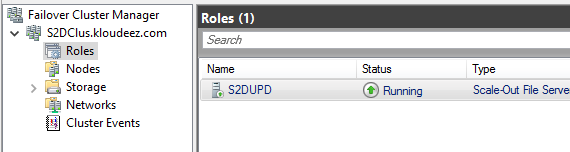
## Scale Out File Server (SOFS) with Storage Spaces Direct (S2D)

If you already have (or planning) to use either roaming profiles or FSlogix solutions for your users` profile data in Azure, then a Scale out File server (SOFS) with Storage Spaces Direct (S2D) is the recommended storage solution in Azure to host those user profiles.

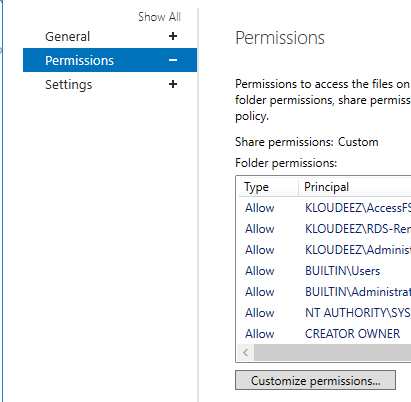
NOTE: The Windows Virtual Desktop (WVD) service offers FSLogix containers as the recommended user profile solution. The user profile disk (UPD) solution is not recommended and will be deprecated in future versions of Windows Virtual Desktop.

1. Based on the total # of users and their profile size requirements, first plan for the SOFS cluster size and SKU requirements in Azure using these [guidelines](https://docs.microsoft.com/en-us/windows-server/remote/remote-desktop-services/rds-storage-spaces-direct-deployment)
2. Deploy the SOFS cluster either manually or using ARM templates
   1. [Manual Deployment](https://docs.microsoft.com/en-us/windows-server/storage/storage-spaces/deploy-storage-spaces-direct)
   2. [ARM Template](https://azure.microsoft.com/en-us/resources/templates/301-storage-spaces-direct/)
3. *For the sake of this guide, the cluster details are as follows:*
   1. *Cluster Name: S2DCLUS.Kloudeez.com*
   2. *SOFS/S2D Name : S2DUPD.Kloudeez.com*
4. After the CSV file shares are created to host user profile data, the correct NTFS and Share permissions must be applied **on each share** for data security & integrity using the steps below.
   1. Logon to any of the file server nodes and click Start > Failover cluster manager > expand cluster > Click roles > Click S2DUPD Role





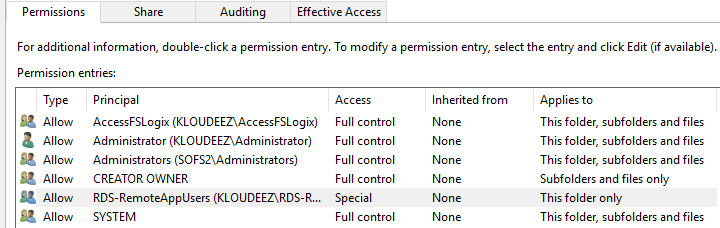
* 1. Now click Shares at the bottom > Right click the Share (Ex: RemoteApps) > click properties
  2. In the new window > click permissions > customize permissions



* 1. Now let’s set the NTFS Permissions. In the new window, ensure you are under the Permissions tab > click Add
  2. Click Select a principal > Select the respective AD object we want to set permissions > click ok > Set Type = Allow > Applies To = value under Folder in the table below > Click Show advanced permissions and select respective values from Permissions column below

|  |  |  |  |
| --- | --- | --- | --- |
| **User Account** | **Description** | **Folder** | **Permissions** |
| CREATOR OWNER | CREATOR OWNER | Subfolders and Files Only | Full Control |
| SYSTEM | SYSTEM | This Folder, Subfolders and Files | Full Control |
| Domain Administrators | Your Domain Administrator AD Security Group | This Folder, Subfolders and Files | Full Control |
| File cluster Administrators | The local File cluster Administrator | This Folder, Subfolders and Files | Full Control |
| Domain\AccessFSLogix | AD Security group containing Session Host computer objects that can access/control these shares to store ser profile data | This Folder, Subfolders and Files | Full Control |
| Domain\RDS-RemoteAppUsers | The AD security group containing users that use RemoteApps | This Folder, Subfolders and Files | Create Folder/Write Data  List Folder/Read Data  Read Attributes Traverse Folder/Execute File |

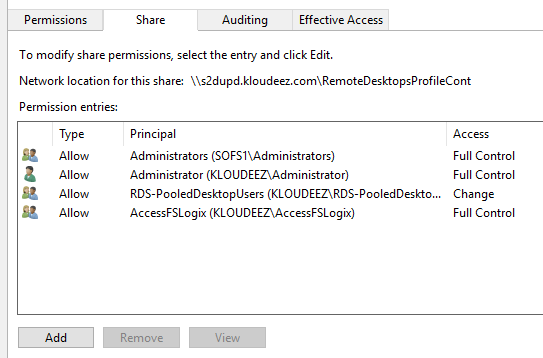
* 1. Repeat step 5 for all other objects (in the above table) you need to set permissions for
  2. Once done, your NTFS permissions window should look relative to below. Now click Apply



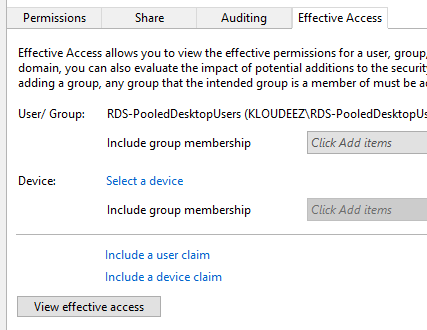
* 1. Now we will set Share permissions. Click on Share at the top > click Add
  2. Click Select a principal > Select the respective AD object we want to set permissions > click ok > Set Type = Allow > Permissions = value From Permissions column in the table below

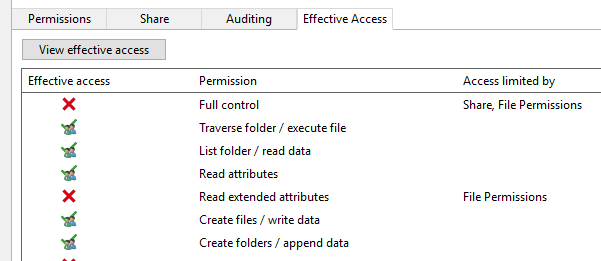
|  |  |  |
| --- | --- | --- |
| **User Account** | **Description** | **Permissions** |
| Domain Administrators | Your Domain Administrator AD Security Group | Full Control |
| File cluster Administrators | The local File cluster Administrator | Full Control |
| Domain\AccessFSLogix | AD Security group containing Session Host computer objects that can access/control these shares to store ser profile data | Full Control |
| Domain\RDS-RemoteAppUsers | The AD security group containing users that use RemoteApps | Change |

* 1. Once done, your Share permissions window should look relative to below. Now click Apply

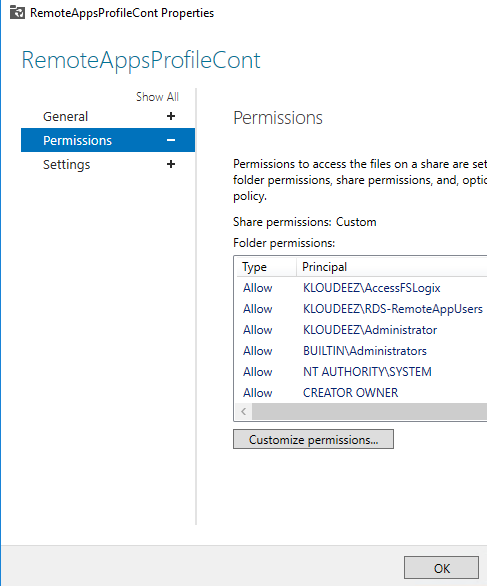


* 1. Validate the required users have access by doing the following. Click Effective access at the top > click Select User, choose respective Security Group OR user, click ok > click effective access > scroll down and ensure the minimum access to list/read/write files & folders is present.





* 1. Click OK > again OK in the Properties window to save your changes

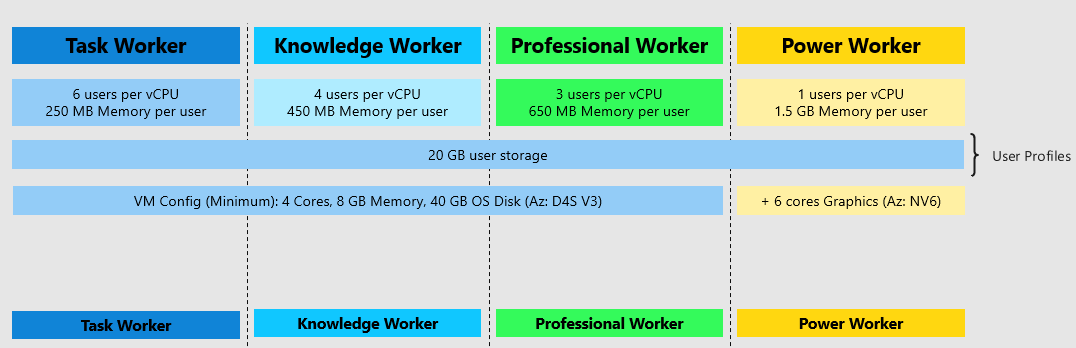


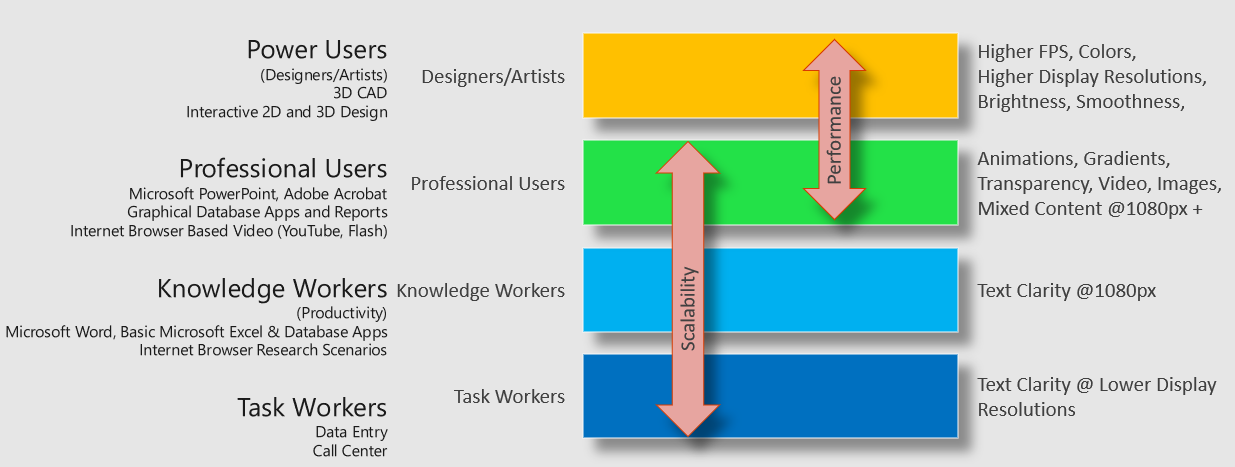
* 1. Now, repeat step 3 for all the other shares you wish to use for storing user profile data in Azure.

## WVD Session Host VM & Storage SKU Guidance

Based on your end goals & requirements, the planning & selection of the WVD session host VM SKUs can be done in a couple of different ways.

1. Using the recommendations in the chart(s) below, split your current users into different WVD personas based on their workload requirements.





* For example, if there are 100 users of each WVD persona. At a minimum, you would need the below VM requirements for a Host Pool with at least 2 session hosts

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| WVD User Persona | Min vCPU per server | Min Memory (GB) per server | Min SMB Storage endpoint (TB) per server | Notes |
| Task Worker | 8 (~6 users per vCPU) | 16 (~250 MB per user) | 2 (~ 20GB per user) | Ideally, let’s assume each server will have a max of 50 user at any given time. Although, if one of the session hosts goes down (maintenance Etc.), there should be enough capacity on the other server to accommodate additional users. *Apply a relative model when planning for more than 2 session hosts in your hostpool (scale-out)*  Min RAM is rounded off to the next highest available config. [EX: Knowledge users need min of 45GB of RAM, so it is rounded off to higher config of 64GB and not then lower config of 32GB] |
| Knowledge Worker | 24 (~4 users per vCPU) | 64 (~450 MB per user) | 2 (~ 20GB per user) |
| Professional Users | 32(~3 users per vCPU) | 64 (~650 MB per user) | 2 (~ 20GB per user) |
| Power Users *[GPU enabled VMs]* | 100 (~1 user per vCPU) | 1600 (~1.6 GB per user) | 2 (~ 20GB per user) |

3 WVD Session Host guidance

1. Depending on the customer’s flexibility, they can also choose a SKU that better suits their needs. Such as a Compute Optimized vs Memory Optimized vs High Performance Compute etc.

# WVD PoC 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.



## 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**.
3. 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).
4. Select **Accept** > wait for one minute.
5. Navigate back to the [Windows Virtual Desktop consent page.](https://rdweb.wvd.microsoft.com/)
6. Go to **Consent Option** > **Client App**, enter the same Azure AD tenant name or Directory ID, then select **Submit**.
7. Sign into the Windows Virtual Desktop consent page as global administrator like you did back in step 3. Select **Accept**.

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

1. Open a browser and connect to the [Azure Portal](https://portal.azure.com) with your global administrator account.
   1. If you're working with multiple Azure AD tenants, it's best practice to open a private browser session and copy and paste URLs into the address.
2. Select Enterprise applications, search for Windows Virtual Desktop and select the application.
3. Select Users and groups, then select Add user. *Ensure that this is either a service or a user account that does not have MFA/CA enabled*
4. Select Users and groups in the Add Assignment blade
5. Search for a user account that will create your Windows Virtual Desktop tenant.
   1. For simplicity, this can be the global administrator account.
6. Select the user account, click the Select button > now click Assign at the bottom

## Download the WVD PowerShell Module

1. Download the Windows Virtual Desktop module and save the package in a known location on your computer.
2. Find the downloaded package. Right-click the zip file, select Properties, select Unblock, then select OK. This will allow your system to trust the module.
3. Right-click the zip file, select Extract all..., choose a file location, then select Extract.
4. First, run this cmdlet to save the file location of the extracted .zip file into a variable:

$module = "<extracted-module-location>"

1. Second, run this cmdlet to import the DLL for the module:

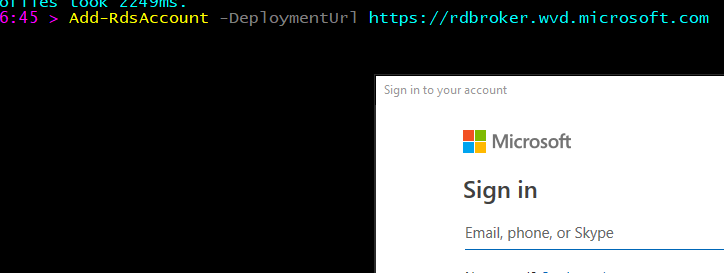
Import-Module $module\Microsoft.RDInfra.RDPowershell.dll

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.

## Create the WVD Tenant

1. In the PowerShell session, login as a Tenant Creator using the command below

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



1. Now run the below commands to create a new WVD Tenant

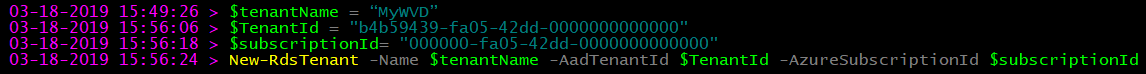
**#Setting Variables. Update the demo values within “ “ based on your specifics**

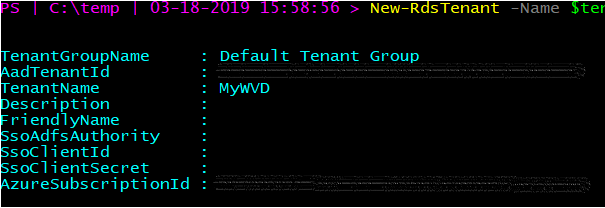
$tenantName = “MyWVD”

$TenantId = "0000000-0000-0000-0000000000000"

$subscriptionId= "0000000-0000-0000-0000000000000"

New-RdsTenant -Name $tenantName -AadTenantId $TenantId -AzureSubscriptionId $subscriptionId





## Deploy the WVD Host Pools

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.

Follow the steps in this article to create a host pool within a Windows Virtual Desktop tenant. This includes creating a host pool in Windows Virtual Desktop, creating a resource group with VMs in an Azure subscription, joining those VMs to the Active Directory domain, and registering the VMs with Windows Virtual Desktop.

Very important to note that at this time, WVD only supports the OS versions listed at [Requirements](https://docs.microsoft.com/en-us/azure/virtual-desktop/overview#requirements)



### Deploy VMs with Client Operating System

* 1. **Windows 10 Multi-Session**

Choose any deployment style from below based on your preference (GUI Vs command line)

* + 1. For deploying using the Azure portal (GUI) refer the section [Deploy HostPool using Azure Portal (Marketplace)](#_Deploy_HostPool_using_1)
    2. For deploying using the ARM template (GitHub) refer section [Deploy HostPool using ARM template](#_Deploy_HostPool_using_2)

### Deploy VMs with Server Operating System

* 1. **Windows Server 2016**

Choose any deployment style from below based on your preference (GUI Vs command line)

* + 1. For deploying using the Azure portal (GUI) refer the section [Deploy HostPool using Azure Portal (Marketplace)](#_Deploy_HostPool_using_1)
    2. For deploying using the ARM template (GitHub) refer section [Deploy HostPool using ARM template](#_Deploy_HostPool_using_2)
  1. **Windows Server 2012 R2 and Windows Server 2019**

Currently, the Azure Portal (GUI) or existing ARM templates (GitHub) do not support deploying these Operating Systems. So, to deploy VMs with this OS refer steps from [Deploy HostPool using modified ARM template](#_Deploy_HostPool_using_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.

#UPDATE THESE VALUES FIRST

$module = "C:\temp\RDPowershell"

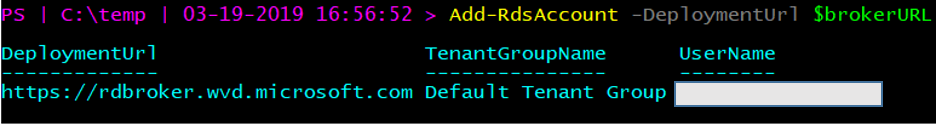
$TenantGroupName = “Default Tenant Group”

$brokerURL= “<https://rdbroker.wvd.microsoft.com>”

Import-Module $module\Microsoft.RDInfra.RDPowershell.dll

Add-RdsAccount -DeploymentUrl $brokerURL

Set-RdsContext -TenantGroupName $TenantGroupName

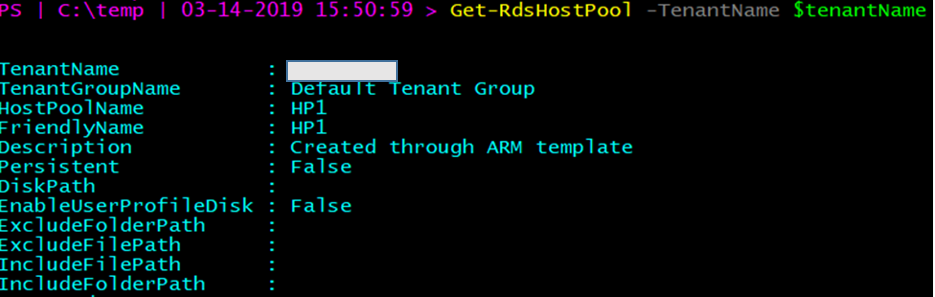


1. Check for the new Host Pool using below command

#UPDATE THESE VALUES FIRST

$TenantName = “MyWVD”

Get-RdsHostPool -TenantName $tenantName

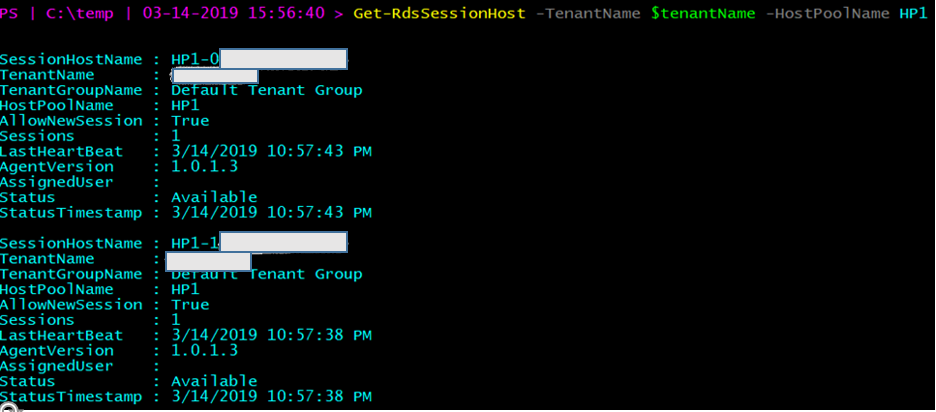


1. Check for the Session hosts in the Host Pool and ensure the status is **Available**

#UPDATE THESE VALUES FIRST

$HostPoolName = “HP1”

Get-RdsSessionHost -TenantName $tenantName -HostPoolName $HostPoolName

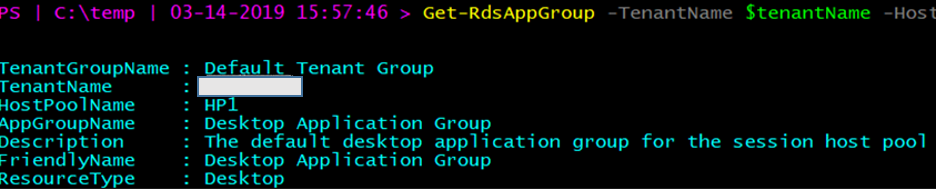


## Manage App Groups

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

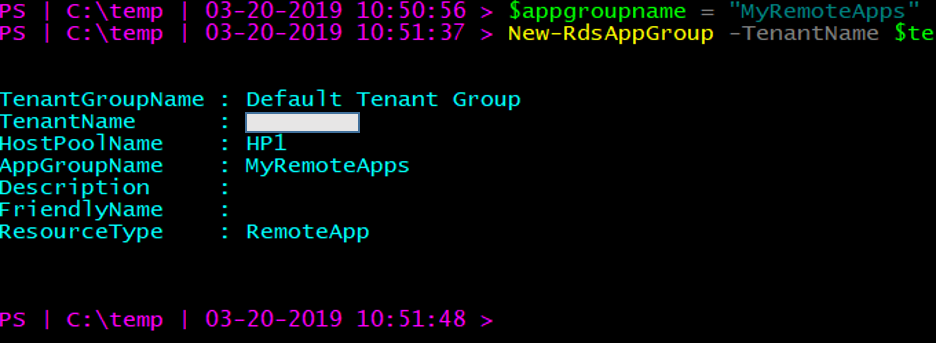


1. Now run the following PowerShell cmdlet to create a new empty RemoteApp group

#UPDATE THESE VALUES FIRST

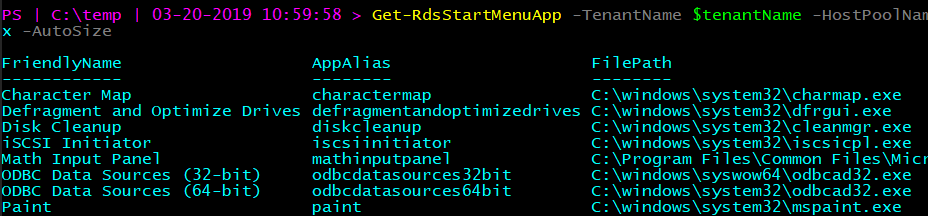
$appgroupname = “MyRemoteApps”

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.

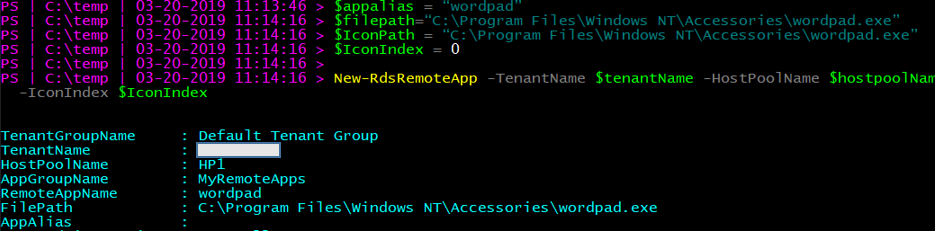
$name = “wordpad”

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

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

$IconIndex = 0

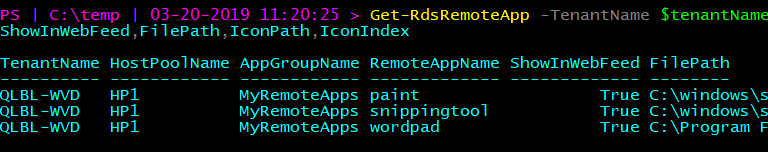
New-RdsRemoteApp -TenantName $tenantName -HostPoolName $hostpoolName -appgroupname $appgroupname -Name $name -Filepath $filepath -IconPath $IconPath -IconIndex $IconIndex



Now, update the variables and repeat the above commands for any other applications you want to publish. As an example, we are publishing 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”

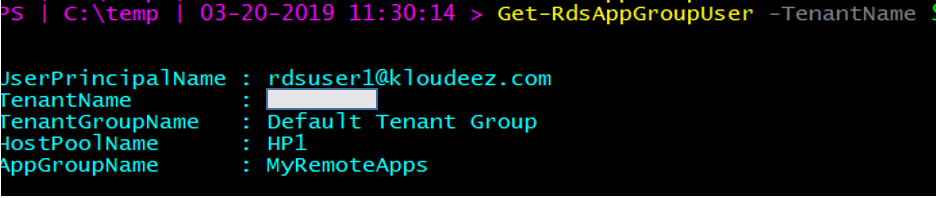
$upn = “[rdsuser1@domain.com](mailto:rdsuser1@domain.com)” #this should be the user that will access WVD resources from your domain

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



## O365 Optimization using FSLogix

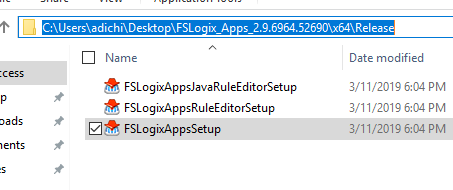
The Windows Virtual Desktop service offers FSLogix containers as the recommended user profile solution. The traditional windows user profile disk (UPD) solution is not recommended and will be deprecated in future versions of Windows Virtual Desktop.

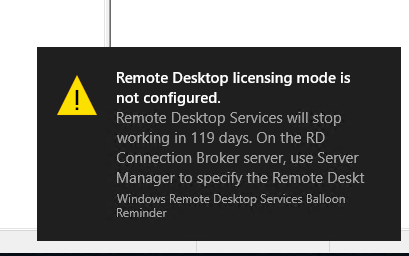
For any reason, if you have NOT already completed provisioning a SOFS/S2D cluster, please complete that [section](#_Scale_Out_File) first and continue from here.



### Install FSLogix on Session Hosts

1. Use [this link](https://go.microsoft.com/fwlink/?linkid=2084562) to download FSLogix.
2. From your WVD Host pool in Azure, login to one of the session hosts using an administrator account and copy the FSLogix bits locally. *Please ignore the RDLicensing warning at this time.*



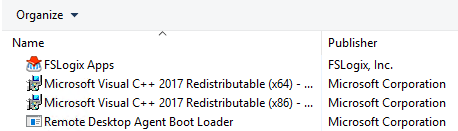


1. CD to the path where you copied > open PowerShell > run the below command ***AND replace the Product key with the licensed one you obtained either directly from FSLogix or Microsoft***

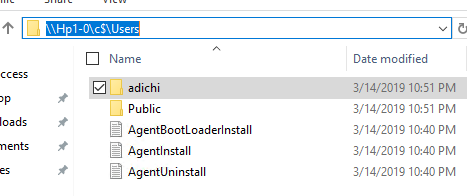
.\fslogixappssetup.exe /install /ProductKey=TRIAL-G6KID-WKRLO-JAOIA-OXARY /quiet

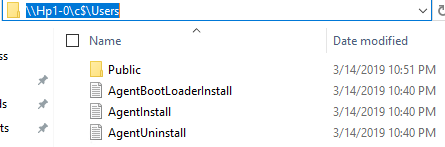


1. After couple of minutes > goto control Panel and see that FSLogix is now installed.



1. **Logoff (not disconnect/close session)** from this Session host
2. Now repeat steps 2-5 for the other Session Host(s) in the HostPool.
3. RDP to one of the server nodes of the SOFS cluster (EX: SOFS1) > open Windows explorer > and goto path [\\SessionHost\C$\Users](file://SessionHost/C$/Users) > delete the Admin profile that you just logged with after which you should only see the Public folder and the Agent\* text files



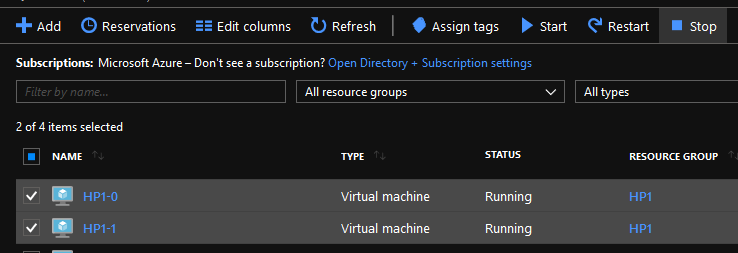


1. Repeat Step 7 for ALL other session hosts in your host pool.

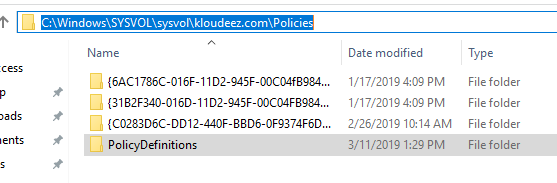
### Configure FSLogix GPO Settings

This section will involve making changes to the AD infrastructure along with the Group Policy Objects so please consult/have an AD expert/administrator team present while executing.

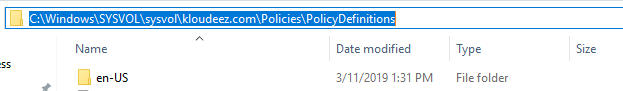
1. Login to the [Azure Portal](https://portal.azure.com) > goto Virtual Machines and STOP the session hosts



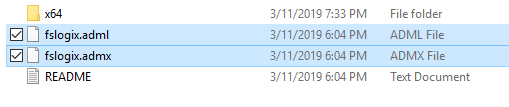
1. Now RDP to your domain controller > goto C:\Windows\SYSVOL\sysvol\<<YourDomain>>\Policies and create a folder called **PolicyDefinitions**

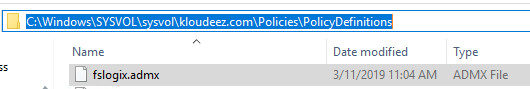


1. Go into the above folder and create another folder called “en-US”

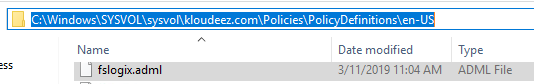
s

1. From the FSlogix installation folder in the section [Install FSLogix on Session Hosts](#_Install_FSLogix_on) copy the **fslogix.ADMX** file to C:\Windows\SYSVOL\sysvol\<<YourDomain>>\Policies\**PoliciesDefinition** folder on the domain controller

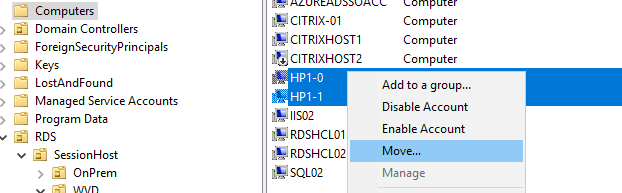


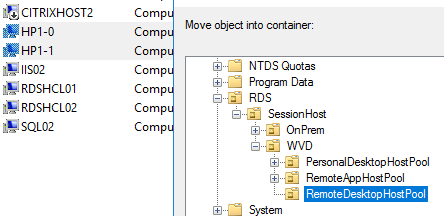


1. Similarly, copy the **fslogix.ADML** file to C:\Windows\SYSVOL\sysvol\<<YourDomain>>\Policies\**PoliciesDefinition\en-US** folder on the domain controller

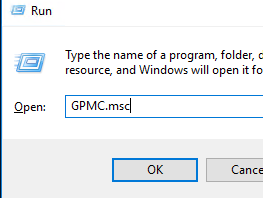


1. Open Active directory users & computers (dsa.msc)
2. Goto the Container/OU where your session hosts are present > CTRL + select the WVD session hosts > right click > move > and move them to the Respective WVD OU > Click OK

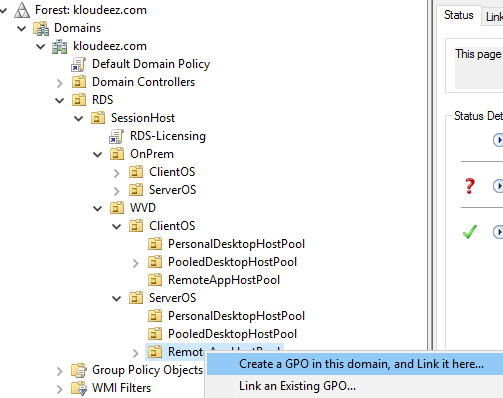




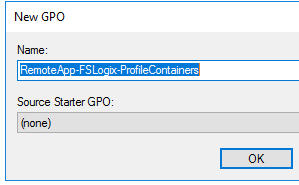
1. Hold Windows key + R > type GPMC.msc to open the group policy management console



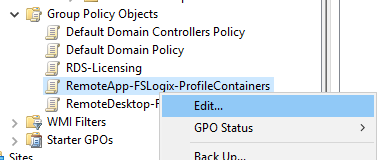
1. Expand the domain and get to the OU where your session hosts exist to create and link a new GPO that will deploy FSlogix container settings



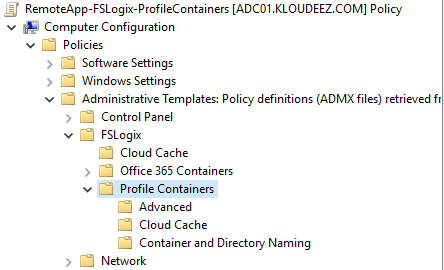
1. Provide a meaningful name for the GPO > OK



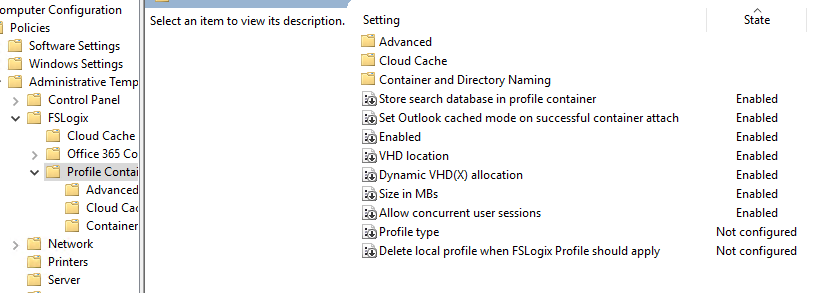
1. Expand the OU to see the new GPO > right click > Edit



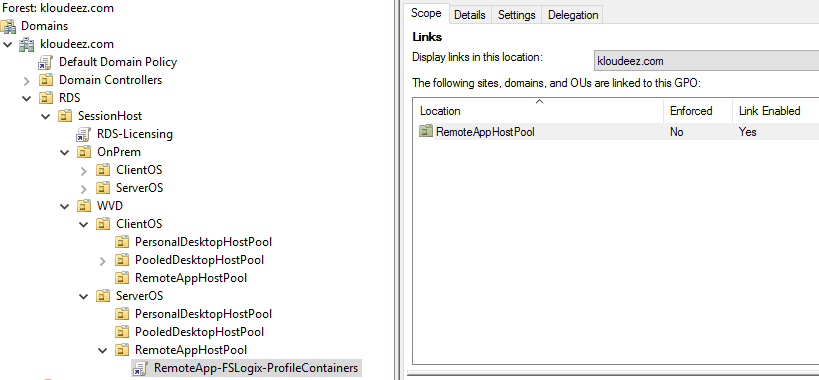
1. This should open the GPO editor
2. Expand Computer Configuration > Policies > Administrative Templates > FSLogix > Profile Containers



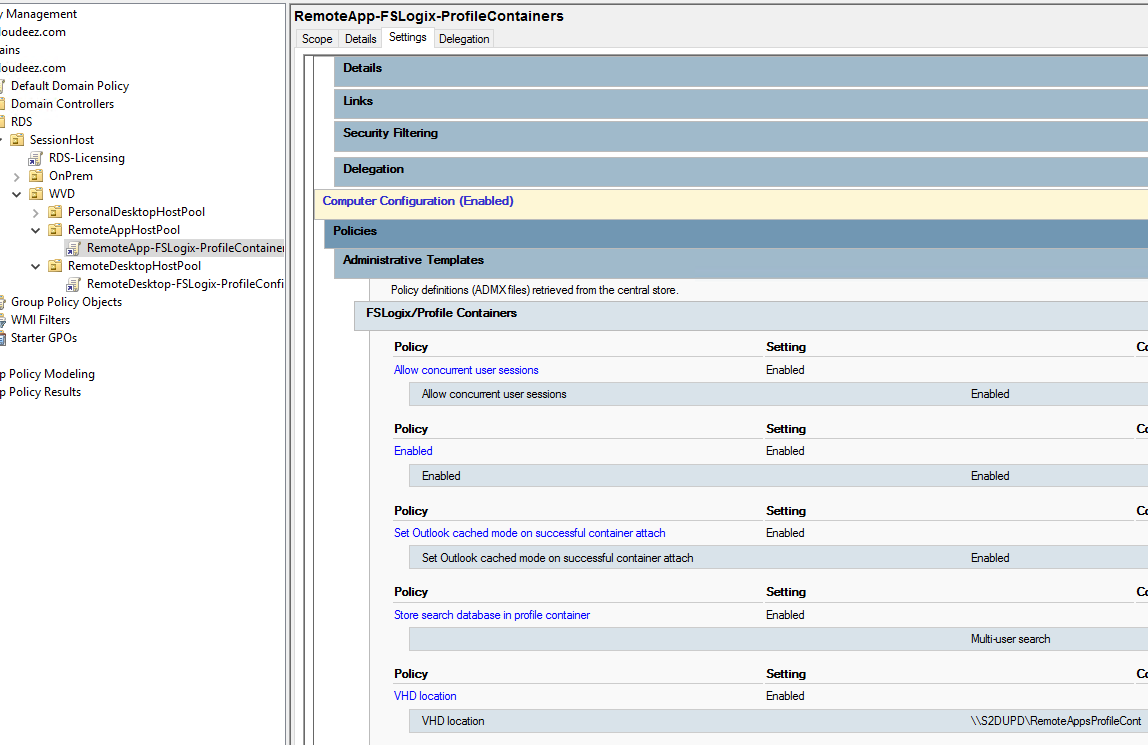
1. Now, using the list below, double click on the respective setting and Enable it so that the session hosts with FSLogix installed will have the same settings updated
   * **Core Settings (These MUST be enabled for FSLogix to function)**
     + Enabled
     + Size in MBs
       - Provide Size in MBs for each user profile (Ex: 10000MB / 10GB)
     + VHD location
       - Use the respective S2D share path you created in section [Scale Out File Server (SOFS) with Storage Spaces Direct (S2D)](#_Scale_Out_File) (EX: [\\S2DUPD\RemoteAppsProfileCont](file://S2DUPD/RemoteAppsProfileCont))
   * *Optional Settings (Consult your user profile expert as these are subject to your requirements. For this document we are enabling them)*
     + Allow concurrent user sessions
     + Dynamic VHD(X) allocation
     + Set Outlook cached mode on successful container attach



1. Now close the GPO editor and get back to GPMC
2. Select the GPO > on the right under scope ensure that Link Enabled = Yes

ss

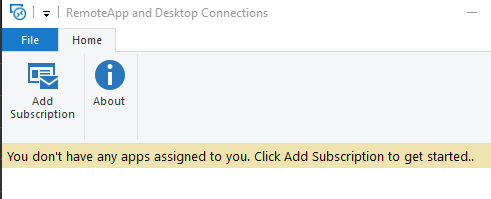
1. Now click on Details > under Computer Configuration > expand Policies. Administrative templates > FSLogix/Profile Containers to see all the settings you have applied



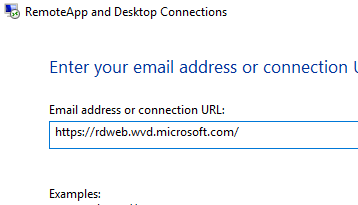
## End User Experience

At this stage, your RemoteApps are deployed on the WVD session hosts along with the FSLogix configurations in place 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.

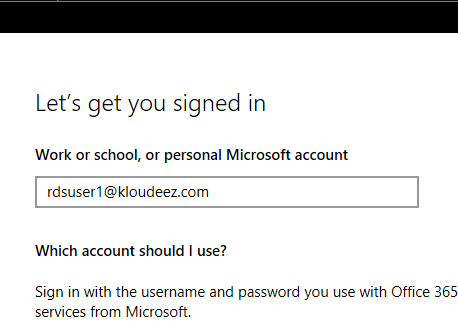
1. [Download the client](https://go.microsoft.com/fwlink/?linkid=2068602) and run the MSI to complete the installation.
2. Start the client from the All Apps List, look for Remote Desktop.

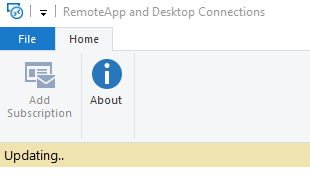


1. Click Add Subscription > provide URL = <https://rdweb.wvd.microsoft.com/> > Next > Next Again

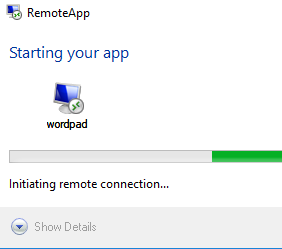


1. Sign in with you’re the user account that was granted access to the WVD-RemoteApps in the earlier section > Next





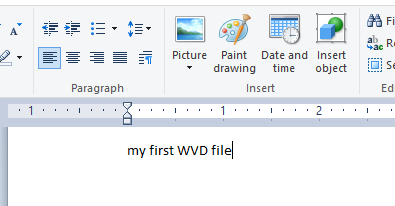
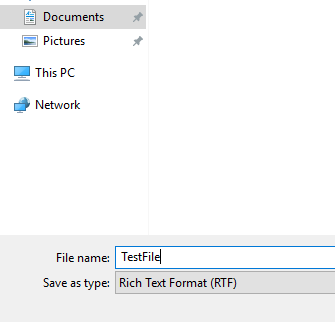
1. After successfully authenticating, you should now see a list of resources available to you.
2. Please launch any of the resources (EX: Wordpad). *please be advised that the first launch may be slow as your user profile is being created.*



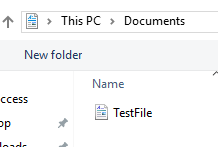
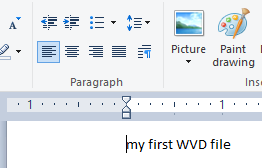
1. Once launched, you can see the icon in your taskbar



1. Now type something > save your file > close WordPad

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

1. If you are trying to launch O365 applications and hit errors for any reason, please refer to ***WVD-FAQ.docx***
2. 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

* + Going to <https://rdweb.wvd.microsoft.com>
  + Login with user domain credentials
  + Access Apps & Desktops

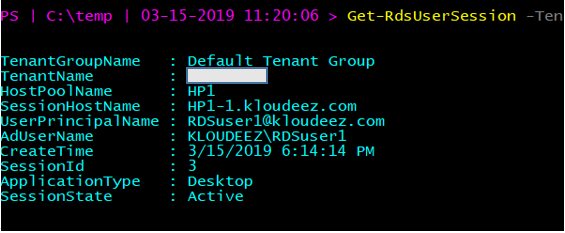
1. As an Admin, you can also validate the User Session data from the WVD end using either of the commands.

#for all AppGroups in a HostPool

Get-RdsUserSession -TenantName $tenantName

#Filter to a specific for all AppGroup in a HostPool

Get-RdsUserSession -TenantName $tenantName -HostPoolName $hostpoolName -Verbose

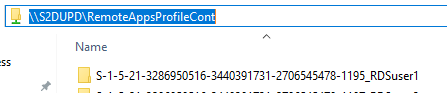




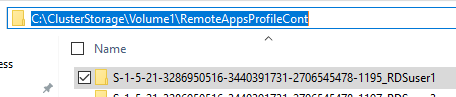
### Validate FSlogix Profile containers

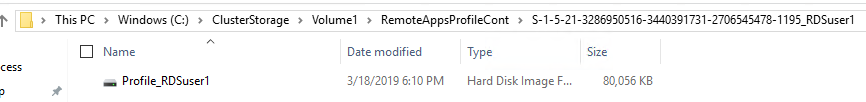
Since we had FSLogix configured on the session hosts for user profile management, all user profiles are saved as FSlogix Profile Containers (VHDx). This section will provide the steps to validate if and where the FSLogix Profile containers are being created.

1. From the Domain controller > open windows explorer
2. In the address bar type the S2D cluster share path [EX: \\S2DUPD\RemoteAppsProfileCont] to see the Profile Container for RDSUSER1



1. Alternatively, you can also RDP to one of the SOFS/S2D cluster nodes and goto the CSV volume [in this case C:\ClusterStorage\Volume1\RemoteAppsProfileContainer] to see the profile container (VHDx) for RDSUser1







# WVD Management

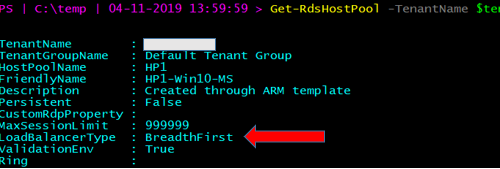
## Configure the Load balancing Method

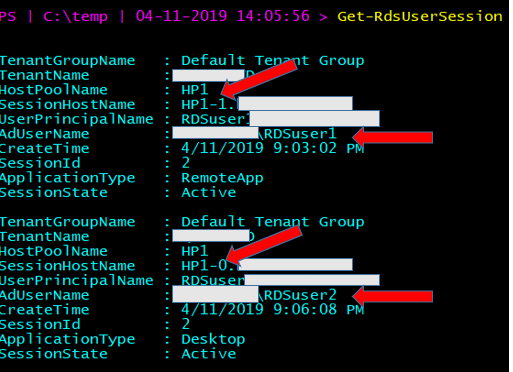
Explanation about the different LB methods is found at [Configure the Windows Virtual Desktop load-balancing method](https://docs.microsoft.com/en-us/azure/virtual-desktop/configure-host-pool-load-balancing) . Below are a couple of screenshots that confirm how the session allocation across the session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



### Breadth First

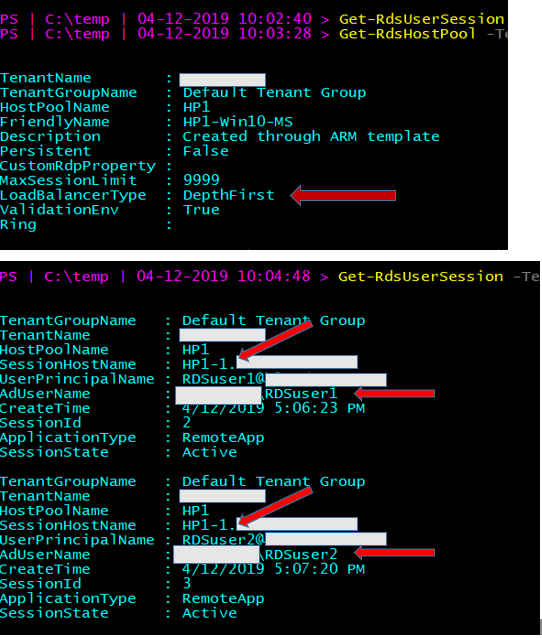
Session Allocation using Breadth-First for a HostPool with 2 VM’s. RDSUser1 & RDSUser2 are scattered across the VM’s for better using experience.





### 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.

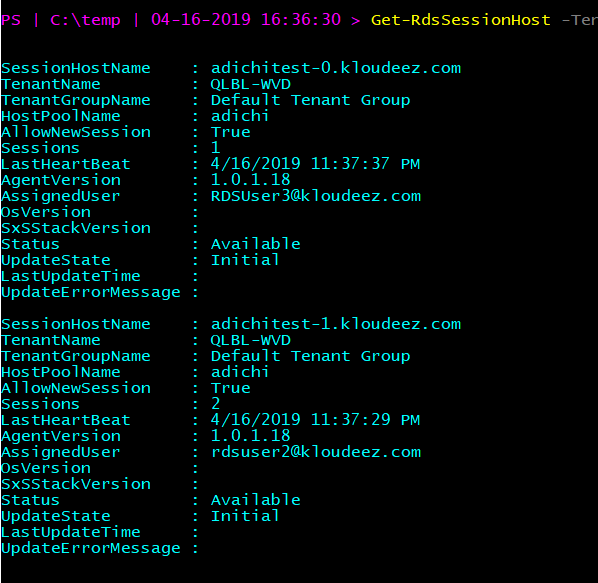


### Persistent Desktops

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.

PS I 
C: \temp 
Get—RdsHos tpool 
QLBL -WVD 
Default Tenant Group 
adi chi 
adi chi 
. Created through ARM template 
True 
. 999999 
persistent 
False 
-Ter 
enantName 
enantGroupName 
Hos tpool Name 
F ri endlyName 
Descri pti on 
persistent 
us tomRdpp roper ty 
axsessi onL imi t 
LoadBa1 ancerType 
al i dati onEnv 
Ri ng 

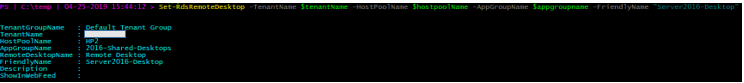


## Set Friendly Names for Published Desktops

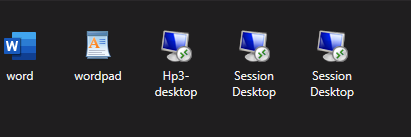
Using the below command, you can set friendly names to uniquely identify multiple desktops published to a user.

#Update the respective $variables first and then execute command

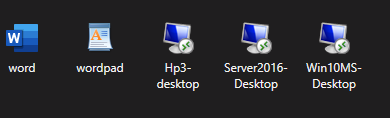
Set-RdsRemoteDesktop -TenantName $tenantName -HostPoolName $ HostPoolName -AppGroupName $AppGroupName -FriendlyName "My custom Desktop"



***Before***



***After***



## 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. Go to [https://github.com/Azure/RDS-Templates/tree/master/wvd-templates/wvd-management-ux/deploy](https://nam06.safelinks.protection.outlook.com/?url=https%3A%2F%2Fgithub.com%2FAzure%2FRDS-Templates%2Ftree%2Fmaster%2Fwvd-templates%2Fwvd-management-ux%2Fdeploy&data=01%7C01%7Cv-adichi%40microsoft.com%7C13d88aebfe2c46c291e708d6d401b700%7C72f988bf86f141af91ab2d7cd011db47%7C1&sdata=61Es2FnRYlJfL11raCzzDUBi8U9T8q4g3b8ygC0Mf8s%3D&reserved=0)
2. Click on Deploy to Azure.
3. ON the custom deployment page, provide the following info on the service:
   * RD Broker URL: [https://rdbroker.wvd.microsoft.com/](https://nam06.safelinks.protection.outlook.com/?url=https%3A%2F%2Frdbroker.wvd.microsoft.com%2F&data=01%7C01%7Cv-adichi%40microsoft.com%7C13d88aebfe2c46c291e708d6d401b700%7C72f988bf86f141af91ab2d7cd011db47%7C1&sdata=T7MzzwfuLH8MZE4qRr%2BKp1xL3ntFpQffcxEvO2GuaLc%3D&reserved=0)
   * Resource URL: [https://mrs-prod.ame.gbl/mrs-RDInfra-prod](https://nam06.safelinks.protection.outlook.com/?url=https%3A%2F%2Fmrs-prod.ame.gbl%2Fmrs-RDInfra-prod&data=01%7C01%7Cv-adichi%40microsoft.com%7C13d88aebfe2c46c291e708d6d401b700%7C72f988bf86f141af91ab2d7cd011db47%7C1&sdata=vIb9mpg8CZN5dv0EOdEp6SMxtLUW%2BvwHX996nHCAY%2F8%3D&reserved=0)
   * 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. Apr3UX)

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 appservice with name you provided in the template (e.g. Apr3UX) and navigate to the URL that shows up on the top right side

**NOTE: This works only for admins that have access to Default Tenant Group since we have a fix that needs to be done on the service side for other TGs to work.**

## Check Diagnostic data

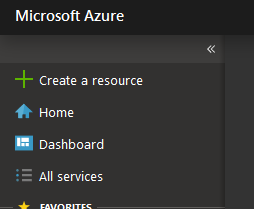
Please follow instructions at [Identify issues with the diagnostics feature](https://docs.microsoft.com/en-us/azure/virtual-desktop/diagnostics-role-service)

# Appendix

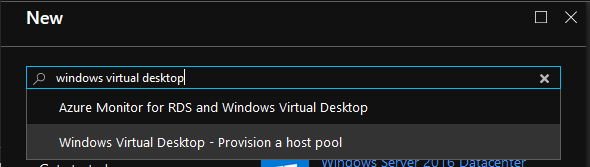


## Deploy HostPool using Azure Portal (Marketplace)

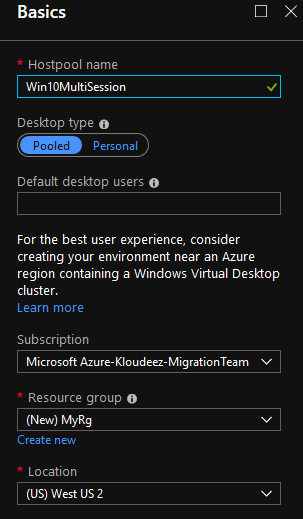
1. Login into the Azure portal
2. Select + or + Create a resource.



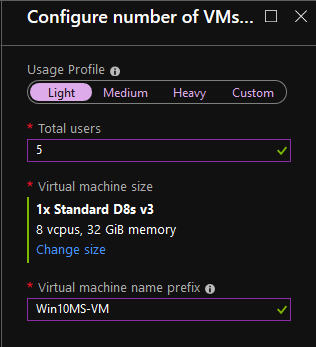
1. Enter Windows Virtual Desktop in the Marketplace search window.
2. Select Windows Virtual Desktop - Provision a host pool, then select Create



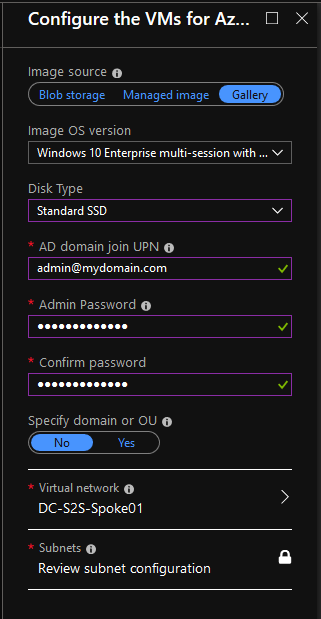
1. For Basic Settings, update as required
   * + Enter a name for the host pool that’s unique within the Windows Virtual Desktop tenant.
     + Select the appropriate option for personal desktop. If you select Yes, each user that connects to this host pool will be permanently assigned to a virtual machine.
     + (optional as this can be done later) Enter a comma-separated list of users who can sign in to the Windows Virtual Desktop clients and access a desktop after the Azure Marketplace offering completes. For example, if you'd like to assign user1@contoso.com and user2@contoso.com access, enter "user1@contoso.com,user2@contoso.com."
     + Select Create new and provide a name for the new resource group
     + For Location, select the same location as the virtual network that has connectivity to the Active Directory server.
     + Select OK.



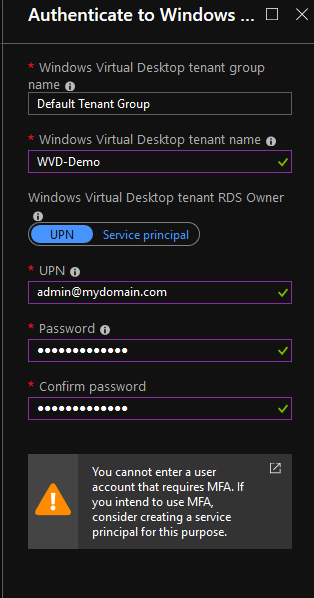
1. For Usage Profile & VM Count, update as required
   * + Choose a usage profile and Provide the total # of users
     + You can change the VM size if required
     + Enter a prefix for the names of the virtual machines. For example, if you enter the name "prefix," the virtual machines will be called "prefix-0," "prefix-1," and so on.
     + Select ok



1. For VM Configuration, do the following
   * + Select the Image source and enter the appropriate information on how to find and use it.
       - **Gallery** – Deploy using the approved images readily available from the gallery
       - **Managed Image** – Deploy an existing Azure Image (with your custom applications and configurations saved)
       - **Blob Storage** – If you choose not to use managed disks, select the storage account containing the .vhd file.
     + Enter the user principal name and password for the domain account that will join the VMs to the Active Directory domain. This same username and password will be created on the virtual machines as a local account. You can reset these local accounts later.
     + Select the virtual network that has connectivity to the Active Directory server, then choose a subnet to host the virtual machines.
     + Select OK.



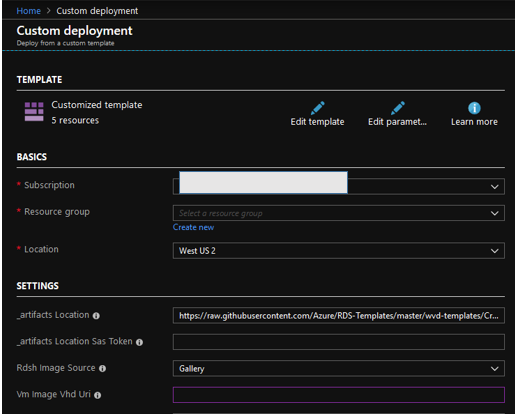
1. For the Windows Virtual Desktop tenant information blade
   * Enter the Windows Virtual Desktop tenant group name for the tenant group that contains your tenant. Leave it as the default unless you were provided a specific tenant group name.
   * Enter the Windows Virtual Desktop tenant name for the tenant you'll be creating this host pool in.
   * Specify the type of credentials you want to use to authenticate as the Windows Virtual Desktop tenant RDS Owner. If you completed the [Create service principals and role assignments with PowerShell tutorial](https://docs.microsoft.com/en-us/azure/virtual-desktop/create-service-principal-role-powershell), select Service principal. You will now need to enter the Azure AD tenant ID of the Azure Active Directory that contains the service principal.
   * Enter either the credentials for the tenant admin account. Only service principals with a password credential are supported.
   * Select OK.



1. In the Summary blade, review the setup information. If you need to change something, go back to the appropriate blade and make your change before continuing. If the information looks right, select OK.
2. In the Buy blade, review the additional information about your purchase from Azure Marketplace.
3. Select Create to deploy your host pool
4. Follow the deployment progress under notifications and if you get any errors, please refer [Tenant and host pool creation](https://docs.microsoft.com/en-us/azure/virtual-desktop/troubleshoot-set-up-issues)

## Deploy HostPool using ARM template

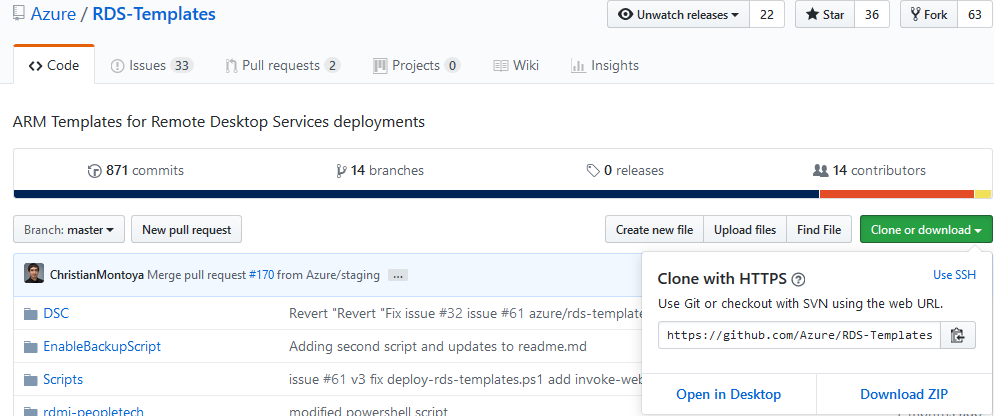
1. Goto [here](https://github.com/Azure/RDS-Templates/tree/master/wvd-templates/Create%20and%20provision%20WVD%20host%20pool) > scroll to the bottom and click on Deploy to Azure
2. If required, ensure you authenticate / login to azure using the correct credentials to land on the custom deployment page



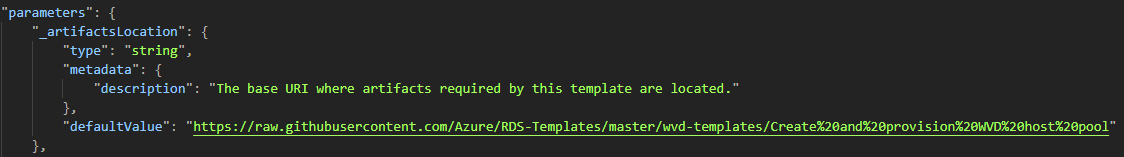
1. On the custom deployment page, complete all the required fields (which are mostly self-explanatory). If there are default values in any of the fields, leave them for the most part unless you know if they need to be updated.
2. Once all required fields are completed > accept terms & conditions & click purchase
3. Wait for the deployment to complete and if there are any errors refer [**here**](https://docs.microsoft.com/en-us/azure/virtual-desktop/troubleshoot-set-up-overview)

## Deploy HostPool using modified ARM template

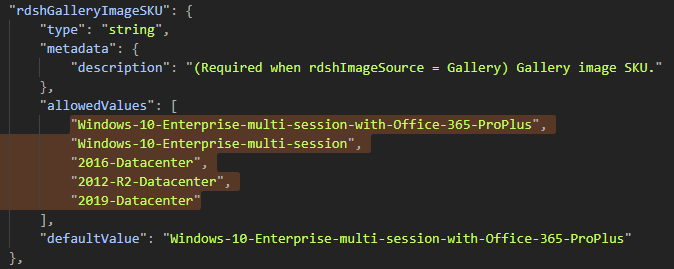
* + 1. You will need a GitHub account for this.
    2. Goto [here](https://github.com/Azure/RDS-Templates/tree/master/wvd-templates/Create%20and%20provision%20WVD%20host%20pool) > click on Fork > to obtain the repo



* + 1. In your repo, update the \_artifactsLocation parameter in **Create and provision WVD host pool\mainTemplate.json** to the raw URL of the file on GitHub (Ex: <https://raw.githubusercontent.com/yourusername/RDS-Templates/master/wvd-templates/Create%20and%20provision%20WVD%20host%20pool/mainTemplate.json>)



* + 1. Update the rdshGalleryImageSKU parameter as per the below image in the files below:
       - ***Create and provision WVD host pool\mainTemplate.json***
       - ***Create and provision WVD host pool\nestedtemplates\managedDisks-galleryvm.json***
       - ***Create and provision WVD host pool\nestedtemplates\unmanagedDisks-galleryvm.json***



* + 1. Now follow instructions [here](https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-template-deploy-portal#deploy-resources-from-custom-template) to deploy using a custom ARM template