**Windows Virtual Machine**

 You choose a VM when you need more control over the computing environment than the other choices offer. An Azure VM gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it. However, you still need to maintain the VM by performing tasks, such as configuring, patching, and installing the software that runs on it.

**Things to Consider before creating a VM**

**Name** : Name of the VM and Computer Name

**Locations**

**VM Size** : Size is determined by Processing Power, Memory and Storage Capacity

**VM Limits** : Your subscription has default [quota limits](https://docs.microsoft.com/en-us/azure/azure-subscription-service-limits) in place that could impact the deployment of many VMs for your project. The current limit on a per subscription basis is 20 VMs per region.

### Operating system disks and images

Virtual machines use [virtual hard disks (VHDs)](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/about-disks-and-vhds?toc=%2fazure%2fvirtual-machines%2fwindows%2ftoc.json) to store their operating system (OS) and data. VHDs are also used for the images you can choose from to install an OS.

Azure provides many [marketplace images](https://azure.microsoft.com/marketplace/virtual-machines/) to use with various versions and types of Windows Server operating systems. Marketplace images are identified by image publisher, offer, sku, and version (typically version is specified as latest). Only 64-bit operating systems are supported.

**Extensions**

VM [extensions](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/extensions-features?toc=%2fazure%2fvirtual-machines%2fwindows%2ftoc.json) give your VM additional capabilities through post deployment configuration and automated tasks.

These common tasks can be accomplished using extensions:

* Run custom scripts – The [Custom Script Extension](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/extensions-customscript?toc=%2fazure%2fvirtual-machines%2fwindows%2ftoc.json) helps you configure workloads on the VM by running your script when the VM is provisioned.
* Deploy and manage configurations – The [PowerShell Desired State Configuration (DSC) Extension](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/extensions-dsc-overview?toc=%2fazure%2fvirtual-machines%2fwindows%2ftoc.json) helps you set up DSC on a VM to manage configurations and environments.

**Manage availability**

In order for your deployment to qualify for our 99.95 VM Service Level Agreement, you need to deploy two or more VMs running your workload inside an [availability set](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/tutorial-availability-sets?toc=%2fazure%2fvirtual-machines%2fwindows%2ftoc.json). This configuration ensures your VMs are distributed across multiple fault domains

## **Create virtual machine**

Create a virtual machine using powershell.

New-AzureRmVm `

-ResourceGroupName "myResourceGroup" `

-Name "myVM" `

-Location "East US" `

-VirtualNetworkName "myVnet" `

-SubnetName "mySubnet" `

-SecurityGroupName "myNetworkSecurityGroup" `

-PublicIpAddressName "myPublicIpAddress" `

-OpenPorts 80,3389

**Understand VM images**

The Azure marketplace includes many virtual machine images that can be used to create a new virtual machine. In the previous steps, a virtual machine was created using the Windows Server 2016 Datacenter image. In this step, the PowerShell module is used to search the marketplace for other Windows images, which can also be used as a base for new VMs. This process consists of finding the publisher, offer, SKU, and optionally a version number to [identify](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/cli-ps-findimage#terminology) the image.

Get-AzureRmVMImagePublisher -Location "East US"

MicrosoftWindowsServer is the output

Get-AzureRmVMImageOffer -Location "East US" -PublisherName "MicrosoftWindowsServer"

WindowsServer is the output

Get-AzureRmVMImageSku -Location "East US" -PublisherName "MicrosoftWindowsServer" -Offer "WindowsServer"

2016-Datacenter-with-Containers is the output

Above Information can be used to deploy a VM as per the following script

New-AzureRmVm `

-ResourceGroupName "myResourceGroupVM" `

-Name "myVM2" `

-Location "EastUS" `

-VirtualNetworkName "myVnet" `

-SubnetName "mySubnet" `

-SecurityGroupName "myNetworkSecurityGroup" `

-PublicIpAddressName "myPublicIpAddress2" `

-ImageName "MicrosoftWindowsServer:WindowsServer:2016-Datacenter-with-Containers:latest" `

-Credential $cred `

-AsJob

**Manage Azure disks with Azure PowerShell**

**Default Azure disks**

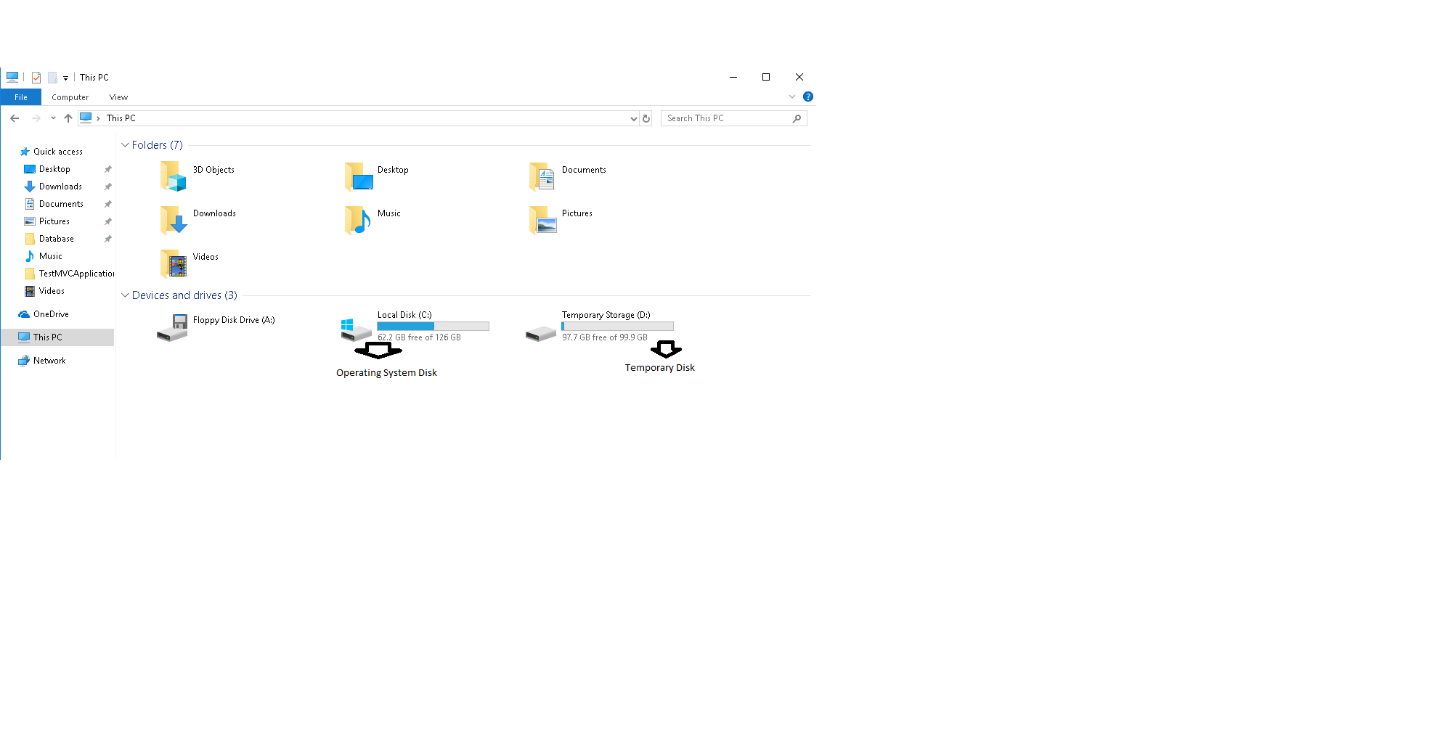
When an Azure virtual machine is created, two disks are automatically attached to the virtual machine.

**Operating system disk**

Operating system disks can be sized up to 4 terabytes, and hosts the VMs operating system. The OS disk is assigned a drive letter of *C:* by default. The disk caching configuration of the OS disk is optimized for OS performance. The OS disk **should not** host applications or data. For applications and data, use a data disk, which is detailed later in this article.

**Temporary disk**

Temporary disks use a solid-state drive that is located on the same Azure host as the VM. Temp disks are highly performant and may be used for operations such as temporary data processing. However, if the VM is moved to a new host, any data stored on a temporary disk is removed. The size of the temporary disk is determined by the [VM size](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes). Temporary disks are assigned a drive letter of *D:* by default.

****

**Azure data disks**

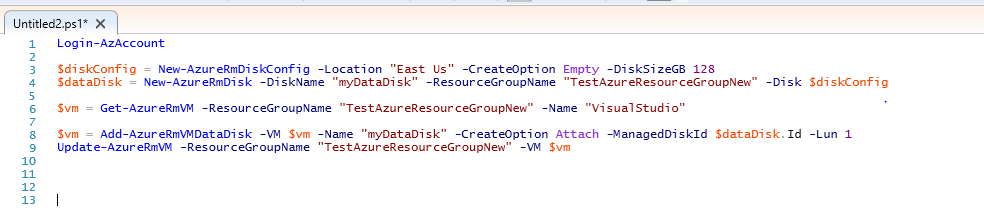
Additional data disks can be added for installing applications and storing data. Data disks should be used in any situation where durable and responsive data storage is needed.

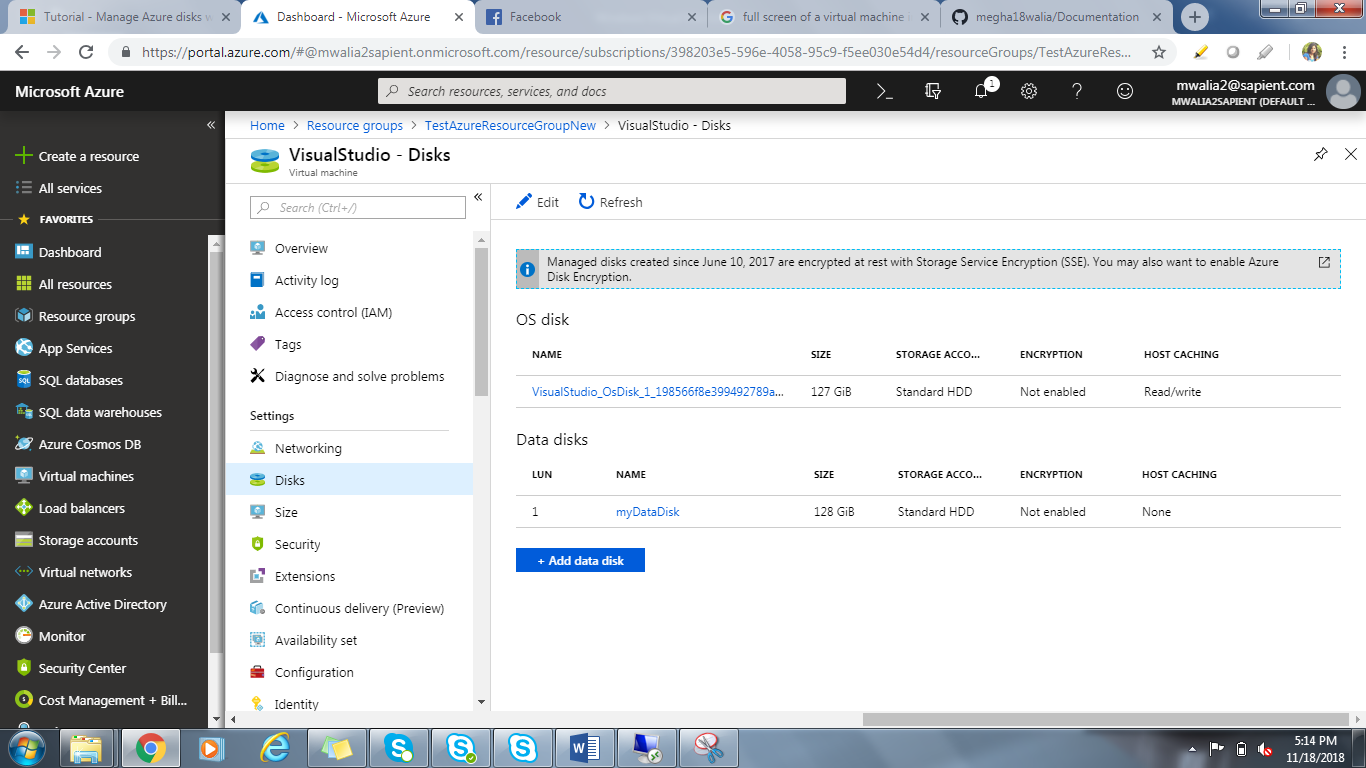
**VM disk types**

**Standard disks** - backed by HDDs, and delivers cost-effective storage while still being performant

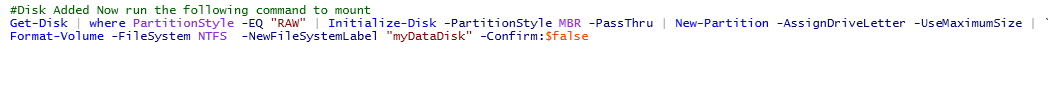
**Premium disks -** backed by SSD-based high-performance, low-latency disk. Perfect for VMs running production workload.

**Create and attach disks**





Now the disk is bound and attached, Finally it has to be mount in the existing VM. Take Remote of the VM and run the PowerShell command.



# **Deploy applications to a Windows virtual machine in Azure with the Custom Script Extension**

The Custom Script Extension downloads and executes scripts on Azure VMs. This extension is useful for post deployment configuration, software installation, or any other configuration / management task. Scripts can be downloaded from Azure storage or GitHub, or provided to the Azure portal at extension run time.

# **Create a custom image of an Azure VM with Azure PowerShell**

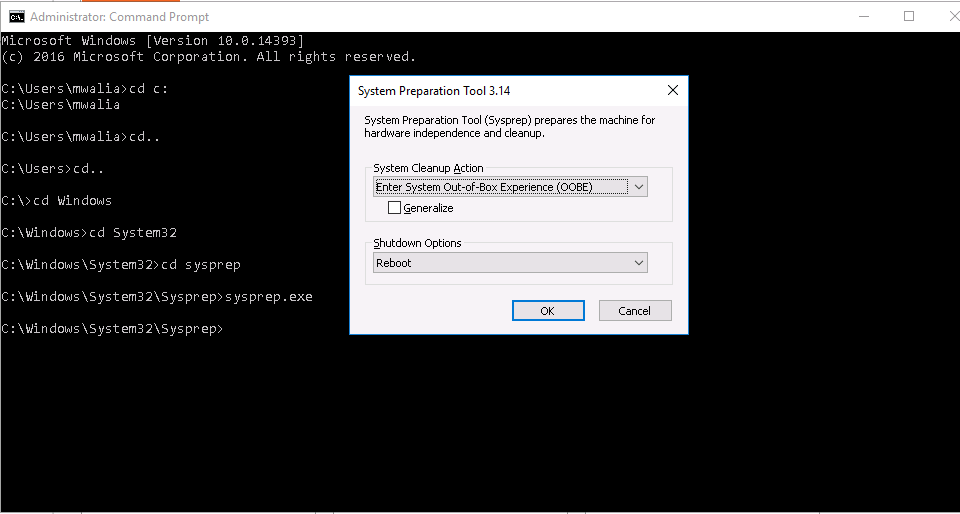
how to take an existing VM and turn it into a re-usable custom image that you can use to create new VM instances.

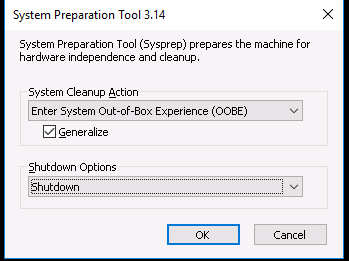
## **Prepare VM**

To create an image of a virtual machine, you need to prepare the VM by generalizing the VM, deallocating, and then marking the source VM as generalized in Azure.

## **Generalize the Windows VM using Sysprep**

1. Connect to the virtual machine.
2. Open the Command Prompt window as an administrator. Change the directory to *%windir%\system32\sysprep*, and then run *sysprep.exe*.
3. In the **System Preparation Tool** dialog box, select *Enter System Out-of-Box Experience (OOBE)*, and make sure that the *Generalize* check box is selected.
4. In **Shutdown Options**, select *Shutdown* and then click **OK**.
5. When Sysprep completes, it shuts down the virtual machine. **Do not restart the VM**.





## **Deallocate and mark the VM as generalized**

Deallocated the VM using [Stop-AzureRmVM](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/stop-azurermvm).

Stop-AzureRmVM -ResourceGroupName myResourceGroup -Name myVM –Force

Set the status of the virtual machine to -Generalized using [Set-AzureRmVm](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/set-azurermvm).

Set-AzureRmVM -ResourceGroupName myResourceGroup -Name myVM –Generalized

## **Create the image**

$vm = Get-AzureRmVM -Name myVM -ResourceGroupName myResourceGroup

$image = New-AzureRmImageConfig -Location EastUS -SourceVirtualMachineId $vm.ID

New-AzureRmImage -Image $image -ImageName myImage -ResourceGroupName myResourceGroup

## **Create VMs from the image**

Now that you have an image, you can create one or more new VMs from the image. Creating a VM from a custom image is similar to creating a VM using a Marketplace image.

New-AzureRmVm `

-ResourceGroupName "myResourceGroup" `

-Name "myVMfromImage" `

-ImageName "myImage" `

-Location "East US" `

-VirtualNetworkName "myImageVnet" `

-SubnetName "myImageSubnet" `

-SecurityGroupName "myImageNSG" `

-PublicIpAddressName "myImagePIP" `

-OpenPorts 3389