**LABS Day 3**

Version 1.1

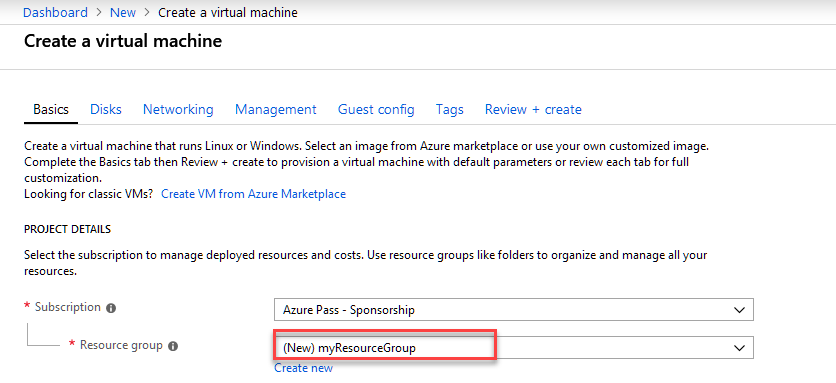
**Lab 2 Create a Windows virtual machine in the Azure portal**

**Task 1 Sign in to Azure**

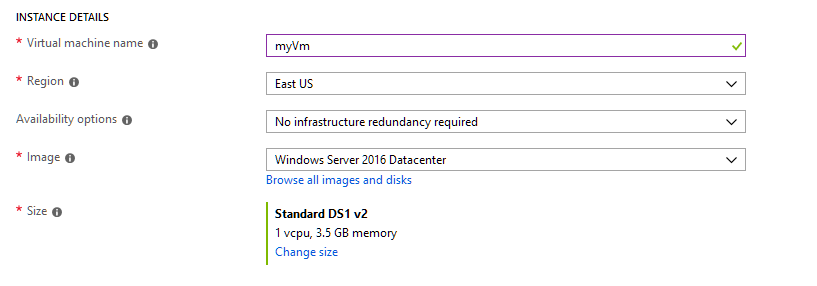
Sign in to the Azure portal at [https://portal.azure.com](https://portal.azure.com/).

**Task 2 Create virtual machine**

1. Choose **Create a resource** in the upper left-hand corner of the Azure portal.
2. In the search box above the list of Azure Marketplace resources, search for and select **Windows Server 2016 Datacenter**, then ensure the deployment model is set to Resource Manager, then choose **Create**.
3. In the **Basics** tab, under **Project details**, make sure the correct subscription is selected and then choose to **Create new** resource group. Type *myResourceGroup* for the name.



1. Under **Instance details**, type *myVM* for the **Virtual machine name** and choose *East US* for your **Location**. Leave the other defaults.

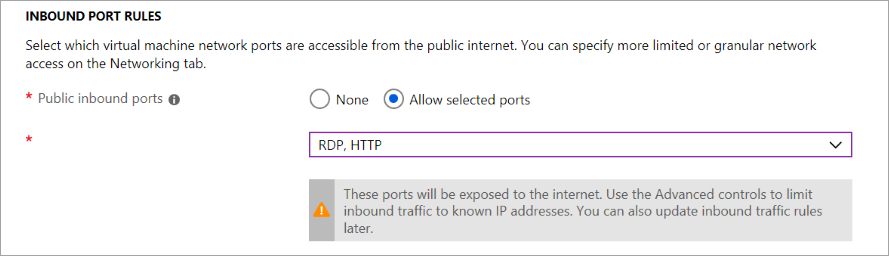


1. Under **Administrator account**, provide a username, such as *azureuser* and a password. The password must be at least 12 characters long and meet the [defined complexity requirements](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/faq#what-are-the-password-requirements-when-creating-a-vm). Try **Pa55w.rd1234**

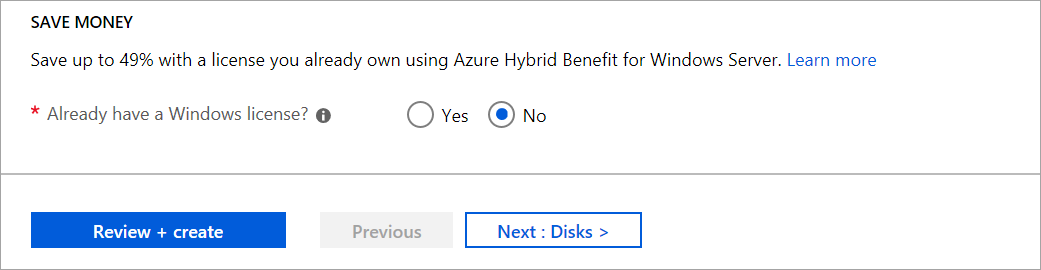
**Note**: Ensure you document the username and password



1. Under **Inbound port rules**, choose **Allow selected ports** and then select **RDP (3389)** and **HTTP** from the drop-down.



1. Leave the remaining defaults and then select the **Review + create** button at the bottom of the page.

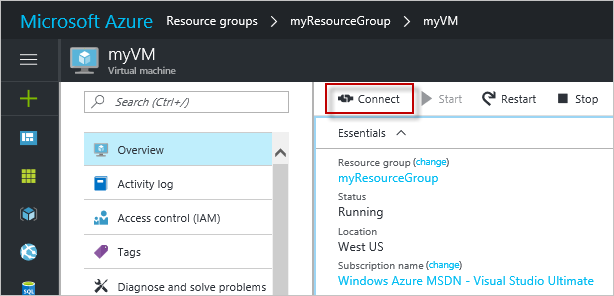


8. Then select **Create**

**Task 3 Connect to virtual machine**

Create a remote desktop connection to the virtual machine. These directions tell you how to connect to your VM from a Windows computer. On a Mac, you need an RDP client such as this [Remote Desktop Client](https://itunes.apple.com/us/app/microsoft-remote-desktop/id715768417?mt=12) from the Mac App Store.

1. Click the **Connect** button on the virtual machine properties page.



1. In the **Connect to virtual machine** page, keep the default options to connect by DNS name over port 3389 and click **Download RDP file**.
2. Open the downloaded RDP file and click **Connect** when prompted.
3. In the **Windows Security** window, select **More choices** and then **Use a different account**. Type the username as **localhost**\*username*, enter password you created for the virtual machine, and then click **OK**.
4. You may receive a certificate warning during the sign-in process. Click **Yes** or **Continue** to create the connection.

**Task 5 Install web server**

To see your VM in action, install the IIS web server. Open a PowerShell prompt on the VM and run the following command:

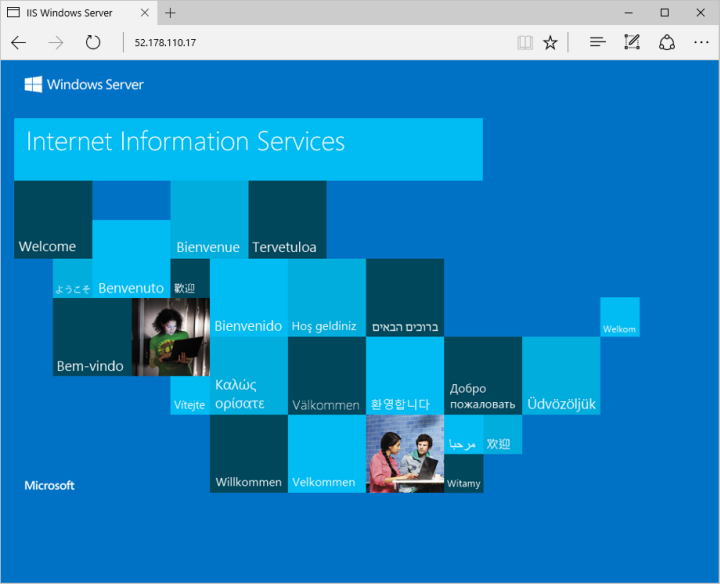
Install-WindowsFeature -name Web-Server -IncludeManagementTools

**Warning If an error occurs remove** -IncludeManagementTools

When done, close the RDP connection to the VM.

**Task 6 View the IIS welcome page**

In the portal, select the VM and in the overview of the VM, use the **Click to copy** button to the right of the IP address to copy it and paste it into a browser tab. The default IIS welcome page will open, and should look like this:



**Lab 3A Create a custom image of an Azure VM with Azure PowerShell**

* ‎Custom images are like marketplace images, but you create them yourself. Custom images can be used to bootstrap configurations such as preloading applications, application configurations, and other OS configurations. In this lab, you create your own custom image of an Azure virtual machine. You learn how to:
* Sysprep and generalize VMs
* Create a custom image
* Create a VM from a custom image
* List all the images in your subscription
* Delete an image

**Before you begin**

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

To complete the example in this lab, you must have an existing virtual machine. If needed, this [script sample](https://docs.microsoft.com/en-us/azure/virtual-machines/scripts/virtual-machines-windows-powershell-sample-create-vm) can create one for you. When working through the tutorial, replace the resource group and VM names where needed.

|  |  |
| --- | --- |
|  |  |

**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.

**Task 1 Generalize the Windows VM using Sysprep**

Sysprep removes all your personal account information, among other things, and prepares the machine to be used as an image. For details about Sysprep, see [How to Use Sysprep: An Introduction](https://technet.microsoft.com/library/bb457073.aspx).

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**.

**Task 2 Launch Azure Cloud Shell**

|  |  |
| --- | --- |
| Open Cloud Shell in your browser. | <https://shell.azure.com/powershell> |
| Click the **Cloud Shell** button on the menu in the upper right of the Azure portal. | [Cloud Shell in the portal](https://portal.azure.com/) |

**Task 3 Deallocate and mark the VM as generalized**

To create an image, the VM needs to be deallocated and marked as generalized in Azure.

* + 1. 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

* + 1. 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

**Task 4 Create the image**

Now you can create an image of the VM by using [New-AzureRmImageConfig](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermimageconfig) and [New-AzureRmImage](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermimage). The following commands create an image named *myImage* from a VM named *myVM, change the name to the name of your VM*.

1. Get the virtual machine.

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

1. Create the image configuration.

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

1. Create the image.

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

**Task 5 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. When you use a Marketplace image, you have to provide the information about the image, image provider, offer, SKU, and version. Using the simplified parameter set for the [New-AzureRMVM](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermvm) cmdlet, you just need to provide the name of the custom image as long as it is in the same resource group.

1. This command creates a VM named *myVMfromImage* from the *myImage*, in the *myResourceGroup*.

New-AzureRmVm `

-ResourceGroupName "myResourceGroup" `

-Name "myVMfromImage" `

-ImageName "myImage" `

-Location "East US" `

-VirtualNetworkName "myImageVnet" `

-SubnetName "myImageSubnet" `

-SecurityGroupName "myImageNSG" `

-PublicIpAddressName "myImagePIP" `

-OpenPorts 3389

**Task 6 Image management**

1. Delete an image. This example deletes the image named *myImage* from the *myResourceGroup*.

Remove-AzureRmImage `

-ImageName myImage `

-ResourceGroupName myResourceGroup

**Lab 4A Filter network traffic with a network security group using the Azure Portal**

* ‎You can filter network traffic inbound to and outbound from a virtual network subnet with a network security group. Network security groups contain security rules that filter network traffic by IP address, port, and protocol. Security rules are applied to resources deployed in a subnet. In this lab, you learn how to:
* Create a network security group and security rules
* Create a virtual network and associate a network security group to a subnet
* Deploy virtual machines (VM) into a subnet
* Test traffic filters

**Task 1 Sign in to Azure**

* 1. Sign in to the Azure portal at [https://portal.azure.com](https://portal.azure.com/).

**Task 2 Create a virtual network**

1. Select **+ Create a resource** on the upper, left corner of the Azure portal.
2. Select **Networking**, and then select **Virtual network**.
3. Enter, or select, the following information, accept the defaults for the remaining settings, and then select **Create**:

| **Setting** | **Value** |
| --- | --- |
| Name | myVirtualNetwork |
| Address space | 10.0.0.0/16 |
| Subscription | Select your subscription. |
| Resource group | Select **Create new** Reource Group named *myResourceGroup*. |
| Location | Select **East US**. |
| Subnet- Name | mySubnet |
| Subnet - Address range | 10.0.0.0/24 |

**Task 3 Create application security groups**

An application security group enables you to group together servers with similar functions, such as web servers.

1. Select **+ Create a resource** on the upper, left corner of the Azure portal.
2. In the **Search the Marketplace** box, enter *Application security group*. When **Application security group** appears in the search results, select it, select **Application security group** again under **Everything**, and then select **Create**.
3. Enter, or select, the following information, and then select **Review** + **Create** then **Create**:

| **Setting** | **Value** |
| --- | --- |
| Name | myAsgWebServers |
| Subscription | Select your subscription. |
| Resource group | Select **Use existing** and then select **myResourceGroup**. |
| Location | East US |

1. Complete step 3 again, specifying the following values:

| **Setting** | **Value** |
| --- | --- |
| Name | myAsgMgmtServers |
| Subscription | Select your subscription. |
| Resource group | Select **Use existing** and then select **myResourceGroup**. |
| Location | East US |

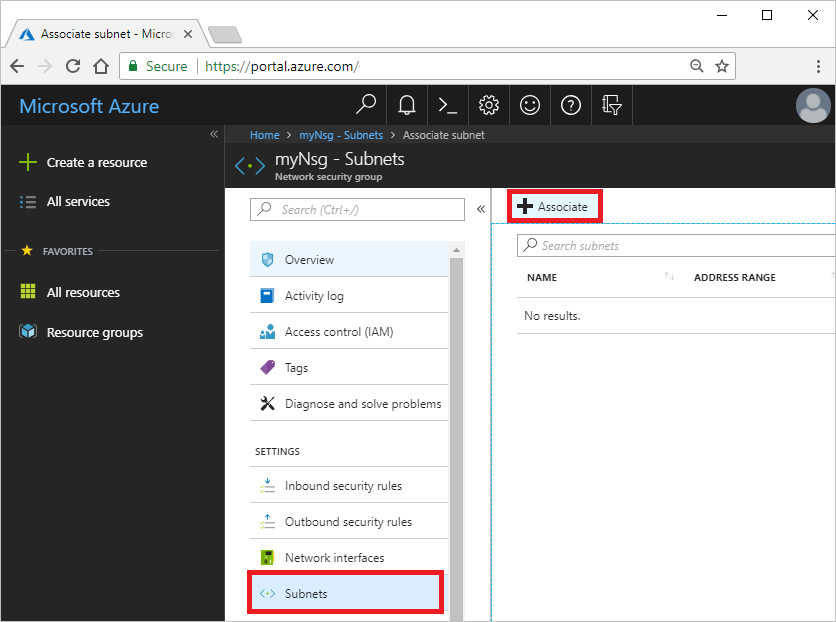
**Task 4 Create a network security group**

1. Select **+ Create a resource** on the upper, left corner of the Azure portal.
2. Select **Networking**, and then select **Network security group**.
3. Enter, or select, the following information, and then select **Create**:

| **Setting** | **Value** |
| --- | --- |
| Name | myNsg |
| Subscription | Select your subscription. |
| Resource group | Select **Use existing** and then select *myResourceGroup*. |
| Location | East US |

**Associate network security group to subnet**

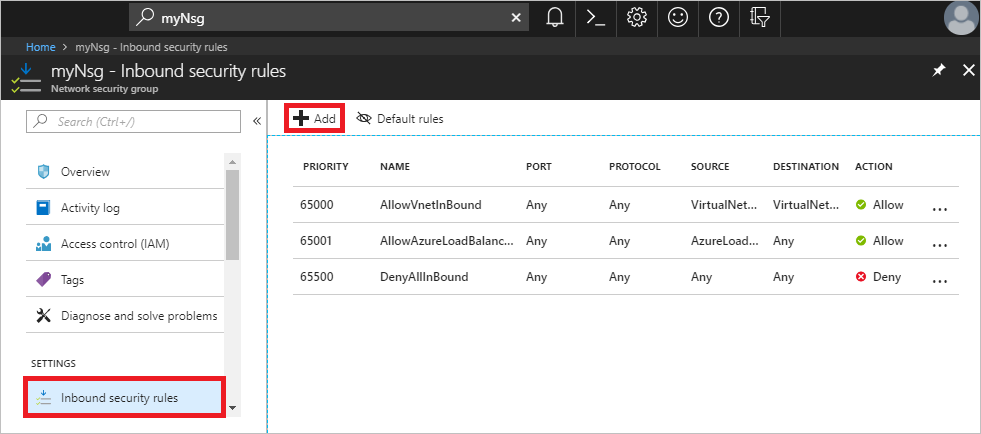
1. In the *Search resources, services, and docs* box at the top of the portal, begin typing *myNsg*. When **myNsg** appears in the search results, select it.
2. Under **SETTINGS**, select **Subnets** and then select **+ Associate**, as shown in the following picture:



1. Under **Associate subnet**, select **Virtual network** and then select **myVirtualNetwork**. Select **Subnet**, select **mySubnet**, and then select **OK**.

**Task 5 Create security rules**

1. Under **SETTINGS**, select **Inbound security rules** and then select **+ Add**, as shown in the following picture:



1. Create a security rule that allows ports 80 and 443 to the **myAsgWebServers** application security group. Under **Add inbound security rule**, enter, or select the following values, accept the remaining defaults, and then select **Add**:

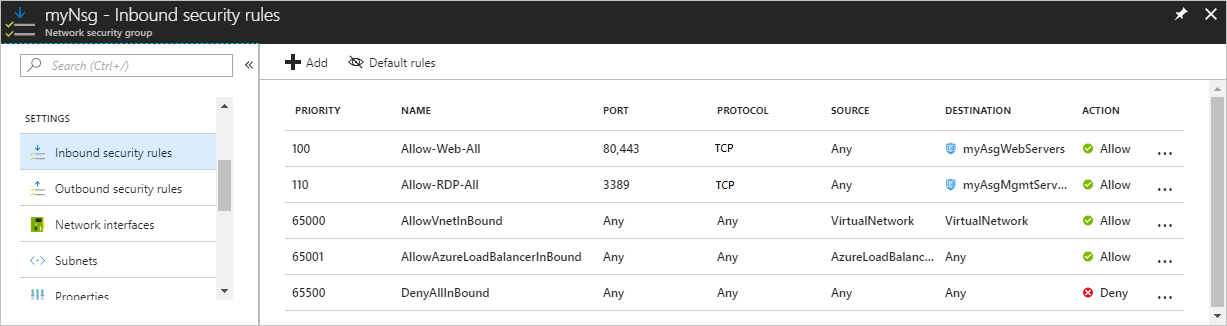
| **Setting** | **Value** |
| --- | --- |
| Destination | Select **Application security group**, and then select **myAsgWebServers** for **Application security group**. |
| Destination port ranges | Enter 80,443 |
| Protocol | Select TCP |
| Name | Allow-Web-All |

1. Complete step 2 again, using the following values:

| **Setting** | **Value** |
| --- | --- |
| Destination | Select **Application security group**, and then select **myAsgMgmtServers** for **Application security group**. |
| Destination port ranges | Enter 3389 |
| Protocol | Select TCP |
| Priority | Enter 110 |
| Name | Allow-RDP-All |

1. In this lab, RDP (port 3389) is exposed to the internet for the VM that is assigned to the *myAsgMgmtServers* application security group. For production environments, instead of exposing port 3389 to the internet, it's recommended that you connect to Azure resources that you want to manage using a VPN or private network connection.

Once you've completed steps 1-3, review the rules you created. Your list should look like the list in the following picture:



**Create virtual machines**

Create two VMs in the virtual network.

**Task 6 Create the first VM**

1. Select **+ Create a resource** found on the upper, left corner of the Azure portal.
2. Select **Compute**, and then select **Windows Server 2016 Datacenter**.
3. Ensure the Resource Manager is selected as the deployment model
4. Select **Create**
5. Enter, or select, the following information, accept the defaults for the remaining settings, and then select **OK**:

| **Setting** | **Value** |
| --- | --- |
| Name | myVmWeb |
| User name | Enter a user name of yo ur choosing. |
| Password | Enter a password of your choosing. The password must be at least 12 characters long and meet the [defined complexity requirements](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/faq?toc=%2fazure%2fvirtual-network%2ftoc.json#what-are-the-password-requirements-when-creating-a-vm). Try **Pa55w.rd1234** |
| Subscription | Select your subscription, which should be already be done |
| Resource group | Select **Use existing** and select **myResourceGroup**. |
| Location | Select **East US** |

1. Select a size for the VM and then select **Select**.
2. Choose the **Networking** tab:

| **Setting** | **Value** |
| --- | --- |
| Virtual network | Select **myVirtualNetwork** |
| Network Security Group | Select **Advanced**. |
| Network security group (firewall) | Select **(new) myVmWeb-nsg**. |

1. Select the **Manangment** Tab and assign Diagnostic storage account.
2. Select **Review + Create** then select **Create** to start VM deployment.

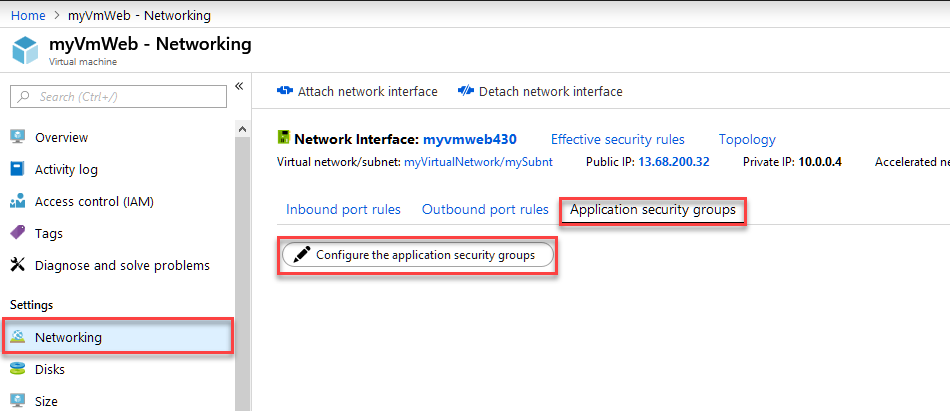
**Create the second VM**

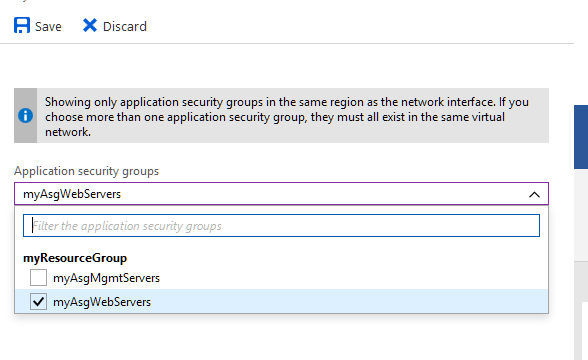
Complete steps 1-9 again, but in step 3, name the VM *myVmMgmt*. The VM takes a few minutes to deploy. Do not continue to the next step until the VM is deployed.

**Task 7 Associate network interfaces to an ASG**

When the portal created the VMs, it created a network interface for each VM, and attached the network interface to the VM. Add the network interface for each VM to one of the application security groups you created previously:

1. In the *Search resources, services, and docs* box at the top of the portal, begin typing *myVmWeb*. When the **myVmWeb** VM appears in the search results, select it.
2. Under **SETTINGS**, select **Networking**. Select **Configure the application security groups**, select **myAsgWebServers** for **Application security groups**, and then select **Save**, as shown in the following pictures:





1. Complete steps 1 and 2 agai n, searching for the **myVmMgmt** VM and selecting the **myAsgMgmtServers** ASG.

**Task 8 Test traffic filters**

1. Connect to the *myVmMgmt* VM. Enter *myVmMgmt* in the search box at the top of the portal. When **myVmMgmt** appears in the search results, select it. Select the **Connect** button.
2. Select **Download RDP file**.
3. Open the downloaded rdp file and select **Connect**. Enter the user name and password you specified when creating the VM. You may need to select **More choices**, then **Use a different account**, to specify the credentials you entered when you created the VM.
4. Select **OK**.
5. You may receive a certificate warning during the sign-in process. If you receive the warning, select **Yes** or **Continue**, to proceed with the connection.

The connection succeeds, because port 3389 is allowed inbound from the internet to the *myAsgMgmtServers* application security group that the network interface attached to the *myVmMgmt* VM is in.

1. Connect to the *myVmWeb* VM from the *myVmMgmt* VM by entering the following command in a PowerShell session:

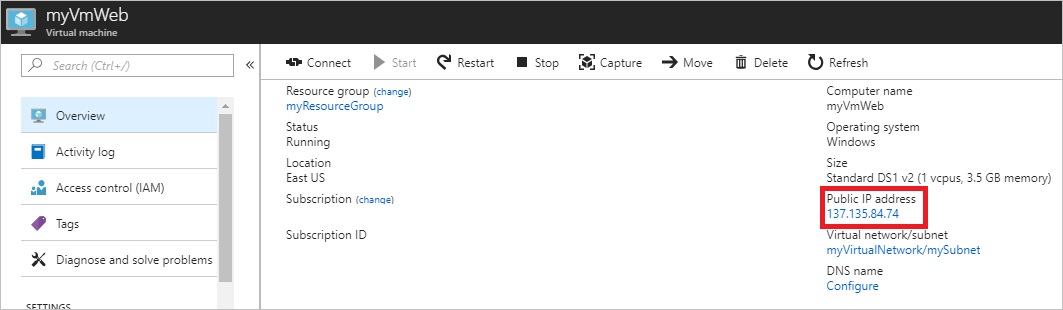
mstsc /v:myVmWeb

You are able to connect to the myVmWeb VM from the myVmMgmt VM because VMs in the same virtual network can communicate with each other over any port, by default. You can't however, create a remote desktop connection to the *myVmWeb* VM from the internet, because the security rule for the *myAsgWebServers* doesn't allow port 3389 inbound from the internet and inbound traffic from the Internet is denied to all resources, by default.

1. To install Microsoft IIS on the *myVmWeb* VM, enter the following command from a PowerShell session on the *myVmWeb* VM:

Install-WindowsFeature -name Web-Server

1. After the IIS installation is complete, disconnect from the *myVmWeb* VM, which leaves you in the *myVmMgmt* VM remote desktop connection.
2. Disconnect from the *myVmMgmt* VM.
3. In the *Search resources, services, and docs* box at the top of the Azure portal, begin typing *myVmWeb* from your computer. When **myVmWeb** appears in the search results, select it. Note the **Public IP address** for your VM. The address shown in the following picture is 137.135.84.74, but your address is different:



1. To confirm that you can access the *myVmWeb* web server from the internet, open an internet browser on your computer and browse to http://<public-ip-address-from-previous-step>. You see the IIS welcome screen, because port 80 is allowed inbound from the internet to the *myAsgWebServers* application security group that the network interface attached to the *myVmWeb* VM is in.

**Task 9 Clean up resources**

When no longer needed, delete the resource group and all of the resources it contains:

1. Enter *myResourceGroup* in the **Search** box at the top of the portal. When you see **myResourceGroup** in the search results, select it.
2. Select **Delete resource group**.
3. Enter *myResourceGroup* for **TYPE THE RESOURCE GROUP NAME:** and select **Delete**.

**Lab 4B Create and manage a Windows virtual machine that has multiple NICs**

* ‎Virtual machines (VMs) in Azure can have multiple virtual network interface cards (NICs) attached to them. A common scenario is to have different subnets for front-end and back-end connectivity. You can associate multiple NICs on a VM to multiple subnets, but those subnets must all reside in the same virtual network (vNet). This lab details how to create a VM that has multiple NICs attached to it. You also learn how to add or remove NICs from an existing VM. Different [VM sizes](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes) support a varying number of NICs, so size your VM accordingly.

**Create a VM with multiple NICs**

**Task 1 First, create a resource group.**

The following example creates a resource group named *myResourceGroup2* in the *EastUs* location:

PowerShell

New-AzureRmResourceGroup -Name "myResourceGroup2" -Location "EastUS"

**Task 2 Create virtual network and subnets**

A common scenario is for a virtual network to have two or more subnets. One subnet may be for front-end traffic, the other for back-end traffic. To connect to both subnets, you then use multiple NICs on your VM.

1. Define two virtual network subnets with [New-AzureRmVirtualNetworkSubnetConfig](https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermvirtualnetworksubnetconfig). The following example defines the subnets for *mySubnetFrontEnd* and *mySubnetBackEnd*:

$mySubnetFrontEnd = New-AzureRmVirtualNetworkSubnetConfig -Name "mySubnetFrontEnd" `

-AddressPrefix "192.168.1.0/24"

$mySubnetBackEnd = New-AzureRmVirtualNetworkSubnetConfig -Name "mySubnetBackEnd" `

-AddressPrefix "192.168.2.0/24"

1. Create your virtual network and subnets with [New-AzureRmVirtualNetwork](https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermvirtualnetwork). The following example creates a virtual network named *myVnet*:

$myVnet = New-AzureRmVirtualNetwork -ResourceGroupName "myResourceGroup2" `

-Location "EastUs" `

-Name "myVnet" `

-AddressPrefix "192.168.0.0/16" `

-Subnet $mySubnetFrontEnd,$mySubnetBackEnd

**Create multiple NICs**

Create two NICs with [New-AzureRmNetworkInterface](https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermnetworkinterface). Attach one NIC to the front-end subnet and one NIC to the back-end subnet. The following example creates NICs named *myNic1* and *myNic2*:

$frontEnd = $myVnet.Subnets|?{$\_.Name -eq 'mySubnetFrontEnd'}

$myNic1 = New-AzureRmNetworkInterface -ResourceGroupName "myResourceGroup2" `

-Name "myNic1" `

-Location "EastUs" `

-SubnetId $frontEnd.Id

$backEnd = $myVnet.Subnets|?{$\_.Name -eq 'mySubnetBackEnd'}

$myNic2 = New-AzureRmNetworkInterface -ResourceGroupName "myResourceGroup2" `

-Name "myNic2" `

-Location "EastUs" `

-SubnetId $backEnd.Id

Typically you also create a [network security group](https://docs.microsoft.com/en-us/azure/virtual-network/security-overview) to filter network traffic to the VM and a [load balancer](https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview) to distribute traffic across multiple VMs.

**Task 3 Create the virtual machine**

Now start to build your VM configuration. Each VM size has a limit for the total number of NICs that you can add to a VM. For more information, see [Windows VM sizes](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes).

1. Set your VM credentials to the $cred variable as follows:

$cred = Get-Credential

1. Define your VM with [New-AzureRmVMConfig](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermvmconfig). The following example defines a VM named *myVM* and uses a VM size that supports more than two NICs (*Standard\_DS3\_v2*):

$vmConfig = New-AzureRmVMConfig -VMName "myVM" -VMSize "Standard\_DS3\_v2"

1. Define the Resource Group. Use a name that does not exist.

$rgName = 'myResourceGroup2'

1. Create the rest of your VM configuration with [Set-AzureRmVMOperatingSystem](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/set-azurermvmoperatingsystem) and [Set-AzureRmVMSourceImage](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/set-azurermvmsourceimage). The following example creates a Windows Server 2016 VM:

$vmConfig = Set-AzureRmVMOperatingSystem -VM $vmConfig `

-Windows `

-ComputerName "myVM" `

-Credential $cred `

-ProvisionVMAgent `

-EnableAutoUpdate

$vmConfig = Set-AzureRmVMSourceImage -VM $vmConfig `

-PublisherName "MicrosoftWindowsServer" `

-Offer "WindowsServer" `

-Skus "2016-Datacenter" `

-Version "latest"

1. Attach the two NICs that you previously created with [Add-AzureRmVMNetworkInterface](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/add-azurermvmnetworkinterface):

$vmConfig = Add-AzureRmVMNetworkInterface -VM $vmConfig -Id $myNic1.Id -Primary

$vmConfig = Add-AzureRmVMNetworkInterface -VM $vmConfig -Id $myNic2.Id

1. Create your VM with [New-AzureRmVM](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermvm):

New-AzureRmVM -VM $vmConfig -ResourceGroupName $rgName -Location "EastUs"

**Task 4 Add a NIC to an existing VM**

To add a virtual NIC to an existing VM, you deallocate the VM, add the virtual NIC, then start the VM. Different [VM sizes](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes) support a varying number of NICs, so size your VM accordingly. If needed, you can [resize a VM](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/resize-vm).

1. Deallocate the VM with [Stop-AzureRmVM](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/stop-azurermvm). The following example deallocates the VM named *myVM* in *myResourceGroup2*:

Stop-AzureRmVM -Name "myVM" -ResourceGroupName $rgName

1. Get the existing configuration of the VM with [Get-AzureRmVm](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/get-azurermvm). The following example gets information for the VM named *myVM* in *myResourceGroup2*:

$vm = Get-AzureRmVm -Name "myVM" -ResourceGroupName $rgName

1. The following example creates a virtual NIC with [New-AzureRmNetworkInterface](https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermnetworkinterface) named *myNic3* that is attached to *mySubnetBackEnd*. The virtual NIC is then attached to the VM named *myVM* in *myResourceGroup* with [Add-AzureRmVMNetworkInterface](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/add-azurermvmnetworkinterface):

# Get info for the back end subnet

$myVnet = Get-AzureRmVirtualNetwork -Name "myVnet" -ResourceGroupName $rgName

$backEnd = $myVnet.Subnets|?{$\_.Name -eq 'mySubnetBackEnd'}

# Create a virtual NIC

$myNic3 = New-AzureRmNetworkInterface -ResourceGroupName $rgName `

-Name "myNic3" `

-Location "EastUs" `

-SubnetId $backEnd.Id

# Get the ID of the new virtual NIC and add to VM

$nicId = (Get-AzureRmNetworkInterface -ResourceGroupName $rgName -Name "MyNic3").Id

Add-AzureRmVMNetworkInterface -VM $vm -Id $nicId | Update-AzureRmVm -ResourceGroupName $rgName

**Primary virtual NICs**

One of the NICs on a multi-NIC VM needs to be primary. If one of the existing virtual NICs on the VM is already set as primary, you can skip this step. The following example assumes that two virtual NICs are now present on a VM and you wish to add the first NIC ([0]) as the primary:

# List existing NICs on the VM and find which one is primary

$vm.NetworkProfile.NetworkInterfaces | fl

# Set NIC 0 to be primary

$vm.NetworkProfile.NetworkInterfaces[0].Primary = $true

$vm.NetworkProfile.NetworkInterfaces[1].Primary = $false

# Update the VM state in Azure

Update-AzureRmVM -VM $vm -ResourceGroupName $rgName

1. Start the VM with [Start-AzureRmVm](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/start-azurermvm):

Start-AzureRmVM -ResourceGroupName $rgName -Name "myVM"

**Task 6 Remove a NIC from an existing VM**

To remove a virtual NIC from an existing VM, you deallocate the VM, remove the virtual NIC, then start the VM.

1. Deallocate the VM with [Stop-AzureRmVM](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/stop-azurermvm). The following example deallocates the VM named *myVM* in *myResourceGroup*:

Stop-AzureRmVM -Name "myVM" -ResourceGroupName $rgName

1. Get the existing configuration of the VM with [Get-AzureRmVm](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/get-azurermvm). The following example gets information for the VM named *myVM* in *myResourceGroup*:

$vm = Get-AzureRmVm -Name "myVM" -ResourceGroupName $rgName

1. Get information about the NIC remove with [Get-AzureRmNetworkInterface](https://docs.microsoft.com/en-us/powershell/module/azurerm.network/get-azurermnetworkinterface). The following example gets information about *myNic3*:

# List existing NICs on the VM if you need to determine NIC name

$vm.NetworkProfile.NetworkInterfaces

$nicId = (Get-AzureRmNetworkInterface -ResourceGroupName $rgName -Name "myNic3").Id

1. Remove the NIC with [Remove-AzureRmVMNetworkInterface](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/remove-azurermvmnetworkinterface) and then update the VM with [Update-AzureRmVm](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/update-azurermvm). The following example removes *myNic3* as obtained by $nicId in the preceding step:

Remove-AzureRmVMNetworkInterface -VM $vm -NetworkInterfaceIDs $nicId | `

Update-AzureRmVm -ResourceGroupName $rgName

1. Start the VM with [Start-AzureRmVm](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/start-azurermvm):

Start-AzureRmVM -Name $vm -ResourceGroupName $rgName

**Lab 4C**

**Exercise 1 Attach a data disk to a Windows VM with PowerShell**

* ‎This Lab shows you how to attach both new and existing disks to a Windows virtual machine by using PowerShell.

First, review these tips:

* The size of the virtual machine controls how many data disks you can attach. For more information, see [Sizes for virtual machines](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes?toc=%2fazure%2fvirtual-machines%2fwindows%2ftoc.json).
* To use Premium storage, you'll need a Premium Storage-enabled VM type like the DS-series or GS-series virtual machine. For more information, see [Premium Storage: High-Performance Storage for Azure Virtual Machine Workloads](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/premium-storage?toc=%2fazure%2fvirtual-machines%2fwindows%2ftoc.json).

## Task 1 Launch Azure Cloud Shell

|  |  |
| --- | --- |
| Open Cloud Shell in your browser. | <https://shell.azure.com/powershell> |
| Click the **Cloud Shell** button on the menu in the upper right of the Azure portal. |  |
|  |  |

## Task 2 Add an empty data disk to a virtual machine

This example shows how to add an empty data disk to an existing virtual machine.

### Using managed disks

PowerShell

**Ensure that the ResourceGroup Name and the VM match your naming.**

$rgName = 'myResourceGroup2'

$vmName = 'myVM'

$location = 'East US'

$storageType = 'Premium\_LRS'

$dataDiskName = $vmName + '\_datadisk1'

$diskConfig = New-AzureRmDiskConfig -SkuName $storageType -Location $location -CreateOption Empty -DiskSizeGB 128

$dataDisk1 = New-AzureRmDisk -DiskName $dataDiskName -Disk $diskConfig -ResourceGroupName $rgName

$vm = Get-AzureRmVM -Name $vmName -ResourceGroupName $rgName

$vm = Add-AzureRmVMDataDisk -VM $vm -Name $dataDiskName -CreateOption Attach -ManagedDiskId $dataDisk1.Id -Lun 1

Update-AzureRmVM -VM $vm -ResourceGroupName $rgName

### Task 3 Using managed disks in an Availability Zone

To create a disk in an Availability Zone, use [New-AzureRmDiskConfig](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermdiskconfig) with the -Zone parameter. The following example creates a disk in zone 1.

PowerShell

**Ensure that the ResourceGroup Name and the VM match your naming.**

$rgName = 'myResourceGroup2'

$vmName = 'myVM'

$location = 'East US 2'

$storageType = 'Premium\_LRS'

$dataDiskName = $vmName + '\_datadisk2'

$diskConfig = New-AzureRmDiskConfig -SkuName $storageType -Location $location -CreateOption Empty -DiskSizeGB 128 -Zone 1

$dataDisk1 = New-AzureRmDisk -DiskName $dataDiskName -Disk $diskConfig -ResourceGroupName $rgName

$vm = Get-AzureRmVM -Name $vmName -ResourceGroupName $rgName

$vm = Add-AzureRmVMDataDisk -VM $vm -Name $dataDiskName -CreateOption Attach -ManagedDiskId $dataDisk1.Id -Lun 2

Update-AzureRmVM -VM $vm -ResourceGroupName $rgName

### Task 4 Initialize the disk

After you add an empty disk, you'll need to initialize it. To initialize the disk, you can sign in to a VM and use disk management.

PowerShell

$disks = Get-Disk | Where partitionstyle -eq 'raw' | sort number

$letters = 70..89 | ForEach-Object { [char]$\_ }

$count = 0

$labels = "data1","data2"

foreach ($disk in $disks) {

$driveLetter = $letters[$count].ToString()

$disk |

Initialize-Disk -PartitionStyle MBR -PassThru |

New-Partition -UseMaximumSize -DriveLetter $driveLetter |

Format-Volume -FileSystem NTFS -NewFileSystemLabel $labels[$count] -Confirm:$false -Force

$count++

}

**Exercise 2** **How to detach a data disk from a Windows virtual machine**

* ‎When you no longer need a data disk that's attached to a virtual machine, you can easily detach it. This removes the disk from the virtual machine, but doesn't remove it from storage.

**Warning**

If you detach a disk it is not automatically deleted. If you have subscribed to Premium storage, you will continue to incur storage charges for the disk. For more information, see [Pricing and Billing when using Premium Storage](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/premium-storage#pricing-and-billing).

If you want to use the existing data on the disk again, you can reattach it to the same virtual machine, or another one.

**Task 1 Detach a data disk using PowerShell**

You can *hot* remove a data disk using PowerShell, but make sure nothing is actively using the disk before detaching it from the VM.

In this example, we remove the disk named **myDisk** from the VM **myVM** in the **myResourceGroup** resource group. First you remove the disk using the [Remove-AzureRmVMDataDisk](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/remove-azurermvmdatadisk) cmdlet. Then, you update the state of the virtual machine, using the [Update-AzureRmVM](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/update-azurermvm) cmdlet, to complete the process of removing the data disk.

PowerShell

**Ensure that the Resource Group name and VM name match yours.**

$VirtualMachine = Get-AzureRmVM -ResourceGroupName "myResourceGroup2" -Name "myVM"

Remove-AzureRmVMDataDisk -VM $VirtualMachine -Name "myDisk"

Update-AzureRmVM -ResourceGroupName "myResourceGroup2" -VM $VirtualMachine

The disk stays in storage but is no longer attached to a virtual machine.

**Task 2 Optional Detach a data disk using the portal**

1. In the left menu, select **Virtual Machines**.
2. Select the virtual machine that has the data disk you want to detach and click **Stop** to deallocate the VM.
3. In the virtual machine pane, select **Disks**.
4. At the top of the **Disks** pane, select **Edit**.
5. In the **Disks** pane, to the far right of the data disk that you would like to detach, click the Detach button imagedetach button.
6. After the disk has been removed, click **Save** on the top of the pane.
7. In the virtual machine pane, click **Overview** and then click the **Start** button at the top of the pane to restart the VM.

The disk stays in storage but is no longer attached to a virtual machine.

**Exercise 3** **Update the storage type of a managed disk**

* ‎Azure Managed Disks offers three storage type options: [Premium SSD](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/premium-storage), [Standard SSD](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/disks-standard-ssd), and [Standard HDD](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/standard-storage). You can switch a managed disk between storage types with minimal downtime, based on your performance needs. Switching between storage types is not supported for an unmanaged disk; however, you can easily [convert an unmanaged disk to a managed disk](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/convert-unmanaged-to-managed-disks).

This article shows how to convert a managed disk from standard to premium, and vice versa, by using Azure PowerShell. If you need to install or upgrade PowerShell, see [Install and configure Azure PowerShell](https://docs.microsoft.com/powershell/azure/install-azurerm-ps?view=azurermps-6.8.1).

**Task 1 Convert all the managed disks of a VM from standard to premium**

The following example shows how to switch all the disks of a VM from standard to premium storage. To use premium-managed disks, your VM must use a [VM size](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes) that supports premium storage. This example also switches to a size that supports premium storage

PowerShell

# Name of the resource group that contains the VM

$rgName = 'myResourceGroup2'

# Name of the your virtual machine

$vmName = 'myVM'

# Choose between StandardLRS and PremiumLRS based on your scenario

$storageType = 'Premium\_LRS'

# Premium capable size

# Required only if converting storage from standard to premium

$size = 'Standard\_DS2\_v2'

# Stop and deallocate the VM before changing the size

Stop-AzureRmVM -ResourceGroupName $rgName -Name $vmName -Force

$vm = Get-AzureRmVM -Name $vmName -resourceGroupName $rgName

# Change the VM size to a size that supports premium storage

# Skip this step if converting storage from premium to standard

$vm.HardwareProfile.VmSize = $size

Update-AzureRmVM -VM $vm -ResourceGroupName $rgName

# Get all disks in the resource group of the VM

$vmDisks = Get-AzureRmDisk -ResourceGroupName $rgName

# For disks that belong to the selected VM, convert to premium storage

foreach ($disk in $vmDisks)

{

if ($disk.ManagedBy -eq $vm.Id)

{

$diskUpdateConfig = New-AzureRmDiskUpdateConfig –AccountType $storageType

Update-AzureRmDisk -DiskUpdate $diskUpdateConfig -ResourceGroupName $rgName `

-DiskName $disk.Name

}

}

Start-AzureRmVM -ResourceGroupName $rgName -Name $vmName

**Task 2 Convert a managed disk from standard to premium**

For your dev/test workload, you may want a mixture of standard and premium disks to reduce your cost. To dos so, upgrade to premium storage only those disks that require better performance. The following example shows how to switch a single disk of a VM from standard to premium storage, and vice versa. To use premium-managed disks, your VM must use a [VM size](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes) that supports premium storage. This example also shows how to switch to a size that supports premium storage:

PowerShell

**Ensure that the Resource Group name and Disk Name name match yours.**

$diskName = 'myDisk'

# resource group that contains the managed disk

$rgName = 'myResourceGroup2'

# Choose between StandardLRS and PremiumLRS based on your scenario

$storageType = 'Premium\_LRS'

# Premium capable size

$size = 'Standard\_DS2\_v2'

$disk = Get-AzureRmDisk -DiskName $diskName -ResourceGroupName $rgName

# Get parent VM resource

$vmResource = Get-AzureRmResource -ResourceId $disk.ManagedBy

# Stop and deallocate the VM before changing the storage type

Stop-AzureRmVM -ResourceGroupName $vmResource.ResourceGroupName -Name $vmResource.Name -Force

$vm = Get-AzureRmVM -ResourceGroupName $vmResource.ResourceGroupName -Name $vmResource.Name

# Change the VM size to a size that supports premium storage

# Skip this step if converting storage from premium to standard

$vm.HardwareProfile.VmSize = $size

Update-AzureRmVM -VM $vm -ResourceGroupName $rgName

# Update the storage type

$diskUpdateConfig = New-AzureRmDiskUpdateConfig -AccountType $storageType -DiskSizeGB $disk.DiskSizeGB

Update-AzureRmDisk -DiskUpdate $diskUpdateConfig -ResourceGroupName $rgName `

-DiskName $disk.Name

Start-AzureRmVM -ResourceGroupName $vm.ResourceGroupName -Name $vm.Name

**Task 3 Convert a managed disk from standard HDD to standard SSD**

The following example shows how to switch a single disk of a VM from standard HDD to standard SSD, and vice versa:

PowerShell

**Ensure that the Resource Group name and Disk Name name match yours.**

$diskName = 'myDisk'

# resource group that contains the managed disk

$rgName = 'ResourceGroup2'

# Choose between Standard\_LRS and StandardSSD\_LRS based on your scenario

$storageType = 'StandardSSD\_LRS'

$disk = Get-AzureRmDisk -DiskName $diskName -ResourceGroupName $rgName

# Get parent VM resource

$vmResource = Get-AzureRmResource -ResourceId $disk.ManagedBy

# Stop and deallocate the VM before changing the storage type

Stop-AzureRmVM -ResourceGroupName $vmResource.ResourceGroupName -Name $vmResource.Name -Force

$vm = Get-AzureRmVM -ResourceGroupName $vmResource.ResourceGroupName -Name $vmResource.Name

# Update the storage type

$diskUpdateConfig = New-AzureRmDiskUpdateConfig -AccountType $storageType -DiskSizeGB $disk.DiskSizeGB

Update-AzureRmDisk -DiskUpdate $diskUpdateConfig -ResourceGroupName $rgName `

-DiskName $disk.Name

Start-AzureRmVM -ResourceGroupName $vm.ResourceGroupName -Name $vm.Name

**Exercise 4 Convert a Windows virtual machine from unmanaged disks to managed disks**

**Task 1 Convert using the Azure portal**

1. Sign in to the [Azure portal](https://portal.azure.com/).
2. Select the VM from the list of VMs in the portal.
3. In the blade for the VM, select **Disks** from the menu.
4. At the top of the **Disks** blade, select **Migrate to managed disks**.
5. If your VM is in an availability set, there will be a warning on the **Migrate to managed disks** blade that you need to convert the availability set first. The warning should have a link you can click to convert the availability set. Once the availability set is converted or if your VM is not in an availability set, click **Migrate** to start the process of migrating your disks to managed disks.

The VM will be stopped and restarted after migration is complete.

**Lab 5A** **Create and deploy highly available virtual machines with Azure PowerShell**

* ‎In this Lab, you learn how to increase the availability and reliability of your Virtual Machine solutions on Azure using a capability called Availability Sets. Availability sets ensure that the VMs you deploy on Azure are distributed across multiple isolated hardware nodes in a cluster. Doing this ensures that if a hardware or software failure within Azure happens, only a subset of your VMs are impacted and that your overall solution remains available and operational.

In this tutorial, you learn how to:

* Create an availability set
* Create a VM in an availability set
* Check available VM sizes
* Check Azure Advisor

**Task 1 Launch Azure Cloud Shell**

|  |  |
| --- | --- |
| Open Cloud Shell in your browser. | <https://shell.azure.com/powershell> |
| Click the **Cloud Shell** button on the menu in the upper right of the Azure portal. | [Cloud Shell in the portal](https://portal.azure.com/) |
|  |  |

If you choose to install and use the PowerShell locally, this tutorial requires the Azure PowerShell module version 5.7.0 or later. Run Get-Module -ListAvailable AzureRM to find the version. If you need to upgrade, see [Install Azure PowerShell module](https://docs.microsoft.com/en-us/powershell/azure/install-azurerm-ps). If you are running PowerShell locally, you also need to run Connect-AzureRmAccount to create a connection with Azure.

**Task 2 Create an availability set**

You can create an availability set using [New-AzureRmAvailabilitySet](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermavailabilityset). In this example, set both the number of update and fault domains at *2* for the availability set named *myAvailabilitySet* in the *myResourceGroupAvailability* resource group.

Create a resource group.

Azure PowerShell

New-AzureRmResourceGroup -Name myResourceGroupAvailability -Location EastUS

Create a managed availability set using [New-AzureRmAvailabilitySet](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermavailabilityset) with the -sku aligned parameter.

Azure PowerShell

New-AzureRmAvailabilitySet `

-Location "EastUS" `

-Name "myAvailabilitySet" `

-ResourceGroupName "myResourceGroupAvailability" `

-Sku aligned `

-PlatformFaultDomainCount 2 `

-PlatformUpdateDomainCount 2

**Task 3 Create VMs inside an availability set**

VMs must be created within the availability set to make sure they are correctly distributed across the hardware. You can't add an existing VM to an availability set after it is created.

The hardware in a location is divided in to multiple update domains and fault domains. An **update domain** is a group of VMs and underlying physical hardware that can be rebooted at the same time. VMs in the same **fault domain** share common storage as well as a common power source and network switch.

When you create a VM with [New-AzureRmVM](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermvm), you use the -AvailabilitySetName parameter to specify the name of the availability set.

1. First, set an administrator username and password for the VM with [Get-Credential](https://msdn.microsoft.com/powershell/reference/5.1/microsoft.powershell.security/Get-Credential): use **azureuser** for user name and **Pa55w.rd1234** for the pwd to keep it standard through out the course.

PowerShell

$cred = Get-Credential

2. Now create two VMs with [New-AzureRmVM](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermvm) in the availability set.

PowerShell

for ($i=1; $i -le 2; $i++)

{

New-AzureRmVm `

-ResourceGroupName "myResourceGroupAvailability" `

-Name "myVM$i" `

-Location "East US" `

-VirtualNetworkName "myVnet" `

-SubnetName "mySubnet" `

-SecurityGroupName "myNetworkSecurityGroup" `

-PublicIpAddressName "myPublicIpAddress$i" `

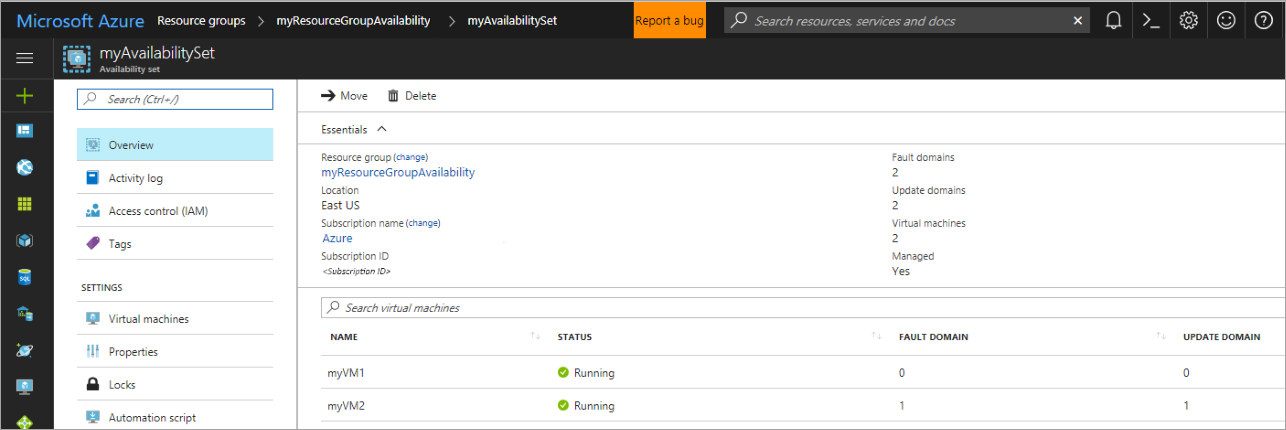
-AvailabilitySetName "myAvailabilitySet" `

-Credential $cred

}

The -AsJob parameter creates the VM as a background task, so the PowerShell prompts return to you. You can view details of background jobs with the Job cmdlet. It takes a few minutes to create and configure both VMs. When finished, you have two virtual machines distributed across the underlying hardware.

If you look at the availability set in the portal by going to Resource Groups > myResourceGroupAvailability > myAvailabilitySet, you should see how the VMs are distributed across the two fault and update domains.



**Task 4 Check for available VM sizes**

You can add more VMs to the availability set later, but you need to know what VM sizes are available on the hardware. Use [Get-AzureRMVMSize](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/get-azurermvmsize) to list all the available sizes on the hardware cluster for the availability set.

Azure PowerShell Copy Try It

Get-AzureRmVMSize `

-ResourceGroupName "myResourceGroupAvailability" `

-AvailabilitySetName "myAvailabilitySet"

**Lab 5B** **Create a virtual machine scale set in the Azure portal**

* ‎A virtual machine scale set allows you to deploy and manage a set of identical, auto-scaling virtual machines. You can scale the number of VMs in the scale set manually, or define rules to autoscale based on resource usage like CPU, memory demand, or network traffic. An Azure load balancer then distributes traffic to the VM instances in the scale set. In this quickstart, you create a virtual machine scale set in the Azure portal.

If you don't have an Azure subscription, create a [free account](https://azure.microsoft.com/free/?WT.mc_id=A261C142F) before you begin.

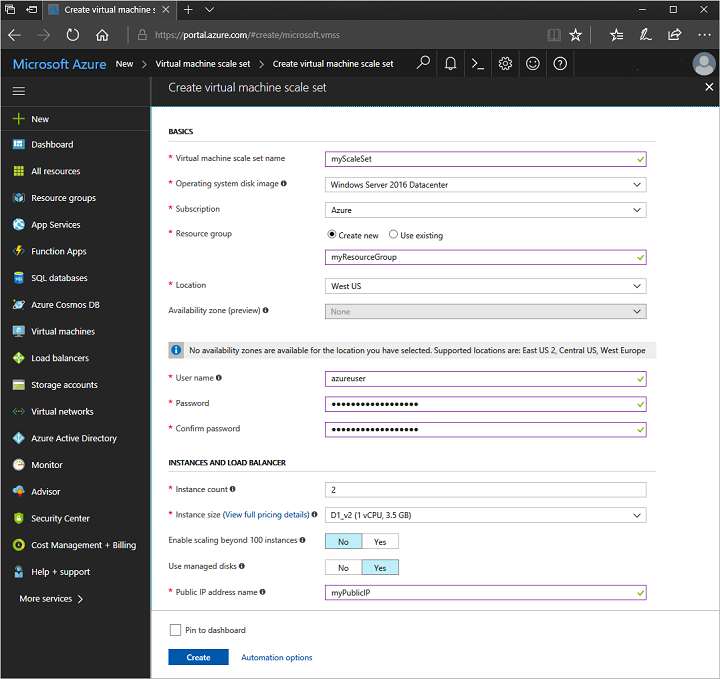
**Task 1 Log in to Azure**

Log in to the Azure portal at [http://portal.azure.com](http://portal.azure.com/).

**Task 2 Create virtual machine scale set**

You can deploy a scale set with a Windows Server image or Linux image such as RHEL, CentOS, Ubuntu, or SLES.

1. Click **Create a resource** in the upper left-hand corner of the Azure portal.
2. Search for *scale set*, choose **Virtual machine scale set**, then select **Create**.
3. Enter a name for the scale set, such as *myScaleSet*.
4. Select your desired OS type, such as *Windows Server 2016 Datacenter*.
5. Enter your desired resource group name, such as *myResourceGroup*, and location, such as *East US*.
6. Enter your desired username, and select which authentication type you prefer.
   * A **Password** must be at least 12 characters long and meet three out of the four following complexity requirements: one lower case character, one upper case character, one number, and one special character. For more information, see [username and password requirements](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/faq#what-are-the-username-requirements-when-creating-a-vm).
   * If you select a Linux OS disk image, you can instead choose **SSH public key**. Only provide your public key, such as *~/.ssh/id\_rsa.pub*. You can use the Azure Cloud Shell from the portal to [create and use SSH keys](https://docs.microsoft.com/en-us/azure/virtual-machines/linux/mac-create-ssh-keys).
7. Enter a **Public IP address name**, such as *myPublicIP*.
8. Enter a unique **Domain name label**, such as *myuniquedns*. This DNS label forms the base of the FQDN for the load balancer in front of the scale set.
9. To confirm the scale set options, select **Create**.

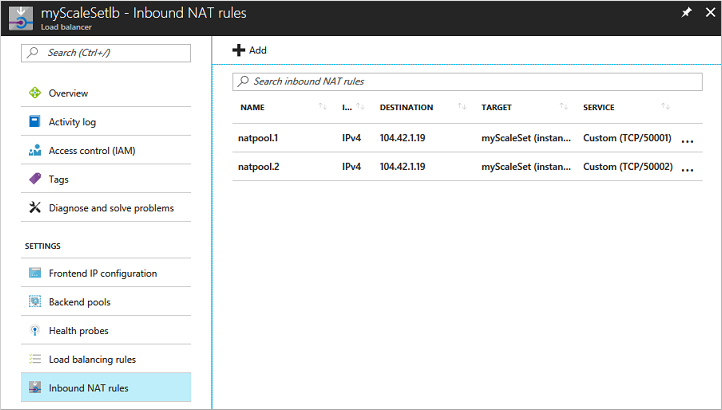


**Task 3 Connect to a VM in the scale set**

When you create a scale set in the portal, a load balancer is created. Network Address Translation (NAT) rules are used to distribute traffic to the scale set instances for remote connectivity such as RDP or SSH.

To view these NAT rules and connection information for your scale set instances:

1. Select the resource group you created in the previous step, such as *myResourceGroup*.
2. From the list of resources, select your **Load balancer**, such as *myScaleSetLab*.
3. Choose **Inbound NAT rules** from the menu on the left-hand side of the window.



You can connect to each VM in the scale set using these NAT rules. Each VM instance lists a destination IP address and TCP port value. For example, if the destination IP address is *104.42.1.19* and the TCP port is *50001*, you connect to the VM instance as follows:

* For a Windows scale set, connect to the VM instance with RDP on 104.42.1.19:50001
* For a Linux scale set, connect to the VM instance with SSH on ssh azureuser@104.42.1.19 -p 50001

When prompted, enter the credentials you specified from the previous step when you created the scale set. The scale set instances are regular VMs that you can interact with as normal. For more information on how to deploy and run applications on your scale set instances, see [Deploy your application on virtual machine scale sets](https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/virtual-machine-scale-sets-deploy-app)

**Lab 5C** **Automatically scale a virtual machine scale set in the Azure portal**

* ‎When you create a scale set, you define the number of VM instances that you wish to run. As your application demand changes, you can automatically increase or decrease the number of VM instances. The ability to autoscale lets you keep up with customer demand or respond to application performance changes throughout the lifecycle of your app.

This article shows you how to create autoscale rules in the Azure portal that monitor the performance of the VM instances in your scale set. These autoscale rules increase or decrease the number of VM instances in response to these performance metrics. You can also complete these steps with [Azure PowerShell](https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/tutorial-autoscale-powershell) or the [Azure CLI](https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/tutorial-autoscale-cli).

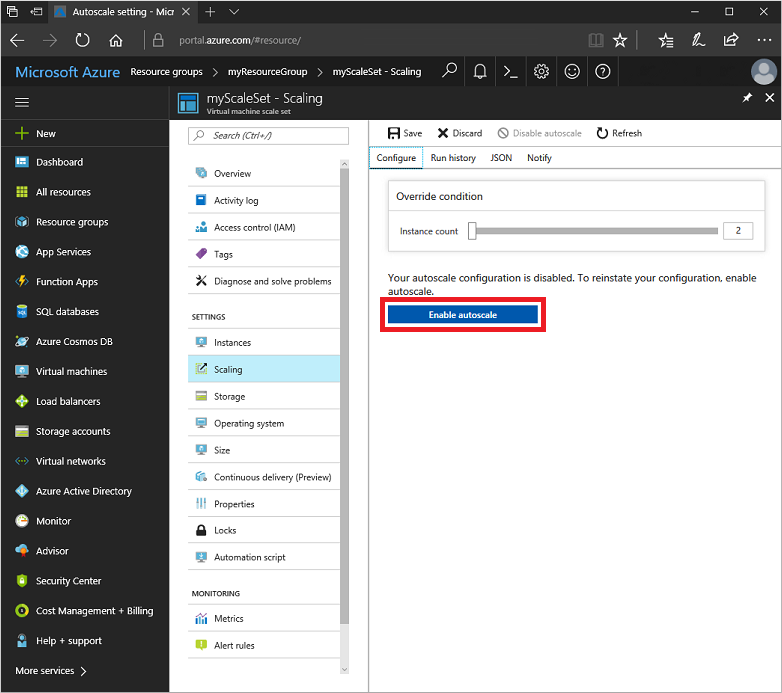
**Prerequisites**

To create autoscale rules, you need an existing virtual machine scale set. You can create a scale set with the [Azure portal](https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/quick-create-portal), [Azure PowerShell](https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/quick-create-powershell), or [Azure CLI](https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/quick-create-cli).

**Task 1 Create a rule to automatically scale out**

If your application demand increases, the load on the VM instances in your scale set increases. If this increased load is consistent, rather than just a brief demand, you can configure autoscale rules to increase the number of VM instances in the scale set. When these VM instances are created and your applications are deployed, the scale set starts to distribute traffic to them through the load balancer. You control what metrics to monitor, such as CPU or disk, how long the application load must meet a given threshold, and how many VM instances to add to the scale set.

1. Open the Azure portal and select **Resource groups** from the menu on the left-hand side of the dashboard.
2. Select the resource group that contains your scale set, then choose your scale set from the list of resources.
3. Choose **Scaling** from the menu on the left-hand side of the scale set window. Select the button to **Enable autoscale**:

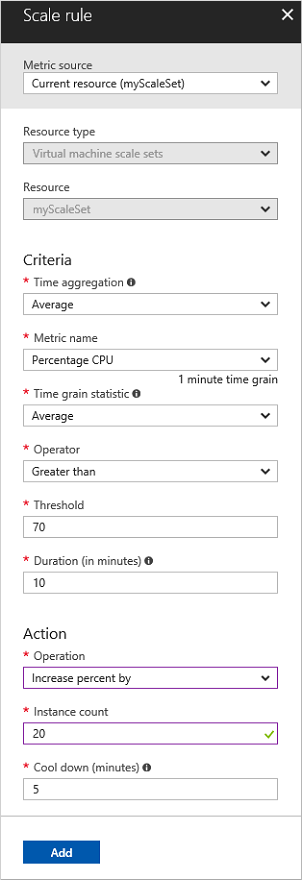


1. Enter a name for your settings, such as *autoscale*, then select the option to **Add a rule**.
2. Let's create a rule that increases the number of VM instances in a scale set when the average CPU load is greater than 70% over a 10-minute period. When the rule triggers, the number of VM instances is increased by 20%. In scale sets with a small number of VM instances, you could set the **Operation** to *Increase count by* and then specify *1* or *2* for the *Instance count*. In scale sets with a large number of VM instances, an increase of 10% or 20% VM instances may be more appropriate.

Specify the following settings for your rule:

| **Parameter** | **Explanation** | **Value** |
| --- | --- | --- |
| *Time Aggregation* | Defines how the collected metrics should be aggregated for analysis. | Average |
| *Metric Name* | The performance metric to monitor and apply scale set actions on. | Percentage CPU |
| *Time grain statistic* | Defines how the collected metrics in each time grain should be aggregated for analysis. | Average |
| *Operator* | Operator used to compare the metric data against the threshold. | Greater than |
| *Threshold* | The percentage that causes the autoscale rule to trigger an action. | 70 |
| *Duration* | The amount of time monitored before the metric and threshold values are compared. | 10 minutes |
| *Operation* | Defines if the scale set should scale up or down when the rule applies and by what increment | Increase percent by |
| *Instance count* | The percentage of VM instances should be changed when the rule triggers. | 20 |
| *Cool down (minutes)* | The amount of time to wait before the rule is applied again so that the autoscale actions have time to take effect. | 5 minutes |

The following examples show a rule created in the Azure portal that matches these settings:



1. To create the rule, select **Add**

**Task 2 Create a rule to automatically scale in**

On an evening or weekend, your application demand may decrease. If this decreased load is consistent over a period of time, you can configure autoscale rules to decrease the number of VM instances in the scale set. This scale-in action reduces the cost to run your scale set as you only run the number of instances required to meet the current demand.

1. Choose to **Add a rule** again.
2. Create a rule that decreases the number of VM instances in a scale set when the average CPU load then drops below 30% over a 10-minute period. When the rule triggers, the number of VM instances is decreased by 20%.

Use the same approach as with the previous rule. Adjust the following settings for your rule:

| **Parameter** | **Explanation** | **Value** |
| --- | --- | --- |
| *Operator* | Operator used to compare the metric data against the threshold. | Less than |
| *Threshold* | The percentage that causes the autoscale rule to trigger an action. | 30 |
| *Operation* | Defines if the scale set should scale up or down when the rule applies and by what increment | Decrease percent by |
| *Instance count* | The percentage of VM instances should be changed when the rule triggers. | 20 |

1. To create the rule, select **Add**

**Task 3 Define autoscale instance limits**

Your autoscale profile must define a minimum, maximum, and default number of VM instances. When your autoscale rules are applied, these instance limits make sure that you do not scale out beyond the maximum number of instances, or scale in beyond the minimum of instances.

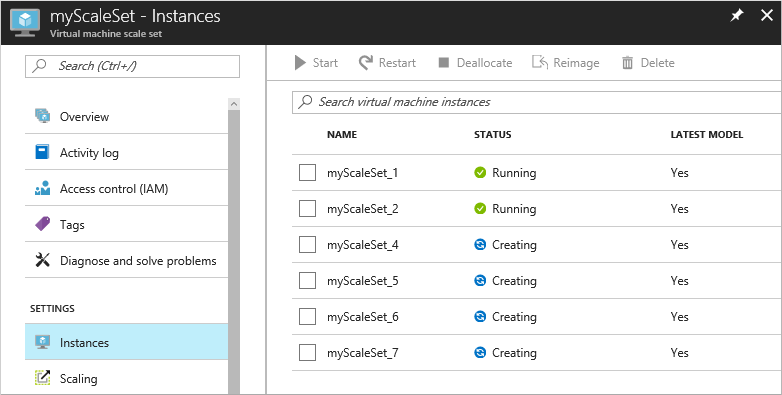
1. Set the following instance limits:

| **Minimum** | **Maximum** | **Default** |
| --- | --- | --- |
| 2 | 10 | 2 |

1. To apply your autoscale rules and instance limits, select **Save**.

**Task 4 Monitor number of instances in a scale set**

To see the number and status of VM instances, select **Instances** from the menu on the left-hand side of the scale set window. The status indicates if the VM instance is *Creating* as the scale set automatically scales out, or is *Deleting* as the scale automatically scales in.



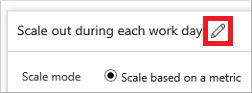
**Task 5 Autoscale based on a schedule**

The previous examples automatically scaled a scale set in or out with basic host metrics such as CPU usage. You can also create autoscale rules based on schedules. These schedule-based rules allow you to automatically scale out the number of VM instances ahead of an anticipated increase in application demand, such as core work hours, and then automatically scale in the number of instances at a time that you anticipate less demand, such as the weekend.

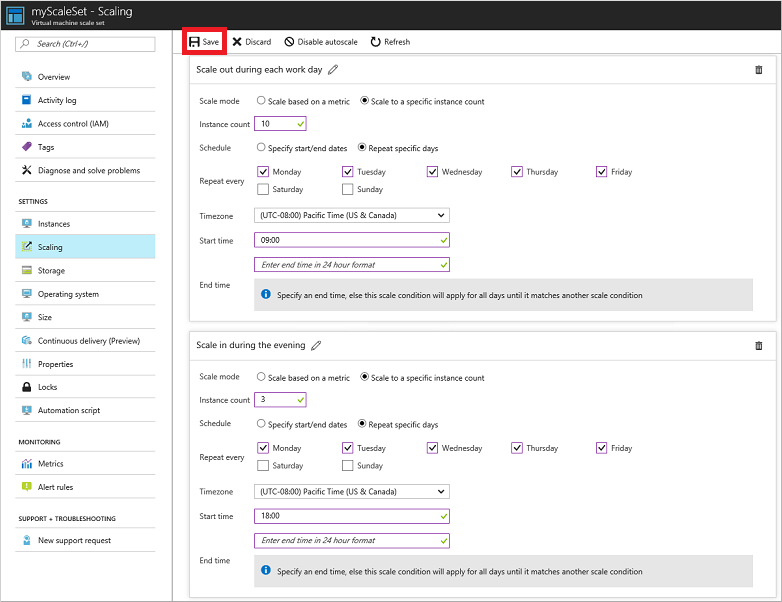
1. Choose **Scaling** from the menu on the left-hand side of the scale set window. To delete the existing autoscale rules created in the previous examples, choose the trash can icon.

Delete the existing autoscale rules

1. Choose to **Add a scale condition**. Select the pencil icon next to rule name, and provide a name such as *Scale out during each work day*.



1. Select the radio button to **Scale to a specific instance count**.
2. To scale up the number of instances, enter *10* as the instance count.
3. Choose **Repeat specific days** for the **Schedule** type.
4. Select all the work days, Monday through Friday.
5. Choose the appropriate timezone, then specify a **Start time** of *09:00*.
6. Choose to **Add a scale condition** again. Repeat the process to create a schedule named *Scale in during the evening* that scales to *3* instances, repeats every weekday, and starts at *18:00*.
7. To apply your schedule-based autoscale rules, select **Save**.



To see how your autoscale rules are applied, select **Run history** across the top of the **Scaling** window. The graph and events list shows when the autoscale rules trigger and the number of VM instances in your scale set increases or decreases.

**Lab 5D** **Deploy applications to a Windows virtual machine in Azure with the Custom Script Extension**

* ‎To configure virtual machines (VMs) in a quick and consistent manner, some form of automation is typically desired. A common approach to customize a Windows VM is to use [Custom Script Extension for Windows](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/extensions-customscript). In this tutorial you learn how to:
* Use the Custom Script Extension to install IIS
* Create a VM that uses the Custom Script Extension
* View a running IIS site after the extension is applied

**Task 1 Launch Azure Cloud Shell**

|  |  |
| --- | --- |
| Open Cloud Shell in your browser. | <https://shell.azure.com/powershell> |
| Click the **Cloud Shell** button on the menu in the upper right of the Azure portal. | [Cloud Shell in the portal](https://portal.azure.com/) |
|  |  |

If you choose to install and use the PowerShell locally, this tutorial requires the Azure PowerShell module version 5.7.0 or later. Run Get-Module -ListAvailable AzureRM to find the version. If you need to upgrade, see [Install Azure PowerShell module](https://docs.microsoft.com/en-us/powershell/azure/install-azurerm-ps). If you are running PowerShell locally, you also need to run Connect-AzureRmAccount to create a connection with Azure.

**Custom script extension overview**

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.

The Custom Script extension integrates with Azure Resource Manager templates, and can also be run using the Azure CLI, PowerShell, Azure portal, or the Azure Virtual Machine REST API.

You can use the Custom Script Extension with both Windows and Linux VMs.

**Task 2 Create virtual machine**

* + 1. First, set an administrator username and password for the VM with [Get-Credential](https://msdn.microsoft.com/powershell/reference/5.1/microsoft.powershell.security/Get-Credential):

PowerShell

$cred = Get-Credential

* + 1. Now you can create the VM with [New-AzureRmVM](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermvm). The following example creates a VM named *myVM* in the *EastUS* location. If they do not already exist, the resource group *myResourceGroupAutomate* and supporting network resources are created. To allow web traffic, the cmdlet also opens port *80*.

PowerShell

New-AzureRmVm `

-ResourceGroupName "myResourceGroupAutomate" `

-Name "myVM" `

-Location "East US" `

-VirtualNetworkName "myVnet" `

-SubnetName "mySubnet" `

-SecurityGroupName "myNetworkSecurityGroup" `

-PublicIpAddressName "myPublicIpAddress" `

-OpenPorts 80 `

-Credential $cred

It takes a few minutes for the resources and VM to be created.

**Task 3 Automate IIS install**

Use [Set-AzureRmVMExtension](https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/set-azurermvmextension) to install the Custom Script Extension. The extension runs powershell Add-WindowsFeature Web-Server to install the IIS webserver and then updates the *Default.htm* page to show the hostname of the VM:

PowerShell

Set-AzureRmVMExtension -ResourceGroupName "myResourceGroupAutomate" `

-ExtensionName "IIS" `

-VMName "myVM" `

-Location "EastUS" `

-Publisher Microsoft.Compute `

-ExtensionType CustomScriptExtension `

-TypeHandlerVersion 1.8 `

-SettingString '{"commandToExecute":"powershell Add-WindowsFeature Web-Server; powershell Add-Content -Path \"C:\\inetpub\\wwwroot\\Default.htm\" -Value $($env:computername)"}'

**Task 4 Test web site**

Obtain the public IP address of your load balancer with [Get-AzureRmPublicIPAddress](https://docs.microsoft.com/en-us/powershell/module/azurerm.network/get-azurermpublicipaddress). The following example obtains the IP address for *myPublicIPAddress* created earlier:

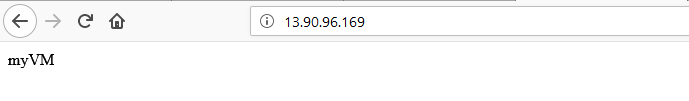
Azure PowerShell Copy Try It

Get-AzureRmPublicIPAddress `

-ResourceGroupName "myResourceGroupAutomate" `

-Name "myPublicIPAddress" | select IpAddress

You can then enter the public IP address in to a web browser. The website is displayed, including the hostname of the VM that the load balancer distributed traffic to as in the following example:



**Lab 6A** **Back up and restore files for Windows virtual machines in Azure**

* ‎You can protect your data by taking backups at regular intervals. Azure Backup creates recovery points that are stored in geo-redundant recovery vaults. When you restore from a recovery point, you can restore the whole VM or specific files. This article explains how to restore a single file to a VM running Windows Server and IIS. If you don't already have a VM to use, you can create one using the [Windows quickstart](https://docs.microsoft.com/en-us/azure/virtual-machines/windows/quick-create-portal). In this tutorial you learn how to:
* Create a backup of a VM
* Schedule a daily backup
* Restore a file from a backup

**Backup overview**

When the Azure Backup service initiates a backup job, it triggers the backup extension to take a point-in-time snapshot. The Azure Backup service uses the *VMSnapshot* extension. The extension is installed during the first VM backup if the VM is running. If the VM is not running, the Backup service takes a snapshot of the underlying storage (since no application writes occur while the VM is stopped).

When taking a snapshot of Windows VMs, the Backup service coordinates with the Volume Shadow Copy Service (VSS) to get a consistent snapshot of the virtual machine's disks. Once the Azure Backup service takes the snapshot, the data is transferred to the vault. To maximize efficiency, the service identifies and transfers only the blocks of data that have changed since the previous backup.

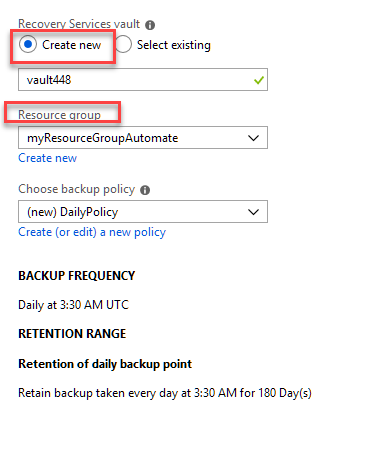
When the data transfer is complete, the snapshot is removed and a recovery point is created.

**Task 1 Create a backup**

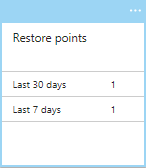
Create a simple scheduled daily backup to a Recovery Services Vault.

1. Sign in to the [Azure portal](https://portal.azure.com/).
2. In the menu on the left, select **Virtual machines**.
3. From the list, select the **myVM** from the **myResourceGroupAutomate**
4. On the VM blade, in the **Operations** section, click **Backup**. Then select **Enable Backup** using the values that match your Resource group.

**Note** We are using the default policy. If you would like you can create a new policy.



1. To create an initial recovery point, on the **Backup** blade click **Backup now**.
2. On the **Backup Now** blade, click the calendar icon, use the calendar control to select the last day this recovery point is retained, and click **OK**.
3. In the **Backup** blade for your VM, you see the number of recovery points that are complete.



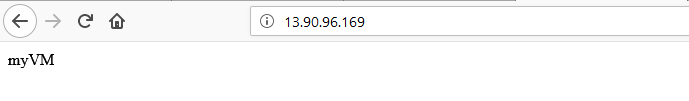
The first backup takes about 20 minutes. Proceed to the next part of this lab **after** your backup is finished.

**Task 2 Recover a file**

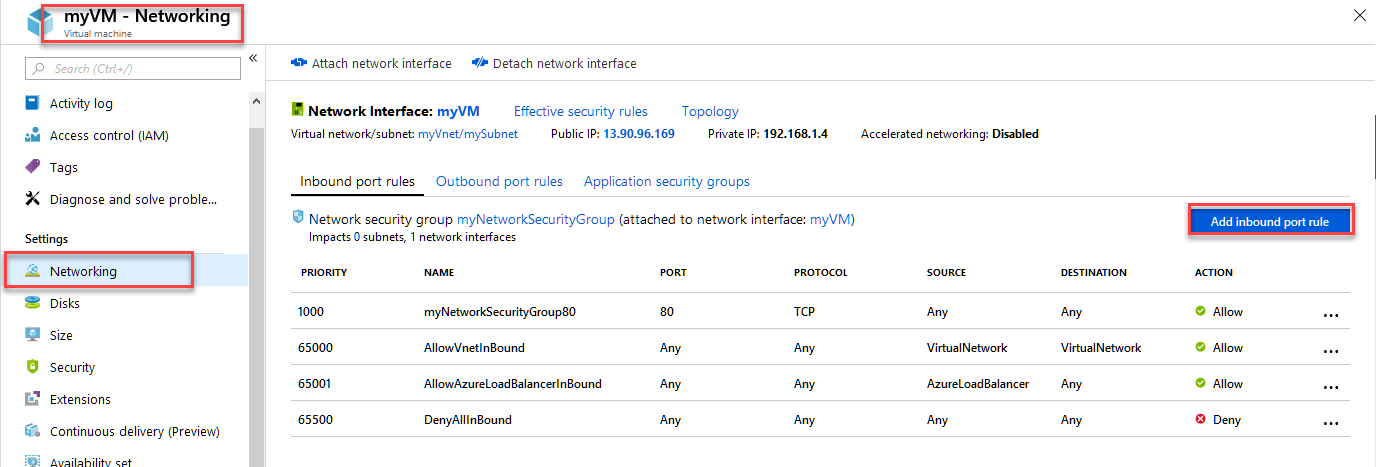
If you accidentally delete or make changes to a file, you can use File Recovery to recover the file from your backup vault. File Recovery uses a script that runs on the VM, to mount the recovery point as local drive. These drives remain mounted for 12 hours so that you can copy files from the recovery point and restore them to the VM.

In this example, we show how to recover the image file that is used in the default web page for IIS.

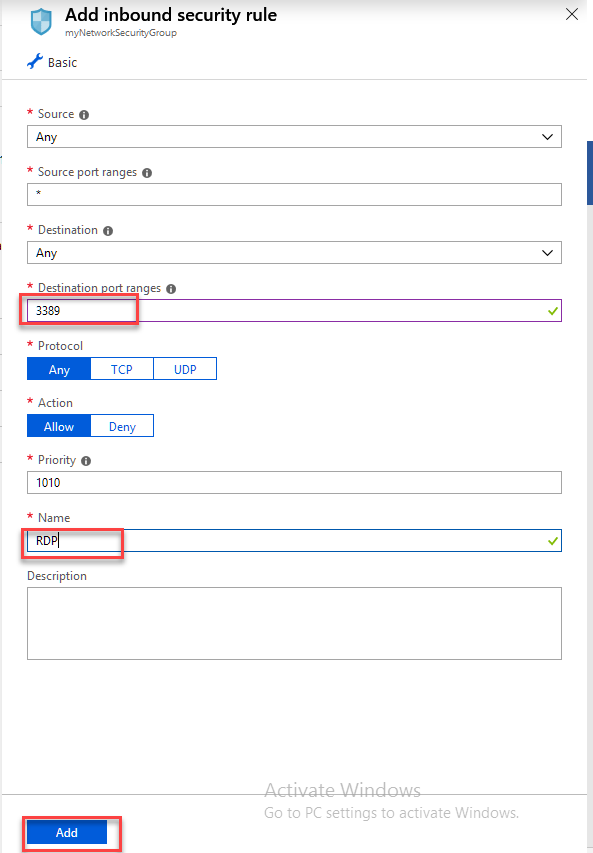
1. Open a browser and connect to the IP address of the VM to show the default IIS page.



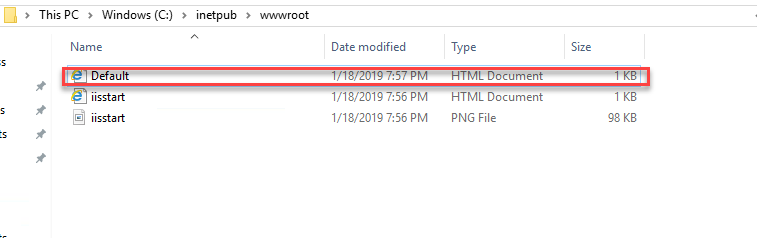
1. Connect to the VM. You may need to create a NetworkSecurtyGroup rule for port 3389.
   * 1. From the Networking Blade select Add inbound port rule



* + 1. Then Change to the port to 3389
    2. Change the name to RDP
    3. Then Click add



1. On the VM, open **File Explorer** and navigate to \inetpub\wwwroot and delete the file highlighted. If the file hightlighted is not there, based on the VM you choose then delete the png file.



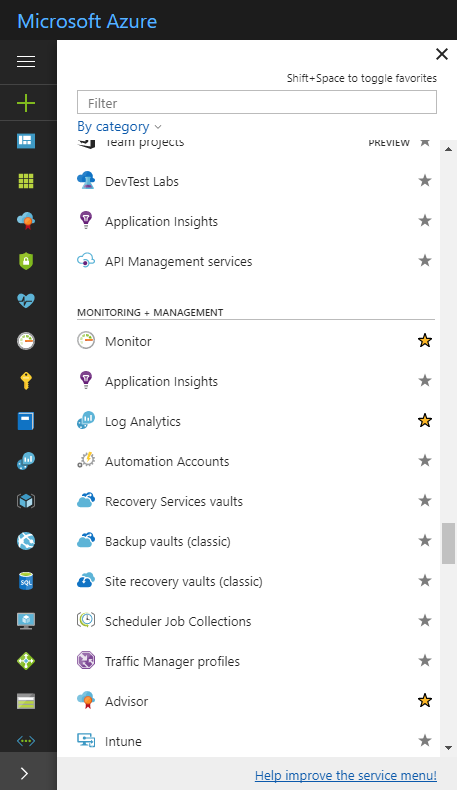
1. On your local computer, refresh the browser to see the page has changed because of the deletion.
2. On your local computer, open a new tab and go the [Azure portal](https://portal.azure.com/).
3. In the menu on the left, select **Virtual machines** and select the VM from the list.
4. On the VM blade, in the **Settings** section, click **Backup**. The **Backup** blade opens.
5. In the menu at the top of the blade, select **File Recovery**. The **File Recovery** blade opens.
6. In **Step 1: Select recovery point**, select a recovery point from the drop-down.
7. In **Step 2: Download script to browse and recover files**, click the **Download Executable** button. Save the file to your **Downloads** folder.
8. On your local computer, open **File Explorer** and navigate to your **Downloads** folder and copy the downloaded .exe file. The filename is prefixed by your VM name.
9. On your VM (over the RDP connection) paste the .exe file to the Desktop of your VM.
10. Navigate to the desktop of your VM and double-click on the .exe. This launches a command prompt and then mount the recovery point as a file share that you can access. When it is finished creating the share, type **q** to close the command prompt.
11. On your VM, open **File Explorer** and navigate to the drive letter that was used for the file share.
12. Navigate to \inetpub\wwwroot and copy the deleted file from the file share and paste it into \inetpub\wwwroot.
13. On your local computer, open the browser tab where you are connected to the IP address of the VM showing the IIS default page. Press CTRL + F5 to refresh the browser page. You should now see that the image has been restored.
14. On your local computer, go back to the browser tab for the Azure portal and in **Step 3: Unmount the disks after recovery** click the **Unmount Disks** button. If you forget to do this step, the connection to the mountpoint is automatically close after 12 hours. After those 12 hours, you need to download a new script to create a new mountpoint.

**Lab 6B** **Get started with Azure Advisor**

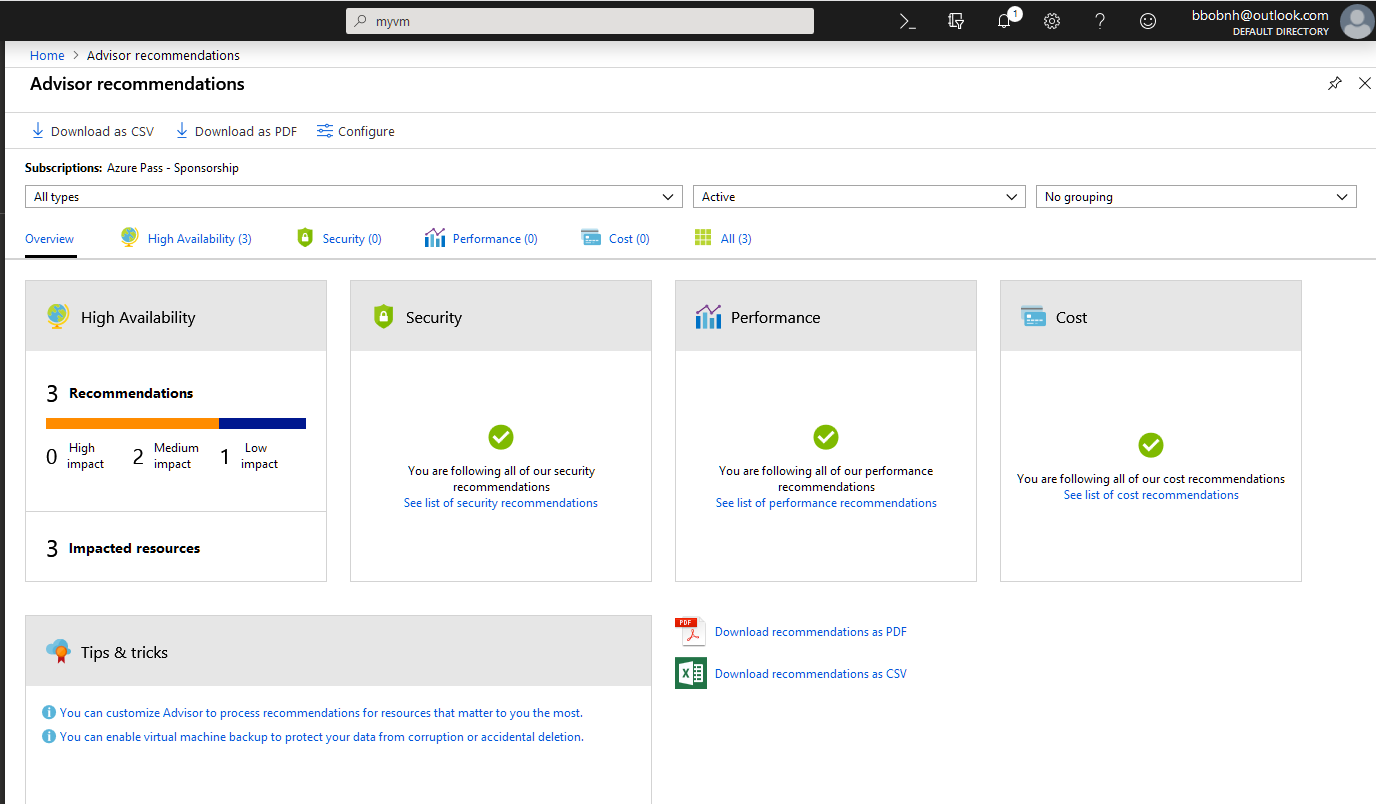
* ‎Learn how to access Advisor through the Azure portal, get recommendations, and implement recommendations.

**Task 1 Get Advisor recommendations**

1. Sign in to the [Azure portal](https://portal.azure.com/).
2. In the left pane, click **Advisor**. If you do not see Advisor in the left pane, click **All services**. In the service menu pane, under **Monitoring and Management**, click **Advisor**. The Advisor dashboard is displayed.



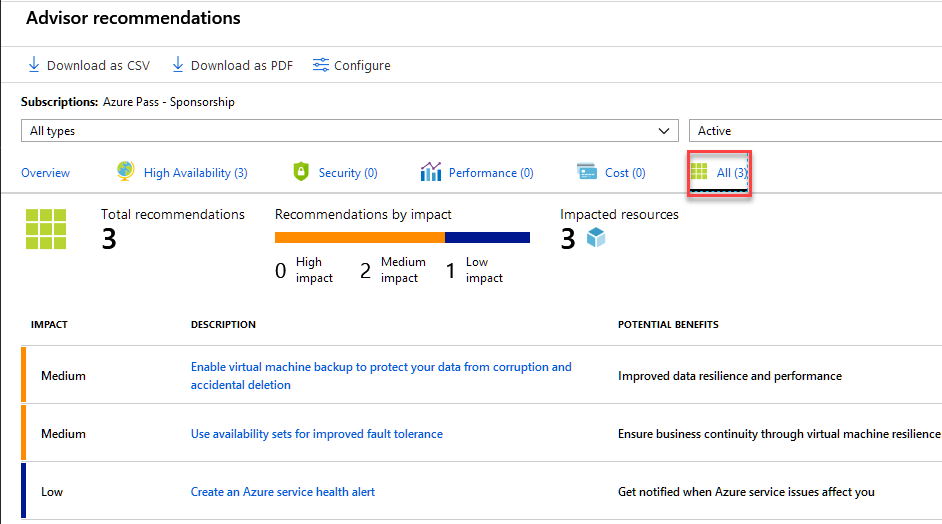
1. The Advisor dashboard will display a summary of your recommendations for all selected subscriptions. You can choose the subscriptions that you want recommendations to be displayed for using the subscription filter dropdown.
2. To get recommendations for a specific category, click one of the tabs: **High Availability**, **Security**, **Performance**, or **Cost**.



**Task 2 Get Advisor recommendation details, and implement a solution**

You can select a recommendation in Advisor to view additional details – such as the recommendation actions and impacted resources – and to implement the solution to the recommendation.

1. Select a recommendation category to display the list of recommendations within that category, or select the **All** tab to view all your recommendations.



1. Click a recommendation that you want to review in detail.
2. Review the information about the recommendation and the resources that the recommendation applies to.
3. Click on the **Recommended Action** to implement the recommendation.

**Task 3 Filter Advisor recommendations**

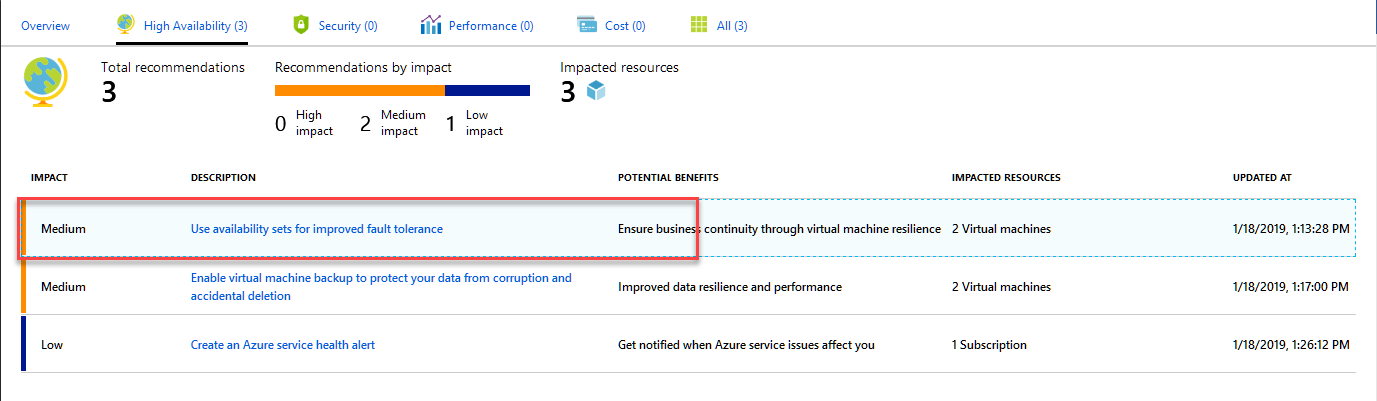
You can filter recommendations to drill down to what is most important to you. You can filter by subscription, resource type, or recommendation status.

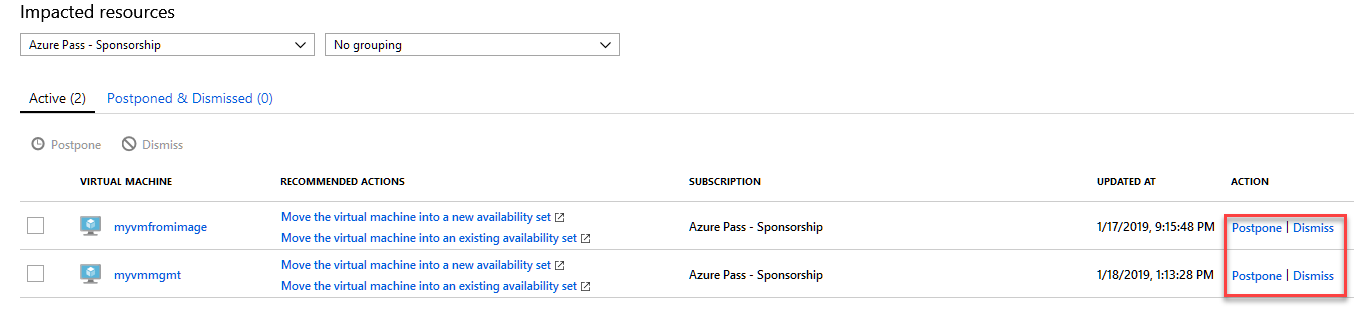
1. Sign in to the [Azure portal](https://portal.azure.com/), and then open [Advisor](https://aka.ms/azureadvisordashboard).
2. Use the dropdowns on the Advisor dashboard to filter by subscription, resource type, or recommendation status.



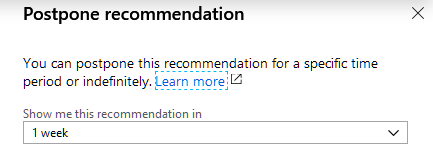
**Task 4 Postpone or dismiss Advisor recommendations**

1. Navigate to the recommendation you want to postpone or dismiss.





1. Click the recommendation.
2. Click **Postpone**.



1. Specify a postpone time period.

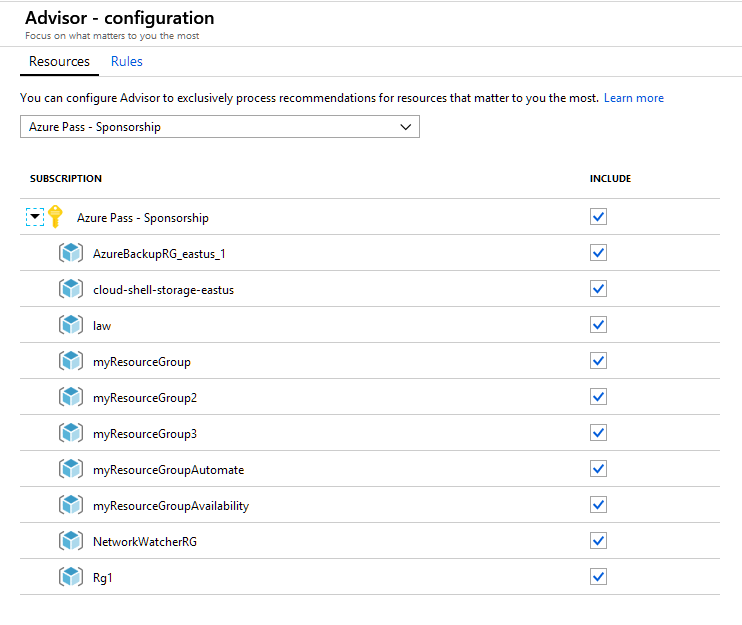
**Task 5 Exclude subscriptions or resource groups from Advisor**

You may have resource groups or subscriptions for which you do not want to receive Advisor recommendations – such as ‘test’ resources. You can configure Advisor to only generate recommendations for specific subscriptions and resource groups.

**Note**

To include or exclude a subscription or resource group from Advisor, you must be a subscription Owner. If you do not have the required permissions for a subscription or resource group, the option to include or exclude it is disabled in the user interface.

1. Sign in to the [Azure portal](https://portal.azure.com/), and then open [Advisor](https://aka.ms/azureadvisordashboard).
2. Click **Configure** in the action bar.
3. Uncheck any subscriptions or resource groups you do not want to receive Advisor recommendations for.



1. Click the **Apply** button.

**Task 5 Configure the average CPU utilization rule for the low usage virtual machine recommendation**

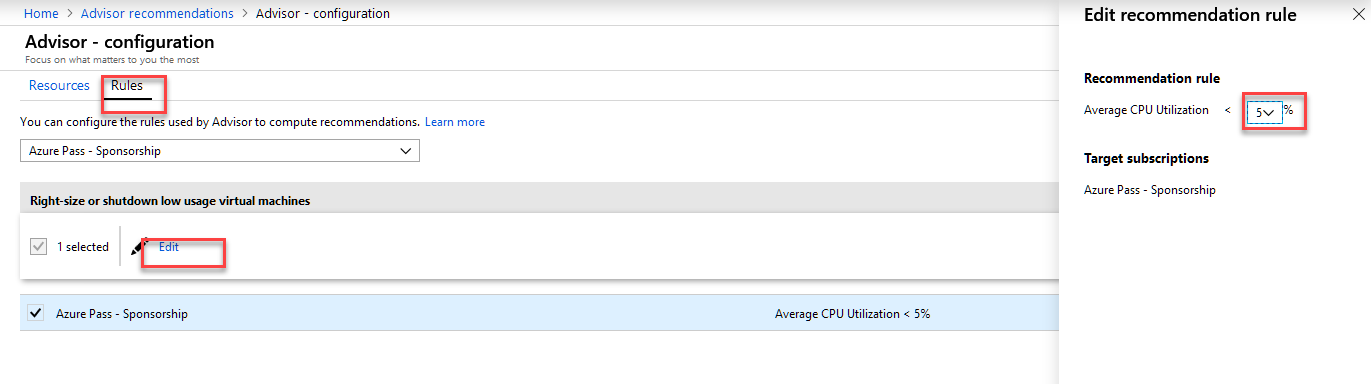
Advisor monitors your virtual machine usage for 14 days and then identifies low-utilization virtual machines. Virtual machines whose average CPU utilization is 5 percent or less and network usage is 7 MB or less for four or more days are considered low-utilization virtual machines.

If you would like to be more aggressive at identifying low usage virtual machines, you can adjust the average CPU utilization rule on a per subscription basis. The average CPU utilization rule can be set to 5%, 10%, 15%, or 20%.

Note

To adjust the average CPU utilization rule for identifying low usage virtual machines, you must be a subscription *Owner*. If you do not have the required permissions for a subscription or resource group, the option to include or exclude it will be disabled in the user interface.

1. Sign in to the [Azure portal](https://portal.azure.com/), and then open [Advisor](https://aka.ms/azureadvisordashboard).
2. Click **Configure** in the action bar.
3. Click the **Rules** tab.
4. Select the subscriptions you’d like to adjust the average CPU utilization rule for, and then click **Edit**.
5. Select the desired average CPU utilization value, and click **Apply**.
6. Click **Refresh recommendations** to update your existing recommendations to use the new average CPU utilization rule.



**Task 6 Download your Advisor recommendations**

Advisor enables you to download a summary of your recommendations. You can download your recommendations as a PDF file or a CSV file. Downloading your recommendations enables you to easily share with your colleagues or perform your own analysis on top of the recommendation data.

1. Sign in to the [Azure portal](https://portal.azure.com/), and then open [Advisor](https://aka.ms/azureadvisordashboard).
2. Click **Download as CSV** or **Download as PDF** on the action bar.

The download option respects any filters you have applied to the Advisor dashboard. If you select the download option while viewing a specific recommendation category or recommendation, the downloaded summary only includes information for that category or recommendation.

**Task 7 Clean up resources**

When you're done with the virtual network, and the VMs, delete the resource group and all of the resources it contains:

Delete all Resource groups except your **cloud-shell**

