**Quickstart: Create a Windows virtual machine in Azure with PowerShell**

**Applies to:** ✔️ Windows VMs

The Azure PowerShell module is used to create and manage Azure resources from the PowerShell command line or in scripts. This quickstart shows you how to use the Azure PowerShell module to deploy a virtual machine (VM) in Azure that runs Windows Server 2016. You will also RDP to the VM and install the IIS web server, to show the VM in action.

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

**Launch Azure Cloud Shell**

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

**Create resource group**

Create an Azure resource group with [New-AzResourceGroup](https://docs.microsoft.com/en-us/powershell/module/az.resources/new-azresourcegroup). A resource group is a logical container into which Azure resources are deployed and managed.

Azure PowerShell

New-AzResourceGroup -Name myResourceGroup -Location EastUS

**Create virtual machine**

Create a VM with [New-AzVM](https://docs.microsoft.com/en-us/powershell/module/az.compute/new-azvm). Provide names for each of the resources and the New-AzVM cmdlet creates if they don't already exist.

When prompted, provide a username and password to be used as the sign-in credentials for the VM:

Azure PowerShell

New-AzVm `

-ResourceGroupName "myResourceGroup" `

-Name "myVM" `

-Location "East US" `

-VirtualNetworkName "myVnet" `

-SubnetName "mySubnet" `

-SecurityGroupName "myNetworkSecurityGroup" `

-PublicIpAddressName "myPublicIpAddress" `

-OpenPorts 80,3389

Note

Azure provides an default outbound access IP for Azure Virtual Machines which aren't assigned a public IP address, or are in the backend pool of an internal Basic Azure Load Balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

The default outbound access IP is disabled when a public IP address is assigned to the virtual machine or the virtual machine is placed in the backend pool of a Standard Load Balancer with or without outbound rules. If a [Azure Virtual Network NAT](https://docs.microsoft.com/en-us/azure/virtual-network/nat-gateway/nat-overview) gateway resource is assigned to the subnet of the virtual machine, the default outbound access IP is disabled.

For more information on outbound connections in Azure, see [Using Source Network Address Translation (SNAT) for outbound connections](https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-outbound-connections).

**Connect to virtual machine**

After the deployment has completed, RDP to the VM. To see your VM in action, the IIS web server is then installed.

To see the public IP address of the VM, use the [Get-AzPublicIpAddress](https://docs.microsoft.com/en-us/powershell/module/az.network/get-azpublicipaddress) cmdlet:

PowerShell

Get-AzPublicIpAddress -ResourceGroupName "myResourceGroup" | Select "IpAddress"

Use the following command to create a remote desktop session from your local computer. Replace the IP address with the public IP address of your VM.

PowerShell

mstsc /v:publicIpAddress

In the **Windows Security** window, select **More choices**, and then select **Use a different account**. Type the username as **localhost**\*username*, enter password you created for the virtual machine, and then click **OK**.

You may receive a certificate warning during the sign-in process. Click **Yes** or **Continue** to create the connection

**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:

PowerShell

Install-WindowsFeature -name Web-Server -IncludeManagementTools

When done, close the RDP connection to the VM.