Contents

[Azure 1](#_Toc452131600)

[Infrastructure as a Service - Azure Virtual Machines 2](#_Toc452131601)

[Choose the VM image 2](#_Toc452131602)

[Create the Windows virtual machine 3](#_Toc452131603)

[Connect to the virtual machine 8](#_Toc452131604)

[Stop the VM 10](#_Toc452131605)

[Platform as a Service - Azure Cloud Services 10](#_Toc452131606)

[Create and deploy 11](#_Toc452131607)

[Upload a certificate 13](#_Toc452131608)

[Verify your deployment 15](#_Toc452131609)

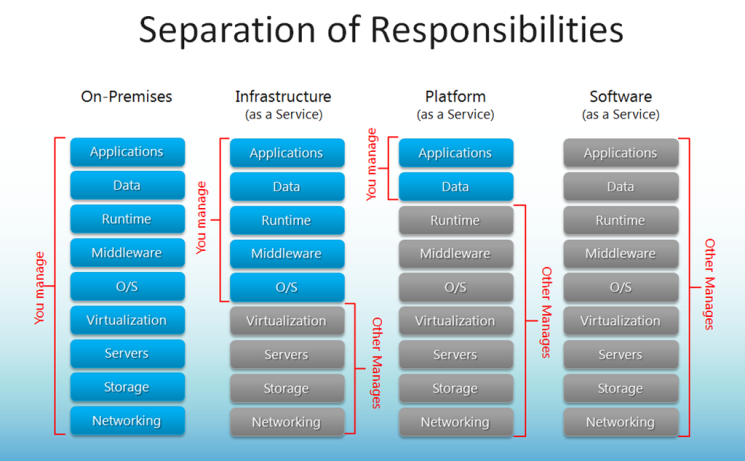
# 

# Azure

Microsoft Azure is an Internet-scale computing and services platform hosted in data centers managed or supported by Microsoft. It includes many separate features with corresponding developer services which can be used individually or together.

Windows Azure’s compute offerings fall into three main categories: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Let’s take a side-by-side look at each offering compared to the traditional On Premises model. It is offering serverless as service is also offering (Microsoft will take care assign resources based on usage).

cloud computing is the delivery of computing services like storage, networking, software etc.., over the Internet to offer faster innovation, flexible resources, and economies of scale. You typically pay only for cloud services you use, helping lower your operating costs, run your infrastructure more efficiently and scale as your business needs change.



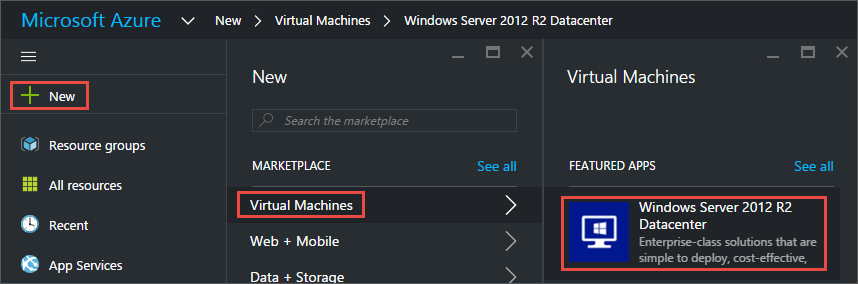
# Infrastructure as a Service - Azure Virtual Machines

In short, IaaS gives you a server in the cloud (virtual machine) that you have complete control over. With an Azure VM, you are responsible for managing everything from the Operating System on up to the application you are running.This mode of operation will feel most like a typical on premises virtual machine where you remote desktop into the server to manage it instead of sitting down in front of a physical keyboard and mouse.

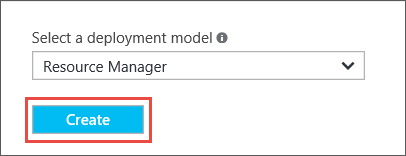
If you need a solution that requires custom third party software or multiple applications running on a single machine, then IaaS might be for you.

# Choose the VM image

1. Sign in to the [Azure portal](https://portal.azure.com/).
2. On the Hub menu, click **New** > **Virtual Machines** > **Windows Server 2012 R2 Datacenter**.



1. On the **Windows Server 2012 R2 Datacenter** page, under **Select a deployment model**, verify that **Resource Manager** is selected. Click **Create**.



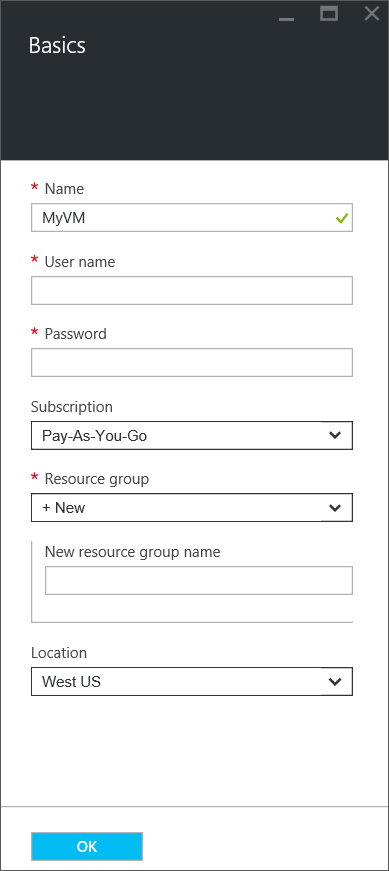
# Create the Windows virtual machine

After you select the image, you can use Azure's default settings for most of the configuration and quickly create the virtual machine.

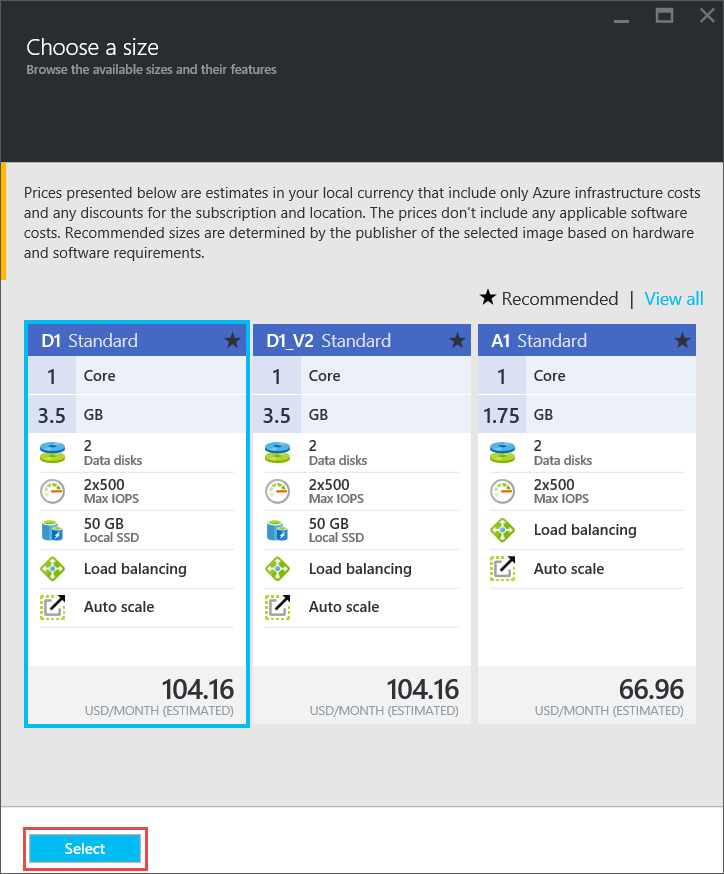
1. On the **Basics** blade, enter a **Name** for the virtual machine. The name must be 1-15 characters long and it cannot contain special characters.
2. Enter a **User name**, and a strong **Password** that will be used to create a local account on the VM. The local account is used to log on to and manage the VM.

The password must be at least 8-123 characters long and have at least 3 of the following: one lower case character, one upper case character, one number, and one special character.

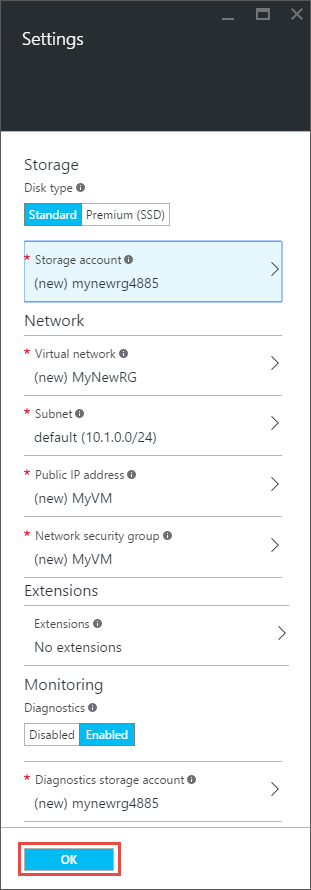
1. Select an existing [Resource group](https://azure.microsoft.com/en-us/documentation/articles/resource-group-overview/#resource-groups) or type the name for a new one. Type an Azure datacenter **Location** such as **West US**.
2. When you are done, click **OK** to continue to the next section.



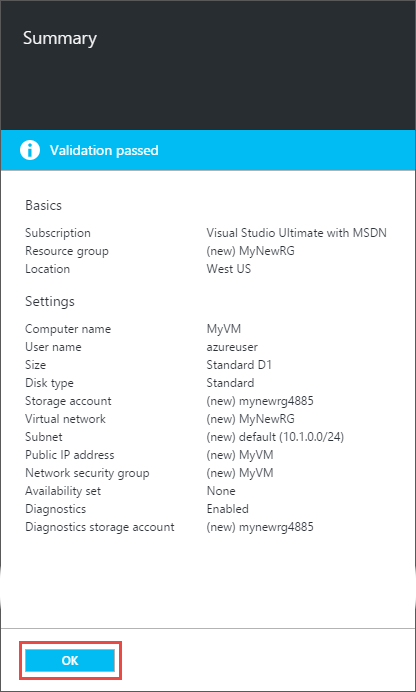
1. Choose a VM [size](https://azure.microsoft.com/en-us/documentation/articles/virtual-machines-windows-sizes/) and then click **Select** to continue.



1. On the **Settings** blade, you can change the storage and networking options. For a first virtual machine, you can generally accept the default settings. If you selected a virtual machine size that supports it, you can try out Premium Storage by selecting **Premium (SSD)** under **Disk type**. When you are done making changes, click **OK**.



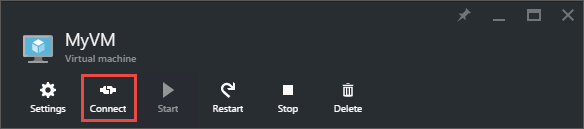
1. Click **Summary** to review your choices. When you're done, click **OK**.



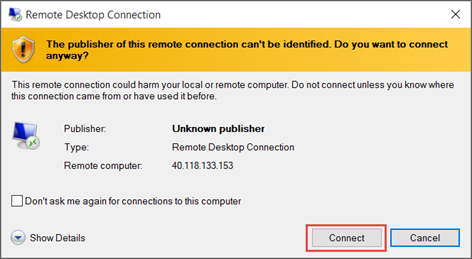
1. While Azure creates the virtual machine, you can track the progress under **Virtual Machines** in the hub menu.

# Connect to the virtual machine

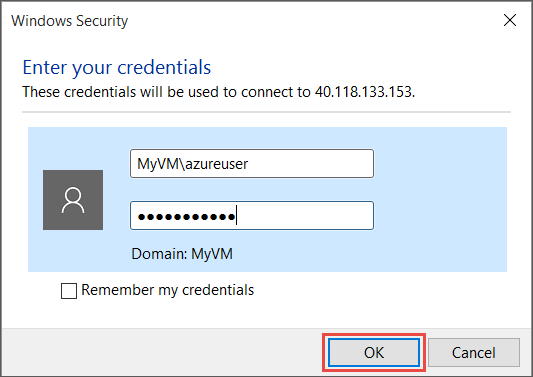
1. On the Hub menu, click **Virtual Machines**.
2. Select the virtual machine from the list.
3. On the blade for the virtual machine, click **Connect**. This creates and downloads a Remote Desktop Protocol file (.rdp file) that is like a shortcut to connect to your machine. You might want to save the file to your desktop for easy access. **Open** this file to connect to your VM.



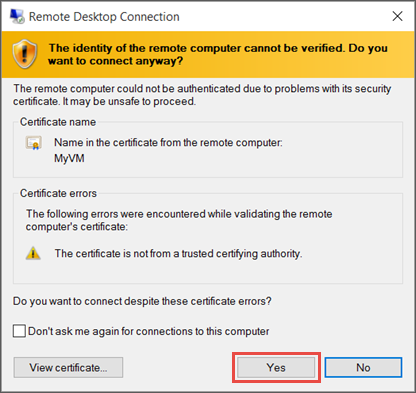
1. You will get a warning that the .rdp is from an unknown publisher. This is normal. In the Remote Desktop window, click **Connect** to continue.



1. In the Windows Security window, type the username and password for the local account that you created when you created the VM. The username is entered asvmname\username, then click **OK**.



1. You will get a warning that the certificate cannot be verified. This is normal. Click **Yes** to verify the identity of the virtual machine and finish logging on.

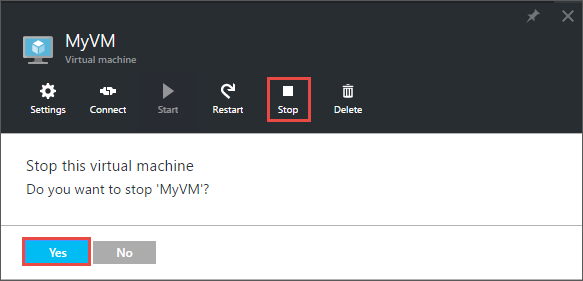


If you run into trouble when you try to connect, see [Troubleshoot Remote Desktop connections to a Windows-based Azure Virtual Machine](https://azure.microsoft.com/en-us/documentation/articles/virtual-machines-windows-troubleshoot-rdp-connection/).

You can now work with the virtual machine just as you would with any other server.

# Stop the VM

It is a good idea to stop the VM so you don't incur charges when you aren't actually using it. Just click the **Stop** button and then click **Yes**.



Just click the **Start** button to restart the VM when you are ready to use it again.

# Platform as a Service - Azure Cloud Services

An Azure Cloud Service consists of two components: your application files (source code, DLLs, etc.) and a configuration file. Together, these two elements will spin up a combination of Web Roles and Worker Roles to execute your application. With Cloud Services, Azure handles all of the tedious Operating System details for you, so you can focus on what matters - building a quality application for your users.A Web Role is an Azure VM that is pre-configured as a web server (running IIS) and will automatically have your application loaded on it by the time the server fully spins up. This will create the public endpoint for your application - usually a website, but it could also be an API or something similar.

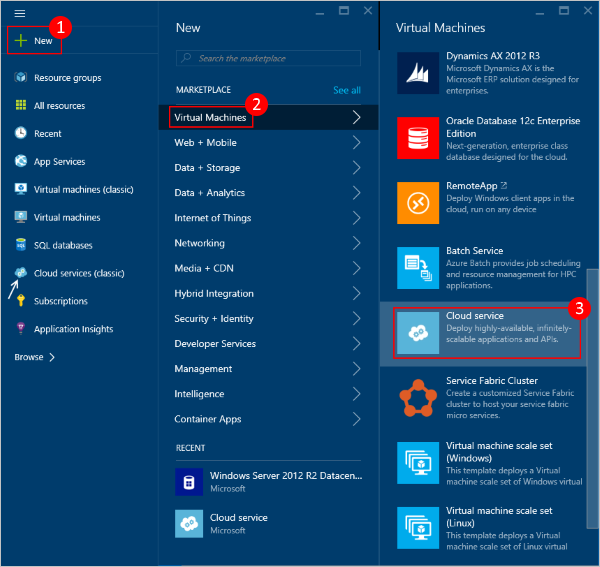
Worker Roles run alongside your Web Roles and are responsible for performing computing functions to support your application. Typically, the Web Role will accept some sort of user input and queue up an action for the Worker Role to process at a later time. This allows the Web Roles to be more responsive and to fire-and-forget tasks to be processed later.

A sample Azure architecture for a Cloud Services solution

You can pre-configure the number of Web and Worker Roles you want to start your application with and Autoscale to add or remove additional resources depending on demand.

# Create and deploy

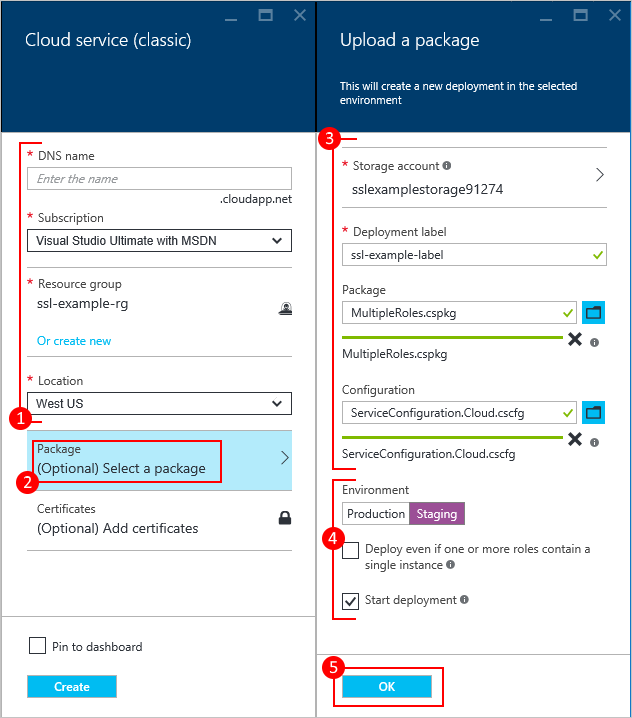
1. Log in to the [Azure portal](https://portal.azure.com/).
2. Click **New > Virtual Machines**, and then scroll down to and click **Cloud Service**.



1. At the bottom of the information page that displays, click **Create**.
2. In the new **Cloud Service** blade, enter a value for the **DNS name**.
3. Create a new **Resource Group** or select an existing one.
4. Select a **Location**.
5. Click **Package**. This will open the **Upload a package** blade. Fill in the required fields.

If any of your roles contain a single instance, ensure **Deploy even if one or more roles contain a single instance** is selected.

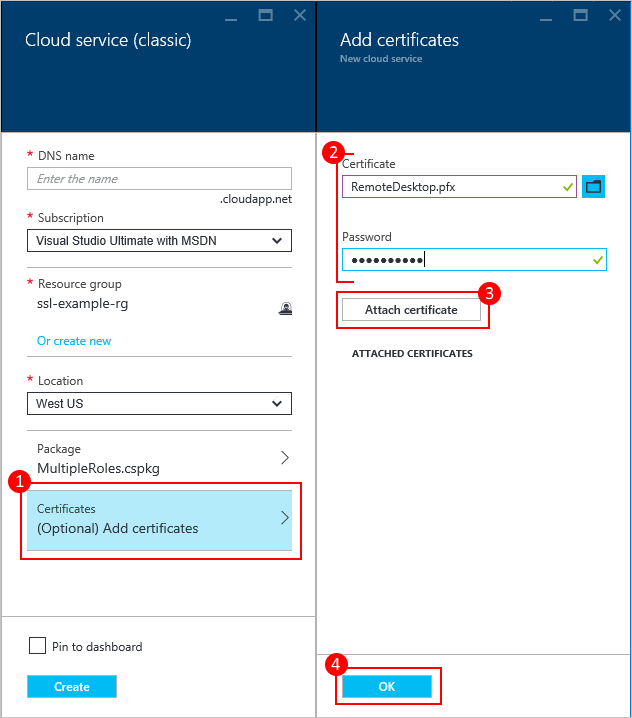
1. Make sure that **Start deployment** is selected.
2. Click **OK** which will close the **Upload a package** blade.
3. If you do not have any certificates to add, click **Create**.



# Upload a certificate

If your deployment package was [configured to use certificates](https://azure.microsoft.com/en-us/documentation/articles/cloud-services-configure-ssl-certificate-portal/#modify), you can upload the certificate now.

1. Select **Certificates**, and on the **Add certificates** blade, select the SSL certificate .pfx file, and then provide the **Password** for the certificate,
2. Click **Attach certificate**, and then click **OK** on the **Add certificates** blade.
3. Click **Create** on the **Cloud Service** blade. When the deployment has reached the **Ready**status, you can proceed to the next steps.

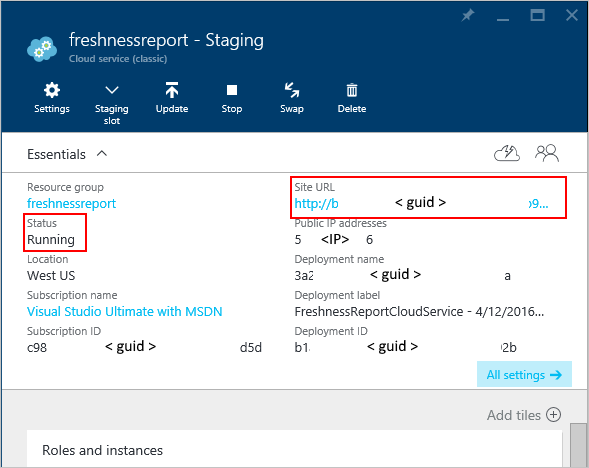


# Verify your deployment

1. Click the cloud service instance.

The status should show that the service is **Running**.

1. Under **Essentials**, click the **Site URL** to open your cloud service in a web browser.



**Platform as a Service (Light) - Azure Websites**

Azure Websites are, by far, the easiest, cheapest, and fastest method of getting your application to the cloud. Websites give you the least amount of configuration control over your application, which is perfect for simple web applications that do not require more complex configurations for scaling and data processing.Deploying your application to an Azure Website is mind-numbingly easy. You can use your favorite IDE or configure source control (Git, GitHub, Bitbucket, CodePlex, TFS, and DropBox) to push to your website and let Azure take care of the rest for you. Continuous Integration, built in, right out-of-the-box.

You can even include a MySQL or SQL Server database when provisioning an Azure Website through the Azure Management Portal. Once your Website has been configured, simply update your Web.config (or similar) with your new database’s credentials,and you are good to go!

**Software as a Service - Basecamp, Salesforce, Office 365, Azure Websites**

Finally, Software as a Service applications are built and hosted through 3rd party vendors who typically charge for a certain level of service - $30/month for X projects and Y users.Azure Websites can serve as a SaaS offering as well. You can configure a Wordpress, Drupal, OpenX, or even phpBB site with a single click. No code, no deployment hassles, and minimal configuration. Azure Websites lets you stand up the service you need in minutes, not hours or days.Most SaaS applications today are built on a cloud platform due to the low cost of entry - with prices continually falling - and the ability to scale up as your customer base grows. If Dropcam, SmugMug, or Netflix got one million new customers tomorrow, their infrastructure (Amazon Web Services) would be able to accommodate them.