

Table of Contents

Overview	1
Installation from Azure Portal	1
Create a Service Fabric Managed Cluster in Azure Portal	2
Publish the Application to the Service Fabric Managed Cluster	6
Obtain Server DNS Name	9
Installation from Visual Studio	9
Create a Service Fabric Cluster in Visual Studio	10
Publish the application in the Service Fabric Cluster	12
Obtain Server IP Address and DNS Name	13
Server Configuration	15
Running the Demo Applications	15
Using the Installed Server	16
Uninstallation	17

Overview

EVO PDF Server can be installed in an Azure Service Fabric Application. After installation the server can be accessed by the assigned IP address or DNS name.

Installation from Azure Portal

In this scenario, the installation is performed in two steps. In the first step, a Service Fabric Managed Cluster is created in the Azure Portal. In the second step, the application is published to the newly created cluster from the Visual Studio 2022 solution for the Azure Service Fabric Application, which can be found in the `Src` subfolder.

Before publishing the service, you can configure various options, such as the service password, in the `Src\EvoPdfServer\_AzureServiceFabric\ApplicationPackageRoot\ApplicationManifest.xml` configuration file. A detailed description of each configuration option is provided in a separate section of this document.

Create a Service Fabric Managed Cluster in Azure Portal

First, use the script below in PowerShell to create a certificate and install it in the current user's Personal certificate store.

```
$dnsName = "evomanagedcluster.westus.cloudapp.azure.com"
$cert = New-SelfSignedCertificate -Subject "CN=$dnsName" `
    -DnsName $dnsName `
    -CertStoreLocation "Cert:\CurrentUser\My" `
    -KeyAlgorithm RSA `
    -KeyLength 2048 `
    -KeyUsage DigitalSignature,KeyEncipherment `
    -Type Custom `
    -TextExtension @"(2.5.29.37={text}1.3.6.1.5.5.7.3.2)" `
    -NotAfter (Get-Date).AddYears(2)
```

The 'dnsName' variable must match the domain name of the cluster, where the first component is the name you will assign to the cluster (evomanagedcluster) and the second component is the region where the cluster will be created.

Then use the following command:

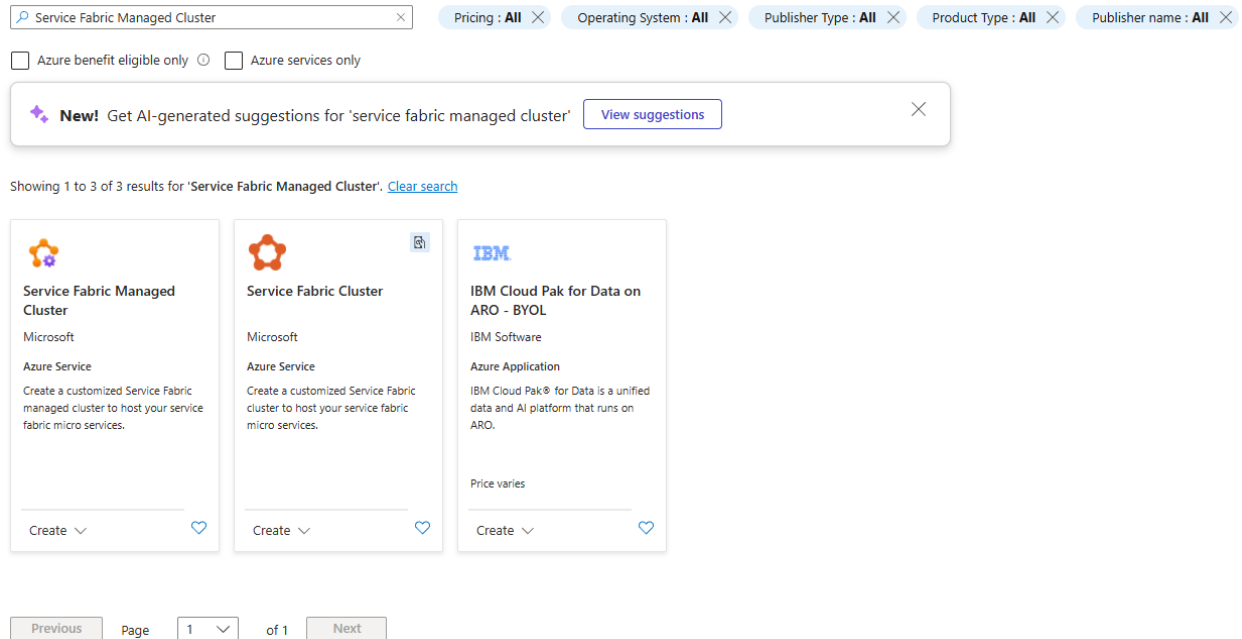
```
$cert.Thumbprint
```

to display the certificate thumbprint, which will be used when creating the cluster in the Azure Portal.

Example:

```
PS> $cert.Thumbprint
15E620839D892C7DA9A90BD86ADB954F57C5CF7C
```

Log in to your Microsoft Azure Portal account, choose to create a new resource and search for the Service Fabric Managed Cluster resource type.



Click the 'Create' button to start the wizard. For the 'Resource group', click 'Create new' and enter the group name. Choose the 'Cluster name' and the region where the cluster will be created in the 'Cluster details' section. Enter the username and password in the 'Administrator account' section. You can see an example in the following image.

Create a Service Fabric managed cluster

Basics Node types Networking Advanced Tags Review + create

Azure Service Fabric is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices and containers. The managed cluster offering makes it easier to deploy and manage Service Fabric clusters by removing the need to individually configure additional resources such as virtual machine scale sets and load balancers. [Learn more](#)

Project details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Pay-As-You-Go
Resource group *	(New) evomanagedcluster

[Create new](#)

Cluster details

Cluster name *	evomanagedcluster	.westus.cloudapp.azure.com
Region *	(US) West US	
SKU *	☐ Basic Intended for development and testing environments using one node type. ☒ Standard Best for general and production scenarios requiring multiple node types.	
Enable zonal resiliency	☐ ⓘ The selected region does not support enough Availability Zones for zonal resiliency. At least three Availability Zones are required. View regions that support availability zones	
Enable automatic OS upgrades	☒	
Administrator account		
Username *	evopdf	
Password *		
Confirm password *		

Select an OS image (Windows 2025 in this example), a VM size and a node count, which is set to 5 by default.

In the 'Authentication methods' section, enable the 'Enable client certificates' option and choose 'Thumbprint' as the 'Certificate source'. Select 'AdminClient' as the certificate type. For 'Common name', enter the cluster DNS value that was also used in the certificate creation script. For 'Thumbprint', use the value displayed by the same script. You can see an example in the following image.

Primary node type

Node types define groups of nodes (virtual machines) that will be used to run your cluster's applications. A Service Fabric cluster can consist of more than one node type, in which case one of the node types will be the primary and other node types will be secondary. You can configure additional node types on the Node types tab above. [Learn more](#)

OS Image *	WindowsServer 2025-datacenter-g2 Show all images
Node size *	Standard DS2 v2 Select VM size
Data disk size *	256 GiB Change size
Node count *	5 100

Authentication methods

When you connect to your cluster, you will authenticate with either a certificate in Azure Key Vault or with Azure Active Directory. You will be able to update this later if you change your mind. If you do not configure either option, you will not be able to directly connect to your cluster. [Learn more](#)

Enable client certificates	☒
Certificate source *	☐ Keyvault ☒ Thumbprint

Certificate type	Common name (optional)	Thumbprint
AdminClient	evomanagedcluster.westus.clou...	15E620839D892C7DA9A908D... ✓

Enable Azure Active Directory	☐
-------------------------------	--------------------------

[Review + create](#)[< Previous](#)[Next : Node types >](#)[Download a template for automation](#)

Click the 'Next: Node types' button to go the next panel and then click 'Next: Networking'.

Create a Service Fabric managed cluster ...

Basics Node types **Networking** Advanced Tags Review + create

Configure network settings for your cluster, such as connection and gateway ports as well as load balancing rules. You will be able to update these settings after deploying your cluster. [Learn more](#)

Client connection port *	19000
HTTP gateway port *	19080

Load balancing rules

Open specific ports on your managed load balancer for inbound traffic from the internet. A probe and an NSG rule with default priority will automatically be created for the specified port mapping. [Learn more](#)

Frontend port	Backend port	Protocol	Probe protocol	Probe request path
40001 ✓	40001 ✓	TCP	TCP	

Review + create

< Previous

Next : Advanced >

[Download a template for automation](#)

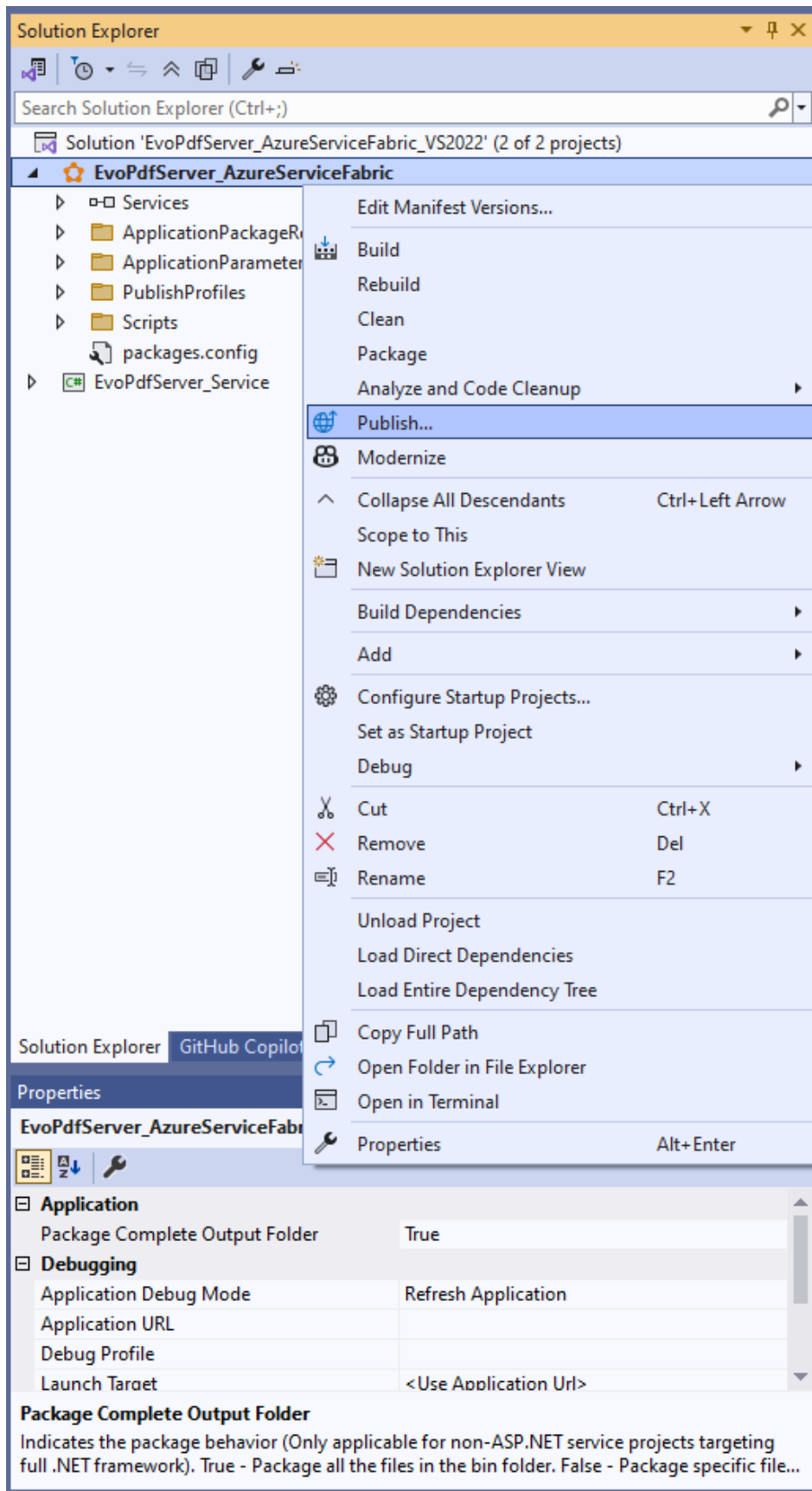
In this panel, you can specify the default port (40001) used by the service, to which the client libraries will connect.

Click the 'Review + create' button to create the cluster. The cluster creation process may take some time. Once the cluster is ready, you can start publishing the Service Fabric application from Visual Studio 2022.

Publish the Application to the Service Fabric Managed Cluster

Open the 'EvoPdfServer_AzureServiceFabric.sln' solution in Visual Studio, right-click the 'EvoPdfServer_AzureServiceFabric' project and choose the 'Publish' option from the context menu.

To publish the application to the Service Fabric Managed Cluster created earlier, open the 'EvoPdfServer_AzureServiceFabric.sln' solution in Visual Studio, right-click the 'EvoPdfServer_AzureServiceFabric' project and choose the 'Publish' option from the context menu.



In the 'Publish Service Fabric Application' panel, choose your Azure account and under 'Connection Endpoint', select the cluster you created in the previous step.

In the 'Advanced Connection Parameters' section, set 'FindType' to 'FindByThumbprint', 'FindValue' to the certificate thumbprint string obtained after certificate creation, 'StoreLocation' to 'CurrentUser' and 'StoreName' to 'My'. The 'ServerCertThumbprint' field should already be filled in by the Visual Studio wizard.

Publish Service Fabric Application ✕

Target profile:
PublishProfiles\Cloud.xml ▼ [Save Profile](#)



Connection Endpoint:
evomanagedcluster.westus.cloudapp.azure.com:19000 ▼ ✓
[Add custom endpoint...](#)
[Create new cluster...](#)

 **Advanced Connection Parameters**

Name	Value	Remove
ServerCertThumbprint	4D056F75F51D118B2923D56CE3869248C090D374	
X509Credential	true	
FindType	FindByThumbprint	
FindValue	15E620839D892C7DA9A90BD86ADB954F57C5CF7C	
StoreLocation	CurrentUser	
StoreName	My	
Add a parameter		

[How to configure secure connections](#)

Application parameters file:
ApplicationParameters\Cloud.xml ▼ [Edit...](#)

☐ Upgrade the Application
[Configure upgrade settings...](#)

[Manifest Versions...](#) [Publish](#) [Cancel](#)

Important Note: If the connection to the cluster fails, the workaround is to cancel the application publishing process by pressing the 'Cancel' button. This forces Visual Studio to update the publish profile

configuration files based on the selected cluster. Then try publishing again. This time, the connection to the cluster should succeed and you can continue with the next step.

Click the 'Publish' button to publish the application to the Service Fabric Cluster. You will be able to see the progress of the deployment process in Visual Studio.

Obtain Server DNS Name

In the Azure Portal, in the 'Overview' panel of the cluster you created, you can see the FQDN. You can use this DNS name as the address to connect the client applications to the installed server. By default, it should look like 'evomanagedcluster.westus.cloudapp.azure.com'.

The screenshot displays the Azure Portal interface for a Service Fabric managed cluster. The left sidebar shows the navigation menu with 'Overview' selected. The main content area is divided into several sections:

- Essentials:**
 - Resource group: [evomanagedcluster](#)
 - Location: [West US](#)
 - Subscription: [Pay-As-You-Go](#)
 - Subscription ID: [f6dc5551-b489-4388-a9d1-b220e60c399](#)
 - Cluster State: [Ready](#)
 - Tags: [SFRP.DisableDefaultOutboundAccess: true](#)
- Properties:**
 - Service Fabric managed cluster:**
 - Cluster version: 11.3.365.1
 - SKU: Standard
 - Cluster ID: f678b2b7-6a8b-46b0-9b8c-8c9d2d7233e
 - Zonal resiliency: Disabled
 - Node types:**
 - Node type count: 1
 - Node types: nodetype1
 - Applications:**
 - Applications count: 0
- Networking:**
 - DNS name: evomanagedcluster
 - FQDN name: [evomanagedcluster.westus.cloudapp.azure.com](#)
 - Client connection port: 19000
 - HTTP gateway port: 19080
 - Load balancing rules: 1
- Fabric upgrades:**
 - Current fabric version: 11.3.365.1
 - Upgrade wave: Wave0

Installation from Visual Studio

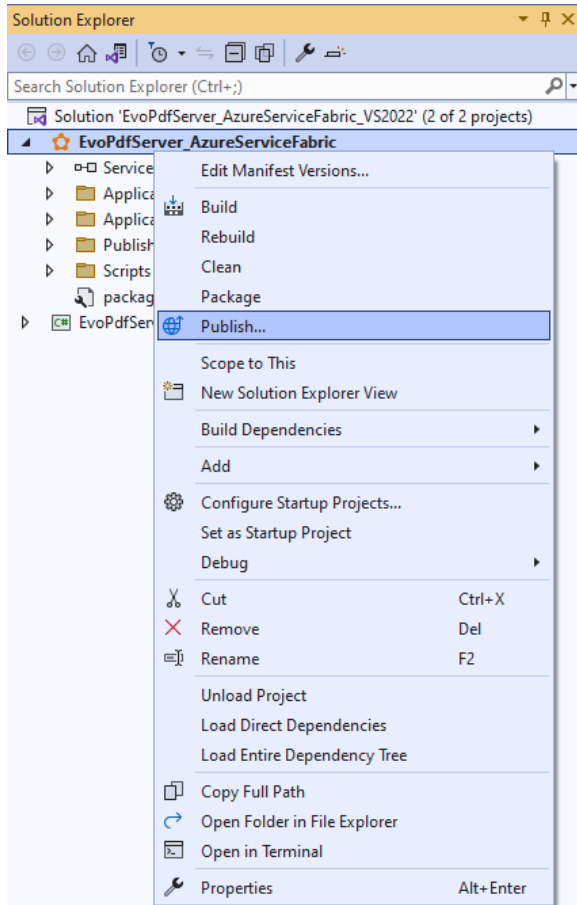
The server can be easily installed using the Visual Studio solution for Azure Service Fabric Application from 'Src' subfolder.

Before publishing the service, you can set various configuration options like a service password in the 'Src\EvoPdfServer_AzureServiceFabric\ApplicationPackageRoot\ApplicationManifest.xml' configuration file. The description of each configuration option can be found in a separate section of this document.

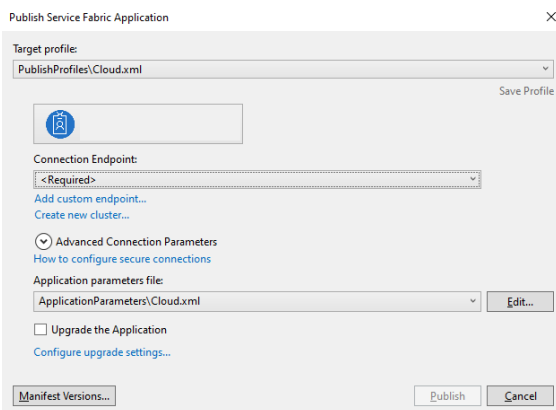
The installation is made in two steps. In the first step a Service Fabric Cluster is created and in the second step the application is published in the created cluster.

Create a Service Fabric Cluster in Visual Studio

To create a Service Fabric Cluster from Visual Studio, open the `EvoPdfServer_AzureServiceFabric.sln` solution in Visual Studio, right click on 'EvoPdfServer_AzureServiceFabric' project and choose the 'Publish' option from the contextual menu.

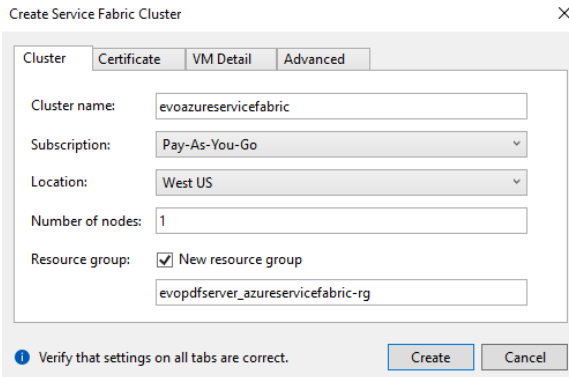


The 'Publish Service Fabric Application' panel will open.



In the 'Publish Service Fabric Application' panel choose your Azure account and under 'Connection Endpoint' choose 'Create new cluster...' option.

In the 'Create Service Fabric Cluster' panel, in the 'Cluster' tab, choose a name for the cluster which must not contain special characters, your Azure account subscription, a location for the service. For the number of nodes, you can enter 1 for a single node. You can use the values from the screenshot below or use your own name for cluster if the name from image is already in use.

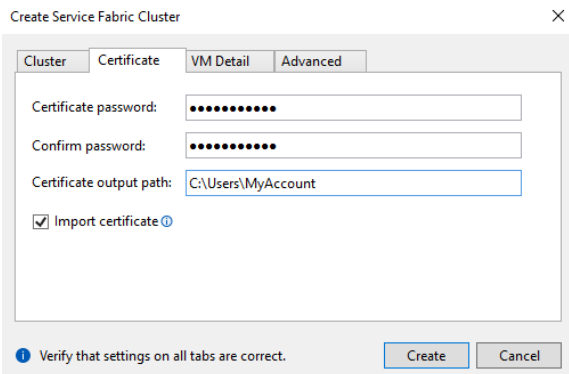


The screenshot shows the 'Cluster' tab of the 'Create Service Fabric Cluster' dialog. The fields are as follows:

- Cluster name:
- Subscription:
- Location:
- Number of nodes:
- Resource group: ☒ New resource group

At the bottom, there is a message: "Verify that settings on all tabs are correct." and two buttons: "Create" and "Cancel".

In 'Certificate' tab enter a password for the certificate which is used when accessing the service explorer web page.

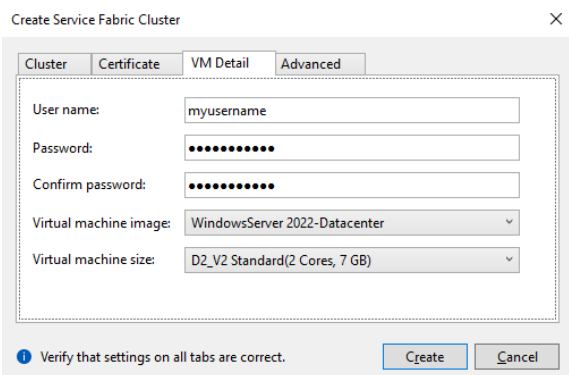


The screenshot shows the 'Certificate' tab of the 'Create Service Fabric Cluster' dialog. The fields are as follows:

- Certificate password:
- Confirm password:
- Certificate output path:
- ☒ Import certificate ⓘ

At the bottom, there is a message: "Verify that settings on all tabs are correct." and two buttons: "Create" and "Cancel".

In the 'VM Detail' tab enter a username and a password for the virtual machine and a 'Virtual machine size' for example a machine with 2 Cores and 7 GB RAM.

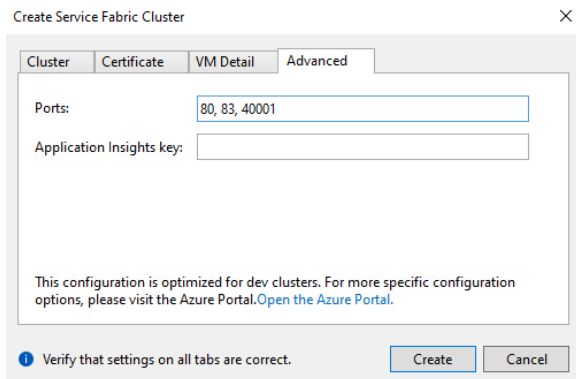


The screenshot shows the 'VM Detail' tab of the 'Create Service Fabric Cluster' dialog. The fields are as follows:

- User name:
- Password:
- Confirm password:
- Virtual machine image:
- Virtual machine size:

At the bottom, there is a message: "Verify that settings on all tabs are correct." and two buttons: "Create" and "Cancel".

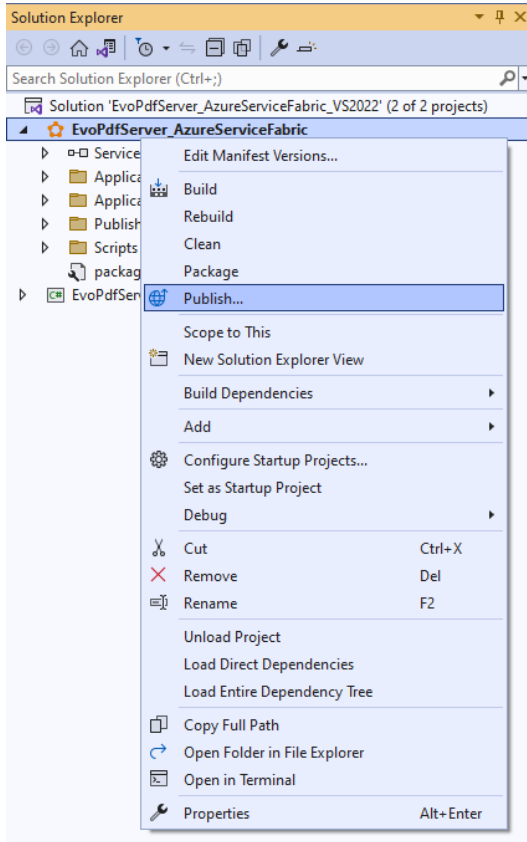
In the 'Advanced' tab, in the ports list add the EVO PDF Server port number 40001 in comma separated list of ports.



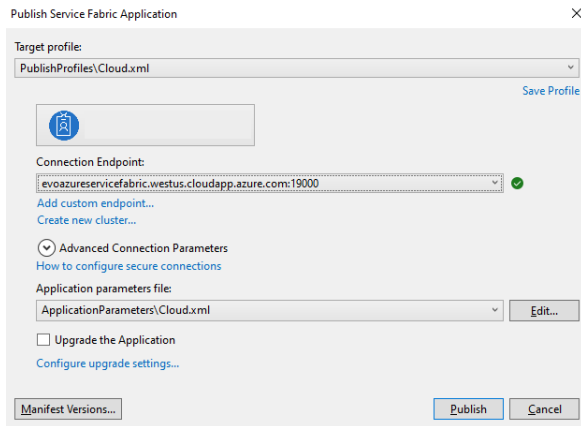
Click the 'Create' button and confirm the Summary with 'OK' button to create the Service Fabric Cluster. Wait a few minutes until the cluster is successfully created.

Publish the application in the Service Fabric Cluster

To publish the application in the Service Fabric Cluster from Visual Studio, right click on 'EvoPdfServer_AzureServiceFabric' project and choose 'Publish' option from the contextual menu.



In the 'Publish Service Fabric Application' panel choose your Azure account and under 'Connection Endpoint' choose the cluster you created at the previous step. You can leave the other options with their default values.



Important Note: If the connection to the cluster fails, then the workaround is to cancel the application publishing by pressing the 'Cancel' button to force Visual Studio to update the publish profile configuration files based on the cluster you selected and then try again to publish. This time the connection to cluster should succeed and you can continue with the next step.

Click the 'Publish' button to publish the application in Service Fabric Cluster. You'll be able to see the evolution of the deployment process in Visual Studio.

Obtain Server IP Address and DNS Name

When the deployment is complete you can login into your account of Microsoft Azure Portal to obtain the public IP address and the DNS name you can use to connect the client applications to server.

In Azure Portal you can find the Azure Service Fabric Application you created if you select 'All resources' item from the menu of the Azure Management Portal. Click the link to 'Service Fabric cluster' resource from the list of resources to open the 'Overview' panel of Service Fabric cluster.



In this panel you can see the service DNS name under 'Client connection endpoint' section.

It should look by default like 'clustername.location.cloudapp.azure.com', without any port number. You can use this DNS name as IP address to connect the client applications to the installed server.

In the example from this document the DNS name of the server you can also use where an IP address is expected is '**evoazureservicefabric.westus.cloudapp.azure.com**'.

The DNS name of the service can also be found, after application publishing, in 'Src\EvoPdfServer_AzureServiceFabric\PublishProfiles\Cloud.xml' configuration file from the Visual Studio project.

You might see in the same panel a warning about the '-with-Container' VM images. You can ignore that generic warning because it is about the old types of VM images and it does not apply to cluster you created.

You can obtain the public IP address to be used in 'Public IP address' resource created for the cluster. Click that resource in the resources list to open its 'Overview' panel.

The screenshot shows the 'Overview' page for the Public IP address resource 'PublicIP-evoazureservicefabric-0'. The left sidebar contains navigation links: Overview, Activity log, Access control (IAM), Tags, Settings, Configuration, Properties, and Locks. The main content area has a top bar with a search box and a warning: 'Upgrade to Standard SKU - Microsoft recommends Standard SKU public IP address for production workloads'. Below this is a toolbar with 'Associate', 'Dissociate', 'Delete', 'Move', 'Refresh', and 'Open in mobile' buttons. The 'Essentials' section lists: Resource group (move) : evopdfserver_azureservicefabric-rg, Location (move) : West US, and Subscription (move) : Pay-As-You-Go. On the right, a table lists properties: SKU : Basic, Tier : Regional, IP address : 20.245.56.246, DNS name : evoazureservicefabric.westus.cloudapp.azure.com, Associated to : LB-evoazureservicefabric-bpzc6dch, Virtual machine : -, and Routing preference : Microsoft network.

In this panel you can find the IP address under the 'IP address' section.

The IP address can also be found in the 'Load balancer' resource created in Azure portal for cluster. If it's not visible then click the 'See more' button.

The screenshot shows the 'Overview' page for the Load balancer resource 'LB-evoazureservicefabric-bpzc6dch'. The left sidebar contains navigation links: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Frontend IP configuration, Backend pools, Health probes, and Load balancing rules. The main content area has a top bar with a search box and buttons: 'Move', 'Delete', 'Refresh', and 'Give feedback'. Below this is a purple banner: 'Learn about the benefits of upgrading to Standard SKU ->'. The 'Essentials' section lists: Resource group (move) : evopdfserver_azureservicefabric-rg, Location : West US, and Subscription (move) : Pay-As-You-Go. On the right, a table lists properties: Backend pool : LoadBalancerBEAddressPool (1 virtual machine), Load balancing rule : 5 rules, Health probe : 5 probes, NAT rules : 1 inbound, Tier : Regional, and Public IP address : 20.245.56.246 (PublicIP-evoazureservicefabric-0). At the bottom, the 'Tags' section shows: resourceType : Service Fabric, clusterName : evoazureservicefabric. A 'See less' link is at the bottom left.

In the picture above the 'See more' button changed to 'See less' after click.

Server Configuration

EVO PDF Server offers various configuration options you can set in the 'Src\EvoPdfServer_AzureServiceFabric\ApplicationPackageRoot\ApplicationManifest.xml' file of the Web Application before installation.

The possible configuration options and their default values are described below.

- **ServicePassword** option offers the possibility to protect the service with a password to prevent unauthorized access. By default, this option is not set.
- **MaxConcurrentClients** option represents the maximum number of request the server can process in parallel. If the maximum number of parallel requests is reached, the subsequent requests are queued until a slot becomes available. The default value for this option is 12.
- **Distributed** option offers the possibility to execute each request in its own process when its value is true. By default, this option is false.
- Impersonation options consisting in **ImpersonationUsername**, **ImpersonationPassword**, **ImpersonationDomain** and **ImpersonationLogonMode** options can be used to run the service under a specific user account on the hosting Windows machine. By default, they are not set.

Running the Demo Applications

After EVO PDF Server installation you are ready to start using the client libraries in your applications. You can start with our demo applications from EVO PDF Client ZIP packages which can be downloaded directly from [EVO PDF Software Downloads Page](#) and extracted in a folder.

There are separate packages for **.NET Framework** and **.NET Core** with demo application for all components and features.

- The package for .NET Core contains the product binaries, demo Visual Studio projects with full C# code for ASP.NET Core targeting .NET Core 6.0 and later versions, the library documentation in CHM format
- The package for .NET Framework contains the product binaries, demo Visual Studio projects with full C# code for ASP.NET Web Forms and ASP.NET MVC targeting .NET Framework 4 and later versions, the library documentation in CHM format

There are separate demo projects for each major component of the library, including **HTML to PDF**, **Word to PDF**, **Excel to PDF**, **PDF to Text**, **PDF to Image**, **PDF to HTML** and **PDF Images Extractor** corresponding to the tools from **EVO PDF Toolkit** bundle.

The demo applications allow you to enter the server IP address and the port number if you installed the server as a TCP/IP service or the server URL if you have installed the server as a web service.

Using the Installed Server

EVO PDF Server was installed in an Azure Service Fabric Application and it can be accessed in client applications by its IP address or DNS name which is **evoazureservicefabric.westus.cloudapp.azure.com** in the example from this document.

In the demo application you can select the **Use TCP/IP** radio button and enter the IP address or the DNS name you previously obtained.

Below is a screenshot of the HTML to PDF demo client application from client package for .NET Core using the IP address of the server installed in the Azure Service Fabric Application.

Below is a screenshot of the HTML to PDF demo client application from client package for .NET Core using the DNS name of the server installed in the Azure Service Fabric Application.


EVO PDF Software
HTML to PDF Converter
for .NET

HTML to PDF Converter

- Getting Started with HTML to PDF
- HTML Content Destination and Scaling in PDF
- Convert the Current HTML Page to PDF
- Convert a HTML Page to PDF in Same Session
- Merge Multiple HTML Pages into a Single PDF
- Merge HTML with Existing PDF Documents
- Partially Convert of a HTML Page to PDF
- Repeat HTML Table Header and Footer in PDF
- ▣ Headers and Footers
- ▣ Page Breaks Control
- ▣ Screen and Print Media Types
- ▣ Conversion Triggering Modes
- ▣ Hierarchical Bookmarks
- ▣ Table of Contents
- ▣ Live PDF Forms
- ▣ HTML5 Features
- ▣ HTTP GET and POST, Proxy Options
- ▣ HTTP Headers and Cookies
- ▣ Fonts Embedding
- ▣ HTML Elements Location in PDF
- ▣ HTML Elements Visibility in PDF
- ▣ URI Links in PDF
- ▣ Internal Links in PDF
- ▣ File Links and Attachments
- ▣ Text Notes in PDF
- ▣ PDF Actions
- ▣ HTML to PDF Elements
- ▣ HTML to Image Elements
- ▣ Images Quality in Generated PDF
- ▣ Flash and Extensions Support
- ▣ Watermarks and Stamps
- ▣ PDF/A and PDF/X Standards
- ▣ CMYK and GrayScale Color Spaces
- ▣ PDF Viewer Preferences
- ▣ PDF Security Features
- ▣ **HTML to Image Converter**
- ▣ **PDF Creator**
- ▣ **PDF Editor**

Getting Started with EVO HTML to PDF Converter
 Convert a HTML page from an URL or a HTML String to PDF using the basic options of the EVO HTML to PDF Converter. The Full Description and a Code Sample can be accessed from the top tabs.

☒ Use TCP/IP Server IP Address: Port:
☐ Use Web Service URL:
 Service Password:

☒ Convert URL or Local File ☐ Convert HTML String

HTML Page URL or Local File to Convert

HTML Viewer Options
 Width: px Height: px

PDF Page Options
 Size: Orientation:

Navigation Options
 Timeout: sec Delay Conversion : sec

☐ Display PDF inline

Uninstallation

You can remove the Service Fabric Cluster from your Azure account management portal.