



Windows PowerShell Get-Help on Cmdlet 'New-AzServiceFabricCluster'

PS: \>Get-HELP New-AzServiceFabricCluster -Full

NAME

New-AzServiceFabricCluster

SYNOPSIS

This command uses certificates that you provide or system generated self-signed certificates to set up a new service fabric cluster. It can use a default template or

a custom template that you provide. You have the option of specifying a folder to export the self-signed certificates to or fetching them later from the key vault.

SYNTAX

```
New-AzServiceFabricCluster [-ResourceGroupName] <System.String> [-CertificateCommonName <System.String>]
[-CertificateIssuerThumbprint <System.String>]
[-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>]
-ParameterFile <System.String> -SecretIdentifier
<System.String> -TemplateFile <System.String> [-Thumbprint <System.String>] [-VmPassword
<System.Security.SecureString>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

```
New-AzServiceFabricCluster [-ResourceGroupName] <System.String> [-CertificateCommonName <System.String>]
-CertificateFile <System.String>
```

```

[-CertificateIssuerThumbprint <System.String>] [-CertificatePassword <System.Security.SecureString>] [-DefaultProfile
    <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-KeyVaultName
<System.String>] [-KeyVaultResourceGroupName
    <System.String>] -ParameterFile <System.String> [-SecondaryCertificateFile <System.String>]
[-SecondaryCertificatePassword <System.Security.SecureString>]
    -TemplateFile <System.String> [-VmPassword <System.Security.SecureString>] [-Confirm] [-WhatIf]
[<CommonParameters>]

```

```

New-AzServiceFabricCluster [-ResourceGroupName] <System.String> [-CertificateOutputFolder <System.String>]
[-CertificatePassword <System.Security.SecureString>]
[-CertificateSubjectName <System.String>] [-ClusterSize <System.Int32>] [-DefaultProfile
    <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-KeyVaultName
<System.String>] [-KeyVaultResourceGroupName
    <System.String>] -Location <System.String> [-Name <System.String>] [-OS {WindowsServer2012R2Datacenter |
WindowsServer2016Datacenter |
    WindowsServer2016DatacenterwithContainers | UbuntuServer1604 | UbuntuServer1804 | UbuntuServer2004 |
WindowsServer2022}] -VmPassword <System.Security.SecureString>
[-VmSku <System.String>] [-VmUserName <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

```

```

New-AzServiceFabricCluster [-ResourceGroupName] <System.String> [-CertificateOutputFolder <System.String>]
[-CertificatePassword <System.Security.SecureString>]
    [-CertificateSubjectName <System.String>] [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-KeyVaultName
    <System.String>] [-KeyVaultResourceGroupName <System.String>] -ParameterFile <System.String> -TemplateFile
<System.String> [-VmPassword
    <System.Security.SecureString>] [-Confirm] [-WhatIf] [<CommonParameters>]

```

DESCRIPTION

The New-AzServiceFabricCluster command uses certificates that you provide or system generated self-signed certificates to set up a new service fabric cluster. The

template used can be a default template or a custom template that you provide. You have the option of specifying a folder to export the self-signed certificates or

fetching them later from the key vault. If you are specifying a custom template and parameter file, you don't need to provide the certificate information in the

parameter file, the system will populate these parameters. The four options are detailed below. Scroll down for explanations of each of the parameters.

PARAMETERS

-CertificateCommonName <System.String>

Certificate common name

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-CertificateFile <System.String>

The existing certificate file path for the primary cluster certificate

Required? true

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-CertificateIssuerThumbprint <System.String>

Certificate issuer thumbprint, separated by commas if more than one

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-CertificateOutputFolder <System.String>

The folder of the new certificate file to be created

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-CertificatePassword <System.Security.SecureString>

The password of the certificate file

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-CertificateSubjectName <System.String>

The subject name of the certificate to be created

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-ClusterSize <System.Int32>

The number of nodes in the cluster. Default are 5 nodes

Required? false

Position? named

Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>

The credentials, account, tenant, and subscription used for communication with Azure.

Required? false
Position? named
Default value None
Accept pipeline input? False
Accept wildcard characters? false

-KeyVaultName <System.String>

Azure key vault name, if not given it will be defaulted to the resource group name

Required? false
Position? named
Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-KeyVaultResourceGroupName <System.String>

Azure key vault resource group name, if not given it will be defaulted to resource group name

Required? false
Position? named
Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-Location <System.String>

The resource group location

Required? true
Position? named
Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-Name <System.String>

Specify the name of the cluster, if not given it will be same as resource group name

Required? false
Position? named
Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-OS <Microsoft.Azure.Commands.ServiceFabric.Models.OperatingSystem>

The Operating System of the VMs that make up the cluster.

Required? false
Position? named
Default value None
Accept pipeline input? True (ByPropertyName)
Accept wildcard characters? false

-ParameterFile <System.String>

The path to the template parameter file.

Required? true
Position? named
Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-ResourceGroupName <System.String>

Specify the name of the resource group.

Required? true

Position? 0

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-SecondaryCertificateFile <System.String>

The existing certificate file path for the secondary cluster certificate

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-SecondaryCertificatePassword <System.Security.SecureString>

The password of the certificate file

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-SecretIdentifier <System.String>

The existing Azure key vault secret URL, for example

'https://mykv.vault.azure.net:443/secrets/mysecrets/55ec7c4dc61a462bbc645ffc9b4b225f'

Required? true

Position? named
Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-TemplateFile <System.String>

The path to the template file.

Required? true
Position? named
Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-Thumbprint <System.String>

The thumbprint for the certificate corresponding to the SecretIdentifier. Use this if the certificate is not managed as the key vault would only have the certificate stored as a secret and the cmdlet is unable to retrieve the thumbprint.

Required? false
Position? named
Default value None
Accept pipeline input? True (ByValue)
Accept wildcard characters? false

-VmPassword <System.Security.SecureString>

The password of the Vm.

Required? true
Position? named
Default value None
Accept pipeline input? True (ByPropertyName)
Accept wildcard characters? false

-VmSku <System.String>

The Vm Sku

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-VmUserName <System.String>

The user name for logging to Vm

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-Confirm <System.Management.Automation.SwitchParameter>

Prompts you for confirmation before running the cmdlet.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-WhatIf <System.Management.Automation.SwitchParameter>

Shows what would happen if the cmdlet runs. The cmdlet is not run.

Required? false

Position? named

Default value False
Accept pipeline input? False
Accept wildcard characters? false

<CommonParameters>

This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable. For more information, see about_CommonParameters (<https://go.microsoft.com/fwlink/?LinkID=113216>).

INPUTS

System.String

System.Security.SecureString

System.Int32

Microsoft.Azure.Commands.ServiceFabric.Models.OperatingSystem

OUTPUTS

Microsoft.Azure.Commands.ServiceFabric.Models.PSDeploymentResult

NOTES

Example 1: Specify only the cluster size, the cert subject name, and the OS to deploy a cluster

```
$password = "Password#1234" | ConvertTo-SecureString -AsPlainText -Force  
$resourceGroupName = "quickstart-sf-group"  
$azureRegion = "southcentralus"  
$clusterDnsName = "{0}.{1}.cloudapp.azure.com" -f $resourceGroupName, $azureRegion  
$localCertificateFolder = "c:\certs"
```

```
Write-Output "Create cluster in '$azureRegion' with cert subject name '$clusterDnsName' and cert output path  
'$localCertificateFolder'"
```

```
New-AzServiceFabricCluster -ResourceGroupName $resourceGroupName -Location $azureRegion -ClusterSize 5  
-VmPassword $password -CertificateSubjectName $clusterDnsName  
-CertificateOutputFolder $localCertificateFolder -CertificatePassword $pass -OS WindowsServer2016Datacenter
```

This command specifies only the cluster size, the cert subject name, and the OS to deploy a cluster.

Example 2: Specify an existing Certificate resource in a key vault and a custom template to deploy a cluster

```
$resourceGroupName = "test20"  
$templateParameterFile = "C:\service-fabric-secure-nsg-cluster-65-node-3-nodetype\azuredeploytest.parameters.json"  
$templateFile = "C:\azure-quickstart-templates\service-fabric-secure-nsg-cluster-65-node-3-nodetype\azuredeploy.json"  
$secretId = "https://test1.vault.azure.net:443/secrets/testcertificate4/56ec774dc61a462bbc645ffc9b4b225f"
```

```
New-AzServiceFabricCluster -ResourceGroupName $resourceGroupName -TemplateFile $templateFile -ParameterFile  
$templateParameterFile -SecretIdentifier $secretId
```

This command specifies an existing Certificate resource in a key vault and a custom template to deploy a cluster.

Example 3: Create a new cluster using a custom template. Specify a different resource group name for the Key Vault and

have the system upload a new certificate to it

```
$password = "Password#1234" | ConvertTo-SecureString -AsPlainText -Force
$resourceGroupName = "quickstart-sf-group"
$keyVaultResourceGroupName = " quickstart-kv-group"
$keyVaultName = "quickstart-kv"
$azureRegion = "southcentralus"
$clusterDnsName = "{0}.{1}.cloudapp.azure.com" -F $resourceGroupName, $azureRegion
$localCertificateFolder = "~\Documents"
$templateParameterFile = "C:\service-fabric-secure-nsg-cluster-65-node-3-nodetype\azuredeploytest.parameters.json"
$templateFile = "C:\service-fabric-secure-nsg-cluster-65-node-3-nodetype\azuredeploy.json"
```

```
New-AzServiceFabricCluster -ResourceGroupName $resourceGroupName -TemplateFile $templateFile -ParameterFile
$templateParameterFile -CertificateOutputFolder
$localCertificateFolder -CertificatePassword $password -KeyVaultResourceGroupName $keyVaultResourceGroupName
-KeyVaultName $keyVaultName -CertificateSubjectName
$clusterDnsName
```

This command creates a new cluster using a custom template. Specify a different resource group name for the key vault and have the system upload a new certificate to it

Example 4: Bring your own Certificate and custom template and create a new cluster

```
$password = "Password#1234" | ConvertTo-SecureString -AsPlainText -Force
$resourceGroupName = "test20"
$keyVaultResourceGroupName = "test20kvrg"
$keyVaultName = "test20kv"
$localCertificateFile = "c:\Mycertificates\my2017Prodcert.pfx"

$templateParameterFile =
"~\Documents\GitHub\azure-quickstart-templates-parms\service-fabric-secure-nsg-cluster-65-node-3-nodetype\azuredeploytest.parameters.json"
```

"~\GitHub\azure-quickstart-templates\service-fabric-secure-nsg-cluster-65-node-3-nodetype\azuredeploy.json"

```
New-AzServiceFabricCluster -ResourceGroupName $resourceGroupName -TemplateFile $templateFile -ParameterFile  
$templateParameterFile -CertificateFile  
$localCertificateFile -CertificatePassword $password -KeyVaultResourceGroupName $keyVaultResourceGroupName  
-KeyVaultName $keyVaultName
```

This command will let you bring your own Certificate and custom template and create a new cluster.

RELATED LINKS

Online Version: <https://learn.microsoft.com/powershell/module/az.servicefabric/new-azservicefabriccluster>