



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

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

NAME

New-AzWebAppSSLBinding

SYNOPSIS

Creates an SSL certificate binding for an Azure Web App.

SYNTAX

```
New-AzWebAppSSLBinding [-ResourceGroupName] <System.String> [-WebAppName] <System.String> [[-Slot]
<System.String>] [-Name] <System.String> [-CertificateFilePath]
<System.String> [[-SslState] {Disabled | SniEnabled | IpBasedEnabled}] [-CertificatePassword] <System.String>
[-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Confirm] [-WhatIf]
[<CommonParameters>]
```

```
New-AzWebAppSSLBinding [-WebApp] <Microsoft.Azure.Commands.WebApps.Models.PSSite> [-Name]
<System.String> [-CertificateFilePath] <System.String> [[-SslState]
{Disabled | SniEnabled | IpBasedEnabled}] [-CertificatePassword] <System.String> [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Confirm] [-WhatIf]
[<CommonParameters>]
```

```
New-AzWebAppSSLBinding [-ResourceGroupName] <System.String> [-WebAppName] <System.String> [[-Slot]
<System.String>] [-Name] <System.String> [[-SslState] {Disabled |
SniEnabled | IpBasedEnabled}] [-Thumbprint] <System.String> [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Confirm] [-WhatIf]
[<CommonParameters>]
```

```
New-AzWebAppSSLBinding [-WebApp] <Microsoft.Azure.Commands.WebApps.Models.PSSite> [-Name]
<System.String> [[-SslState] {Disabled | SniEnabled | IpBasedEnabled}]
[-Thumbprint] <System.String> [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Confirm] [-WhatIf]
[<CommonParameters>]
```

DESCRIPTION

The New-AzWebAppSSLBinding cmdlet creates a Secure Socket Layer (SSL) certificate binding for an Azure Web App. The cmdlet creates an SSL binding in two ways: - You can bind a Web App to an existing certificate.

- You can upload a new certificate and then bind the Web App to this new certificate.

Regardless of which approach you use, the certificate and the Web App must be associated with the same Azure resource group. If you have a Web App in Resource Group A

and you want to bind that Web App to a certificate in Resource Group B, the only way to do that is to upload a copy of the certificate to Resource Group A. If you

upload a new certificate, keep in mind the following requirements for an Azure SSL certificate: - The certificate must contain a private key.

- The certificate must use the Personal Information Exchange (PFX) format.

- The certificate's subject name must match the domain used to access the Web App.

- The certificate must use a minimum of 2048-bit encryption.

PARAMETERS

-CertificateFilePath <System.String>

Specifies the file path for the certificate to be uploaded. The CertificateFilePath parameter is only required if the certificate has not yet been uploaded to Azure.

Required? true

Position? 4

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-CertificatePassword <System.String>

Specifies the decryption password for the certificate.

Required? true

Position? 5

Default value None

Accept pipeline input? False

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

-Name <System.String>

Specifies the name of the Web App.

Required? true

Position? 3

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ResourceGroupName <System.String>

Specifies the name of the resource group that the certificate is assigned to. You cannot use the ResourceGroupName parameter and the WebApp parameter in the same command.

Required? true

Position? 0

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Slot <System.String>

Specifies the name of the Web App deployment slot. You can use the Get-AzWebAppSlot cmdlet to get a slot. Deployment slots provide a way for you to stage and validate web apps without those apps being accessible over the Internet. Typically you will deploy your changes to a staging site, validate those changes, and then deploy to the production (Internet-accessible) site.

Required? false

Position? 2

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-SslState <System.Nullable`1[Microsoft.Azure.Management.WebSites.Models.SslState]>

Specifies whether the certificate is enabled. Set the SSLState parameter to 1 to enable the certificate, or set SSLState to 0 to disable the certificate.

Required?	false
Position?	4
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

-Thumbprint <System.String>

Specifies the unique identifier for the certificate.

Required?	true
Position?	6
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

-WebApp <Microsoft.Azure.Commands.WebApps.Models.PSSite>

Specifies a Web App. To get a Web App, use the Get-AzWebApp cmdlet. You cannot use the WebApp parameter in the same command as the ResourceGroupName parameter and/or the WebAppName .

Required?	true
Position?	0
Default value	None
Accept pipeline input?	True (ByValue)
Accept wildcard characters?	false

-WebAppName <System.String>

Specifies the name of the Web App for which the new SSL binding is being created. You cannot use the WebAppName parameter and the WebApp parameter in the same command.

Required? true
Position? 1
Default value None
Accept pipeline input? False
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

OUTPUTS

Microsoft.Azure.Management.WebSites.Models.HostNameSslState

NOTES

----- Example 1: Bind a certificate to a Web App -----

```
New-AzWebAppSSLBinding -ResourceGroupName "ContosoResourceGroup" -WebAppName "ContosoWebApp"
-Thumbprint "E3A38EBA60CAA1C162785A2E1C44A15AD450199C3" -Name
"www.contoso.com"
```

This command binds an existing Azure certificate (a certificate with the Thumbprint E3A38EBA60CAA1C162785A2E1C44A15AD450199C3) to the web app named ContosoWebApp.

----- Example 2 -----

```
New-AzWebAppSSLBinding -Name 'www.contoso.com' -ResourceGroupName 'ContosoResourceGroup' -SslState
Disabled -Thumbprint 'E3A38EBA60CAA1C162785A2E1C44A15AD450199C3'
-WebAppName 'ContosoWebApp'
```

powershell

----- Example 3 -----

```
New-AzWebAppSSLBinding -CertificateFilePath <String> -CertificatePassword <String> -Name 'www.contoso.com'  
-ResourceGroupName 'ContosoResourceGroup' -SslState  
Disabled -WebAppName 'ContosoWebApp'
```

RELATED LINKS

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

[Get-AzWebAppSSLBinding](#)

[Remove-AzWebAppSSLBinding](#)

[Get-AzWebAppSlot](#)

[Get-AzWebApp](#)