

How to create a virtual directory in an Azure Web application using PowerShell

In the cloud computing era, creating an IIS virtual directory in PowerShell only required running the `New-WebVirtualDirectory` command. Nowadays, things are done a little differently. Perhaps someday we'll use the `New-AzureWebVirtualDirectory` command in the Azure PowerShell module. However, until then, you'll need to create your own virtual directory. This article from TipsMake will guide you on how to create a virtual directory in your Azure Web application using PowerShell.

The process of creating and assigning a virtual directory to an Azure Web application involves the following three steps:

1. Retrieve the `Microsoft.Azure.Management.WebSites.Models.Site` object.
2. Create object `Microsoft.Azure.Management.WebSites.Models.VirtualApplication`.
3. Assign the `VirtualApplication` object to the `Site` object.



How to create a virtual directory in an Azure Web application using PowerShell

There's a bit of a problem with .NET objects when creating virtual directories, so to fix this and create a script, follow these steps:

First, ensure you have installed the latest AzureRm PowerShell module (`Install-Module -Name AzureRm`). After downloading the latest AzureRm PowerShell module, the next step is to retrieve your web application to mount

the virtual directory. You can use `Get-AzureRmWebApp` to do this by specifying the resource group it belongs to and the name of your web application.

```
$webApp = Get-AzureRmWebApp -Name $AzureWebAppName -ResourceGroupName  
$AzureResourceGroup
```

The next step is to create a `VirtualApplication` object. This object has three properties you need to pay attention to: **VirtualPath** , **PhysicalPath** , and **PreloadEnabled** (optional). The `VirtualPath` and `PhysicalPath` properties are displayed in the Azure Web portal if you view the Application Settings of a web application.

This `VirtualApplication` object is created from the `Microsoft.Azure.Management.WebSites.Models.VirtualApplication` class and has the properties mentioned above set.

```
$virtApp = New-Object Microsoft.Azure.Management.WebSites.Models.VirtualApplication
```

```
$virtApp.VirtualPath = $VirtualPath
```

```
$virtApp.PhysicalPath = $PhysicalPath
```

```
$virtApp.PreloadEnabled = $PreloadEnabled
```

After the `VirtualApplication` object is created, the next step is to assign it to your web application. The `Microsoft.Azure.Management.WebSites.Models.Site` object returned from `Get-AzureRmWebApp` contains a `siteconfig` property.

This project contains a `VirtualApplications` property with an `Add()` method that accepts the `Microsoft.Azure.Management.WebSites.Models.VirtualApplication` object. You can use this `Add()` method to pass the `VirtualApplication` object you just created to your web application.

```
$null = $webApp.siteconfig.VirtualApplications.Add($virtApp)
```

So you've just created a virtual directory and added it to your local Web application. The next step is to commit your changes to the Azure Web application using the `Set-AzureRmWebApp` command . Since the `$webApp` variable already has the virtual directory attached, you can overwrite the existing file with the modified object.

```
Set-AzureRmWebApp -WebApp $webApp
```

Now that you have all the code built to mount a virtual directory to your Azure Web application, you can roll all this code into a single parameterized script and no longer have to worry about it. This script (or function) will reside in the Azure Web repository.

The article on [TipsMake](#) just showed you how to create a virtual directory in an Azure Web application using PowerShell.

As you can see, PowerShell's capabilities are vast. Not only is it available on Azure Web, but PowerShell is also integrated into Windows 10, allowing users to perform many tasks such as backing up drivers on Windows 10 using PowerShell . If you have any questions or need clarification, please leave your comments in the section below the article.

You finished reading the article "**How to create a virtual directory in an Azure Web application using PowerShell**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.
