

Teams Phone Numbers Manager

Guide to redeploy the API

There are occasions where you'll need to redeploy the API on the Azure Functions one example being an update of the PowerShell module for Microsoft Teams. Here are the instructions if you need to deploy a new version of the API on the Azure Functions.

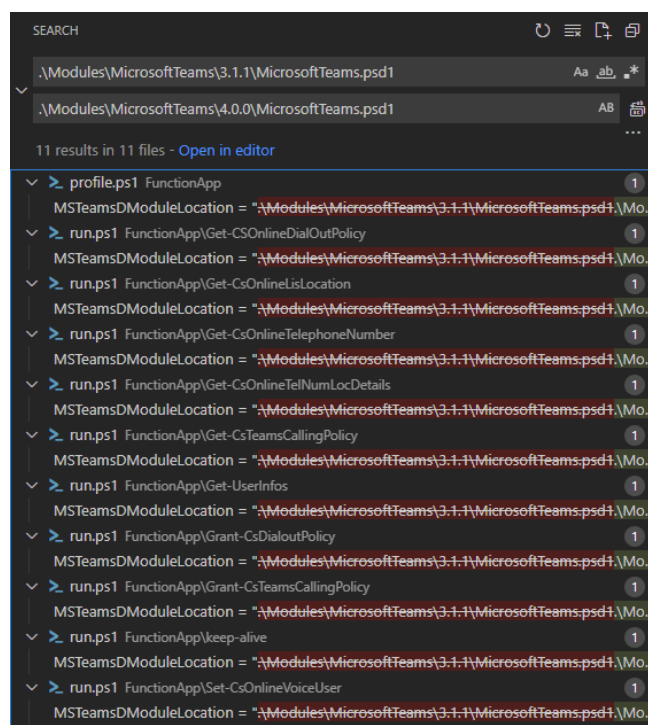
The application has been built and tested against the PowerShell Microsoft Teams module version 3.1.1 and a version 4.0.0 is now available. To upgrade the Azure Functions API to use the latest version of the Microsoft Teams module, here are the steps to follow.

Pre-requisites

- The code of the application is downloaded from GitHub on your local computer
- Install VS Code and the "Azure Tools" extension for VS Code

Steps to update the Microsoft Teams module

1. Download the new module locally
`save-module -Path .\FunctionApp\Modules -Name MicrosoftTeams -Repository PSGallery -MinimumVersion 4.0.0`
2. Delete the folder containing the previous version of the Microsoft Teams module (3.1.1)
Go under `.\FunctionApp\Modules\MicrosoftTeams` and delete folder 3.1.1
You should only keep the folder named 4.0.0
3. In the code, rename all occurrences of
"`.\Modules\MicrosoftTeams\3.1.1\MicrosoftTeams.psd1`"
into
"`.\Modules\MicrosoftTeams\4.0.0\MicrosoftTeams.psd1`"



- Go the Azure portal (<https://portal.azure.com>) and change the configuration setting of the Azure Functions app – Replace the value of the setting “WEBSITE_RUN_FROM_PACKAGE” by “1” (instead of the full URL of the original ZIP package) and press OK to save.

This will allow you to publish your local changes to the Azure Functions app.

Home > tpm-vnext-6cdgs

tpm-vnext-6cdgs | Configuration

Function App

Search (Ctrl+/) Refresh Save Discard Leave Feedback

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Security Events (preview)

Functions

- Functions
- App keys
- App files
- Proxies

Deployment

- Deployment slots
- Deployment Center

Settings

- Configuration
- Authentication

Application settings

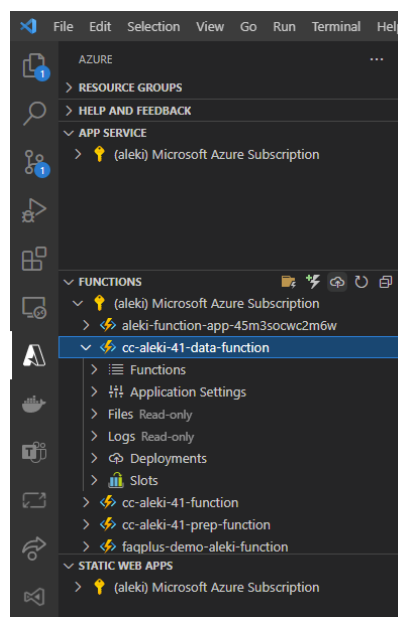
Application settings are encrypted at rest and transmitted over an encrypted channel. You can choose to disp application at runtime. [Learn more](#)

+ New application setting Show values Advanced edit

Filter application settings

Name	Value
AdminAccountLogin	Hidden value. Click to show value
AdminAccountPassword	Hidden value. Click to show value
AzureWebJobsStorage	Hidden value. Click to show value
FUNCTIONS_EXTENSION_VERSION	Hidden value. Click to show value
FUNCTIONS_WORKER_PROCESS_COUNT	Hidden value. Click to show value
FUNCTIONS_WORKER_RUNTIME	Hidden value. Click to show value
FUNCTIONS_WORKER_RUNTIME_VERSION	Hidden value. Click to show value
MICROSOFT_PROVIDER_AUTHENTICATION_SECRET	Hidden value. Click to show value
WEBSITE_RUN_FROM_PACKAGE	1

- Publish the changes to Azure using VS Code
With the Azure Tools for VS Code, connect to your Azure account, select your Azure subscription and the Azure Functions target for the deployment and then press “Deploy to Azure Function”



6. Once the deployment is over, it is recommended to run the warm-up script – This will warm-up the Azure Functions – It usually takes ~5 – 10 min to run this script.

To run the script, you need the following information:

```
$hostname = the domain of your Azure Function
$code      = the code of your Azure Function
$tenantID  = your Azure AD tenant ID
$clientID  = the client ID of your app registration in Azure AD
$secret    = the client secret of your app registration in Azure AD
```

Example:

```
$hostname = 'teams-mng-ks6gh.azurewebsites.net'
$code     = 'ZbIvkeirB7xBXNXXXXXXXXXXXXXXXXXXXXiFxFaNwkzt7g=='
$tenantID = '720b637a-XXXX-XXXX-XXXX-f22f40755c2c'
$clientID = 'af9ebcdd-XXXX-XXXX-XXXX-7741f668e969'
$secret   = '9sT5JCGWXXXXXXXXXXXXXXXXXXXXXe1DW2EhyI='
```

A successful outcome looks like this:

```
.\Deployment\warmup.ps1 -hostname $hostname -code $code -tenantID $tenantID -clientID $clientID -secret $secret
Azure Function warm-up using API call
https://teams-mng-ks6gh.azurewebsites.net/api/Get-CsTeamsCallingPolicy
Access token request success
Function warm-up started at 18/03/2022 18:43:51 - Attempt # 1
```

TriggerTime	WorkerId	Duration	StatusCode	StatusDescription
18/03/2022 18:44:06	3	14,9	200	OK
18/03/2022 18:44:06	2	14,91	200	OK
18/03/2022 18:44:07	1	15,89	200	OK

The script will run up to 3 attempts.