

Innovation Foundation Deployment Guide

**Prepared By**

**Microsoft Services**

MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user.  Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document.  Except as expressly provided in any written license agreement from Microsoft, our provision of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The descriptions of other companies’ products in this document, if any, are provided only as a convenience to you.  Any such references should not be considered an endorsement or support by Microsoft.  Microsoft cannot guarantee their accuracy, and the products may change over time. Also, the descriptions are intended as brief highlights to aid understanding, rather than as thorough coverage. For authoritative descriptions of these products, please consult their respective manufacturers.

© 2017 Microsoft Corporation. All rights reserved. Any use or distribution of these materials without express authorization of Microsoft Corp. is strictly prohibited.

Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

ii

Revision and Sign-Off Sheet

Change Record

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Author | Version | Change Reference |
| 9-Oct-17 | Tony Carmalitano | 0.1 | First draft |
| 19-Dec-17 | John Gemkow | 1.0 | Second revision |
| 10-Jan-18 | John Gemkow | 1.1 | Key Vault implementation |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[1 Introduction 5](#_Toc511231283)

[2 Prerequisites 6](#_Toc511231284)

[3 Assumptions 7](#_Toc511231285)

[4 Directory Setup 8](#_Toc511231286)

[4.1 Configure a new AAD tenant (optional) 8](#_Toc511231287)

[4.2 Connect to existing AAD tenant 10](#_Toc511231288)

[4.3 Update Tenant ID in Inputs.xml 10](#_Toc511231289)

[4.4 Customize Inputs.xml 12](#_Toc511231290)

[4.5 Get Tenant and Profile Info 15](#_Toc511231291)

[5 Create the host SharePoint Site Collection 16](#_Toc511231292)

[6 Prepare Config Data 17](#_Toc511231293)

[6.1 Yammer FeedId 17](#_Toc511231294)

[7 Azure Deployment 19](#_Toc511231295)

[7.1 Preparing Azure Resources 19](#_Toc511231296)

[7.2 Granting permissions to SharePoint app ids 20](#_Toc511231297)

[7.3 Permissions & Registering the App (optional) 24](#_Toc511231298)

[7.4 Deploy SharePoint host web resources 26](#_Toc511231299)

[8 PowerBI 27](#_Toc511231300)

[8.1 Configuring Data Sources 27](#_Toc511231301)

[8.2 Create an App Workspace. 31](#_Toc511231302)

[8.2.2 Publishing Reports 34](#_Toc511231303)

[9 Update Azure Web App Configuration 38](#_Toc511231304)

[10 Subsite Master Page 39](#_Toc511231305)

[11 On-Premises Deployment 40](#_Toc511231306)

# Introduction

The Innovation Foundation Deployment Guide is intended to provide the step-by-step instructions for installation of the solution components required for the MCS Innovation Foundation offer. There may be additional configurations required to tailor the solution to the customer-specific needs (data storage in SQL for example). Those configurations are considered out of scope for this document.

Please be aware that this deployment guide will deploy a vanilla deployment that has been modeled off the BP Ideas deployment. For example, the logo may be “Business Productivity Innovation.” Additionally, the idea statuses and triages may be based on the time zone to global approver model we employ. You should check that the files under **Post\Files\Data** are customized to meet the requirements for data and process in your deployment.

# Prerequisites

For an **Azure deployment**, the following tools must be installed:

* [Azure PowerShell](https://portal.azure.com/)
* [Microsoft Online Services Sign-In Assistant for IT Professionals](https://www.microsoft.com/en-my/download/details.aspx?id=39267)
* [SharePoint Online Management Shell](https://portal.azure.com)
* [AAD Module for PowerShell](https://azure.microsoft.com/en-us/regions/?DownloadID=59185) (V1)
* [SharePoint PnP PowerShell Online](https://docs.microsoft.com/en-us/powershell/sharepoint/sharepoint-pnp/sharepoint-pnp-cmdlets?view=sharepoint-ps)
* [Microsoft PowerBI Desktop](https://www.microsoft.com/en-in/download/details.aspx?id=45331)

Please note that this guide was updated to be compatible with Azure PowerShell version 5.0.0. Using a different version of Azure PowerShell modules may result in deployment errors due to incompatible or deprecated commandlets.

The AAD Module for PowerShell is needed to utilize the MSOnline-based commandlets. This guide has not been tested with version 2 nor are the scripts written to yet support version 2 commandlets.

# Assumptions

This document assumes the following for an **Azure deployment**:

* User is a global administrator for the AAD tenant they wish to deploy the application on
* User has an active Azure subscription
* User has access to the AAD tenant they wish to deploy the application on, whether that be an O365-created tenant or otherwise
* SharePoint App and SharePoint Add-in are used interchangeably
* All PowerShell instances/scripts are **run as an administrator**
* The user has installed all the prerequisite components listed in the prerequisite section
* The user has installed the following PowerShell Modules per the prerequisites:
  + *Azure*
  + *AzureRM (version 5.0.0)*
* The user imports the following modules per PowerShell instance:
  + *Azure*
  + *AzureRM.Resources (version 5.0.0)*
* All script and file paths assume you are in the **Deployment-PowerShell** folder; ensure you have the following scripts:
  + *AzureCreateAllResources.ps1*
  + *AzureDeleteAllResources.ps1*
  + *AzurePublishWebApp.ps1*
  + *AzureUpdateWebConfigAppSettings.ps1*
  + *GetAzureCredential.ps1*
  + *SetDefaultSub.ps1*
  + *SharePoint-SiteColl.ps1*
* Customizations to data to be loaded have been completed as necessary or a vanilla/standard deployment will be completed.

For an **on-premises deployment**, the user should have the existing SharePoint farm and server configuration required. Please note that if you are installing this application in an on-premises environment, that your instructions begin at 7: On-Premises Deployment on page 33.

# Directory Setup

To utilize the script to correctly configure the application on Azure Active Directory, you will need access to the AAD Tenant. This allows the script to configure permissions for accessing AAD, Microsoft Graph, PowerBI, etc.

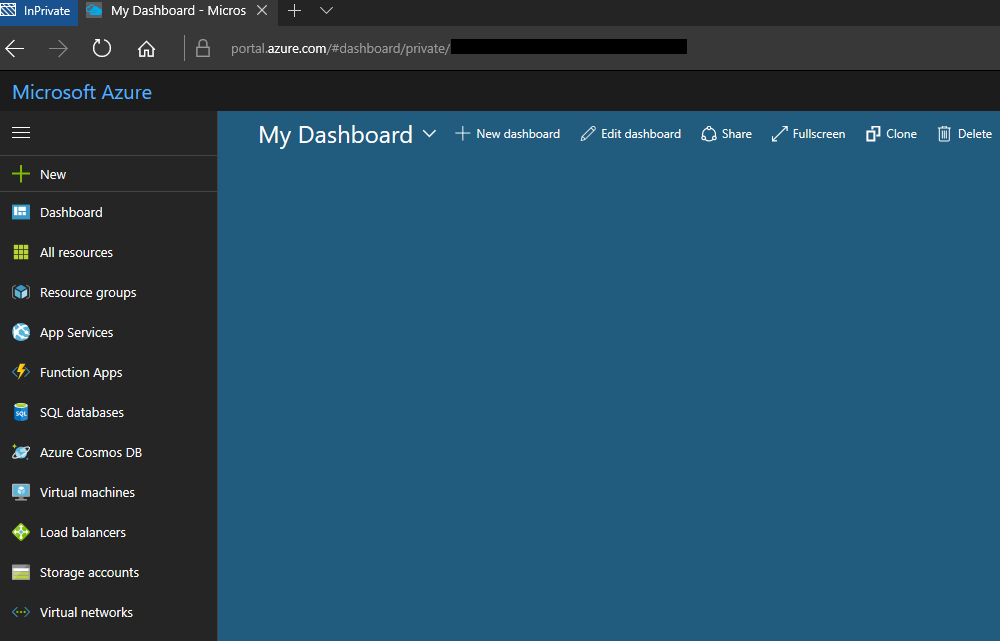
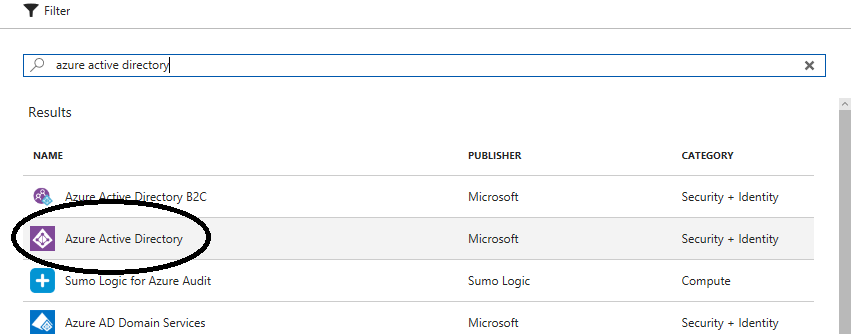
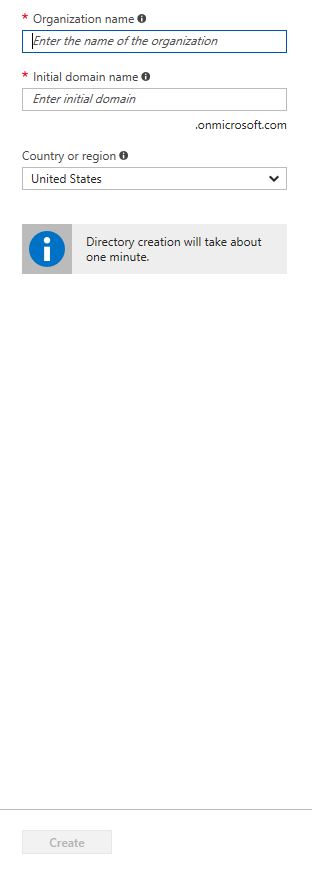
**Note:** If the directory you wish to use for this application already exists, please skip to “Update Tenant ID”.

Two options exist for using AAD:

1. Configure a new AAD tenant
2. Connect to existing AAD tenant (an O365 created AAD tenant for example)

Following tenant configuration, you’ll need to update the Inputs.xml file with configuration values. Lastly, to complete directory setup, run two the Get-AzureCredential.ps1 script to save your Azure profiles for simpler access.

## Configure a new AAD tenant (optional)

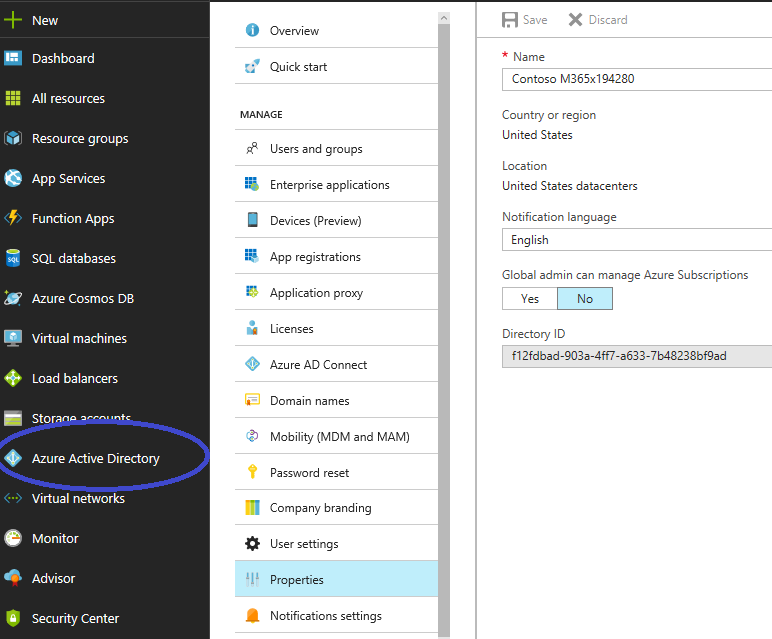
1. Sign in to the [Azure Portal](https://www.microsoft.com/web/handlers/webpi.ashx/getinstaller/WindowsAzurePowershellGet.3f.3f.3fnew.appids) with the global administrator credentials for your tenant.
2. Click **+New** in the left navigation panel.  
     
     
   
3. Search for **Azure Active Directory**. Select it, and then click **Create**.   
     
     
     
   
4. Provide a valid name for the organization and initial domain name, then click **Create**.  
   

## Connect to existing AAD tenant

If you’d like to switch your subscription to another AAD tenant (for consolidation and Key Vault), you can follow the guide recently published here: [https://docs.microsoft.com/en-us/azure/active-directory/active-directory-how-subscriptions-associated-directory](https://www.microsoft.com/en-my/download/details.aspx). It is highly encouraged to have your Azure subscription in the same AAD tenant.

**Important: Azure Key Vault does not support subscriptions in a tenant different than the tenant hosting AAD.**

## Update Tenant ID in Inputs.xml

1. After your directory is created, go to **Azure Active Directory** in the left panel and click **Properties**.  
     
   
2. Copy **Directory ID** and update tenantId in **Inputs.xml** in the root of the Deployment-PowerShell folder.

## Customize Inputs.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <inputs>  <!--# Update with the name of your subscription.-->  <SubscriptionName value="Azure Pass" />  <!--#Unique Key for This instance: Alphanumeric, lowercase only-->  <thisInstance value="ifm365x577711" />  <!--# Choose an Azure Location for resources-->  <Location value="South Central US" />  <!--# Choose an Azure Location for AppInsight resource-->  <!--#East US, North Europe, West Europe, South Central US-->  <AppInsightLocation value="South Central US" />  <!--SharePoint-->  <demoSiteURL value="sites/Innovation.Foundation" />  <demoUserUPN value="JordanM@M365x577711.onmicrosoft.com" />  <adminUPN value="admin@M365x577711.onmicrosoft.com" />  <adminpassword value="password" />  <orgName value="m365x577711" /><!--LowerCase-->  <!--Azure AAD tenant name-->  <tenantId value="8b9c01c8-1cc7-4bc1-a0ee-8d8be71a1019" />  <azureAccountName value="john@lab.johngemkow.com" /><!-- deprecated -->  <azurePass value="" /><!-- deprecated -->  <!--Enable/disable Workflow-->  <Workflow value="1"/>  <!--Enable/disable RedisCache-->  <RedisCache value="1"/>  <!--Enable/disable Key Vault-->  <KeyVault value="1"/>  </inputs> |

Update each of the fields\* in **Inputs.xml** to match that of your Azure subscription name, the/your desired SharePoint site, and the Azure directory matching tenantID. Etc.  
  
\***Important:** thisInstance is an ID to be picked by the user. It is also in part responsible for how resources are named; for example, the storage account becomes *saifm365x577711*. As such, should this conflict with a storage account that exists with the same name, setup for the resources will fail and you will have to come up with a new ID.

The following table will explain the key/value pairs:

|  |  |  |
| --- | --- | --- |
| Key | Description | Example Value |
| Subscription Name | Azure subscription name | Visual Studio Enterprise |
| thisInstance | Unique ID to be used for all resource creations  Recommended to be if<tenant\_name>  Must be alphanumeric, lowercase, less than 19 characters | ifm365x577711 |
| Location | Azure [location](https://aka.ms/rps-not-found) to be used for resources | South Central US |
| AppInsightLocation | Same as location; however, not all regions offer the service. **Leave blank if no AppIn** | South Central US |
| DemoSiteURL | Url of Site collection to install on (server relative) | Sites/Innovation.Foundation |
| DemoUserUPN | UPN of a demo user in the environment | JordanM@M365x577711.onmicrosoft.com |
| AdminUPN | UPN of global admin in AAD environment [used for config and email settings. Leave blank if prompt desired | admin@M365x577711.onmicrosoft.com |
| AdminPassword | Password for above global admin. Leave blank if prompt desired | Pa$$w3rd |
| TenantID | TenantID for AAD directory | 8b9c01c8-1cc7-4bc1-a0ee-8d8be71a1019 |
| AzureAccountName | Deprecated |  |
| AzurePass | Deprecated |  |
| WorkFlow | Switch to enable workflow web job deployment | 1 or 0 |
| RedisCache | Switch to enable RedisCache deployment | 1 or 0 |
| KeyVault | Switch to enable Key Vault deployment | 1 or 0 |

**Credentials will be stored in Azure Key Vault if KeyVault is set to 1.** However, this is only possible when the subscription used to provision Azure resources and tenant for Azure AD are in the same tenant. If this is not the case, you will need to manually set the admin UPN and password in the Web.config, Innovation.Foundation.DeploymentHelper.config, and Innovation.Foundation.Workflow.config files.

Leaving the adminUPN or adminPassword blank/empty in the XML file will prompt the administrator to enter credentials during the script.

## Get Tenant and Profile Info

* Run GetAzureCredential.ps1

*These scripts help load some data files with regards to your specific configuration – the data files (in* ***Profile\*** *- see AzureResources-azureprofile.json and AADResources-azureprofile.json) are then used later to automate the app deployment and resource setup.*

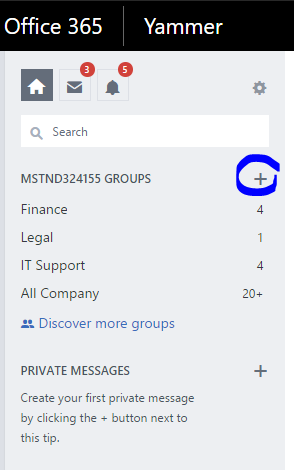
# Create the host SharePoint Site Collection

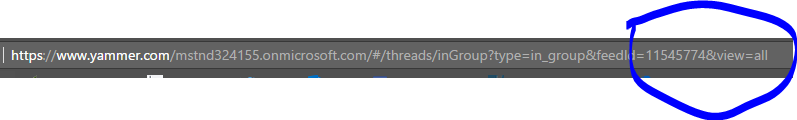
* Run SharePoint-SiteColl.ps1

*This script will connect to SPO and create the host site collection (and delete it if it already exists). The script will also enable the SharePoint Server Publishing Infrastructure, Developer Site Collection Feature, Team Collaboration Lists, and SharePoint Server Publishing features on the site collection and web.*

# Prepare Config Data

## Yammer FeedId

Create a new Yammer Group. Login to your Yammer network homepage (https://www.yammer.com/{yourOrgID}.onmicrosoft.com/#/home).  
  
Click the plus (+) sign in the left column.  
  


Click **Create a New Group**. Select **Internal Group** and then enter a name. The name should be clear and recognizable (i.e. “Innovation Foundation Discussion Group”) to be identifiable to users on Yammer. Select **Public Access** and then create the group.  
  
Copy **FeedID** from the URL:  
  


Update Yammer.InnovationFoundationGroup in **Config\ConfigData.xml** and in **Config\PostConfig.xml**.

# Azure Deployment

## Preparing Azure Resources

Run **AzureCreateAllResources.ps1** to create the required Azure resources.

This script will do the following:

* Create Azure resource group
* Add a storage account, redis cache (if enabled), web app, application insights resource (if location is provided), app service plan to the resource group
* Deploy the web package to the web app
* Create client ID/secret pairs for SharePoint applications using the AAD commandlets (Msonline)
* Create an Azure Active Directory application registration for use in OpenID
* Update the configuration files for additional deployment tools
* Add a web app, web job collection, and web job (if enabled)

**Note:** you may need to register some providers in your Azure Portal for the resources (notably Application Insights) to be created properly, especially if there are errors along the lines of “the subscription is not entitled to the resource”. This can be caused by your subscription not having assigned permissions to create a certain type of resource.

An example of the error that may signal you need to register providers is below:

New-AzureRmResource : MissingSubscriptionRegistration : The subscription is not registered to use namespace 'microsoft.insights'. See [https://aka.ms/rps-not-found](https://docs.microsoft.com/en-us/azure/active-directory/active-directory-how-subscriptions-associated-directory) for how to register subscriptions.

At <deployment location>\Deployment-PowerShell\Deployment-PowerShell\AzureCreateAllResources.ps1:184 char:17

+ ... $resource = New-AzureRmResource -ResourceName $AppInsightsName -Reso ...

+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

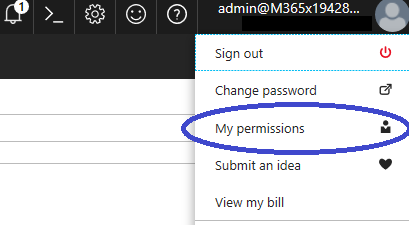
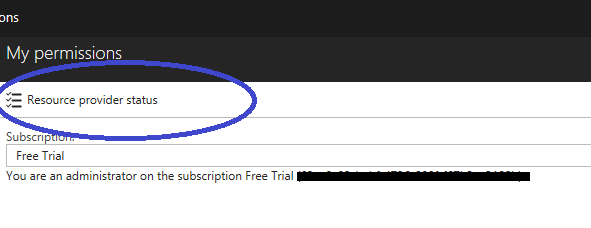
+ CategoryInfo : CloseError: (:) [New-AzureRmResource], ErrorResponseMessageException

+ FullyQualifiedErrorId : MissingSubscriptionRegistration,Microsoft.Azure.Commands.ResourceManager.Cmdlets.Implementation.NewAzureResourceCmdlet

You can solve this by entering the following command:

Register-AzureRmResourceProvider -ProviderNamespace Microsoft.Insights

You can access a GUI of registered resources by going to the Azure Portal, selecting your account, from the dropdown selecting **My Permissions** and then selecting **Resource Providers**.

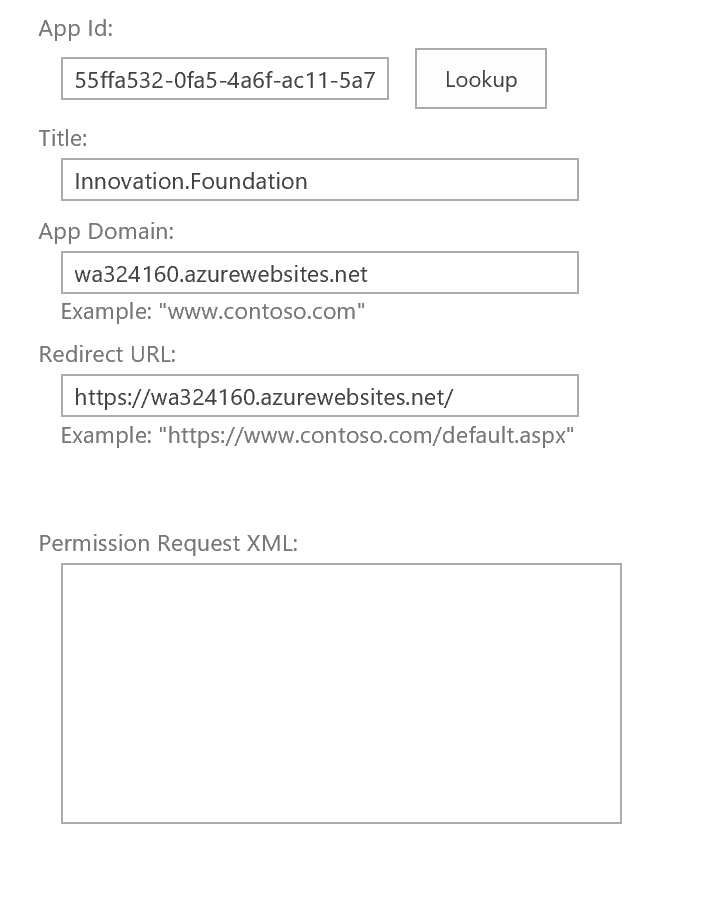
  
  
****

## Granting permissions to SharePoint app ids

1. Go to https://<SharePointWebsite>/\_layouts/15/AppInv.aspx

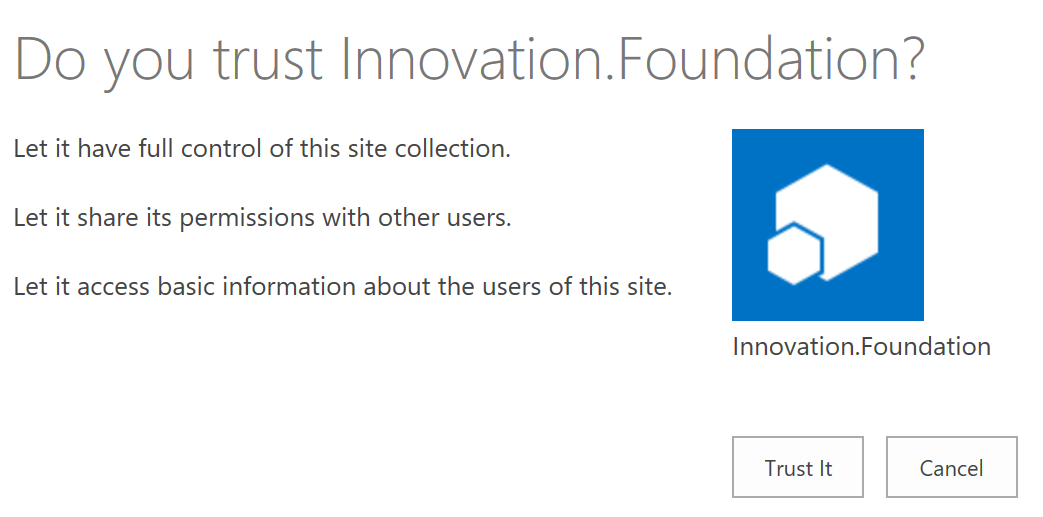
Example: https://m365x111111.sharepoint.com/sites/InnFo/\_layouts/15/AppInv.aspx

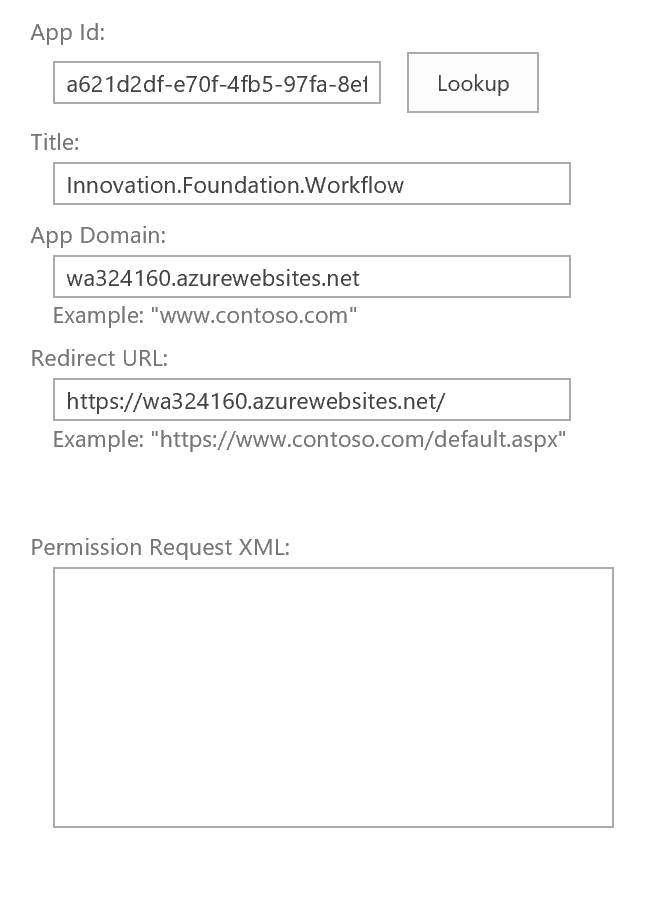
1. (Again) Go to http://<SharePointWebsite>/\_layouts/15/AppInv.aspx
2. Get ClientIdApp from **Log\<OrgName>-output.xml**
3. Enter the add-in's ID, also called the client ID, in the **Add-in Id** box and click **Lookup**. The other boxes on the form are then populated with information about the add-in.



1. Paste the following **Permissions Request** XML to grant full control permission.

|  |
| --- |
| <AppPermissionRequests>  <AppPermissionRequest Scope="http://sharepoint/content/sitecollection" Right="FullControl" />  </AppPermissionRequests> |

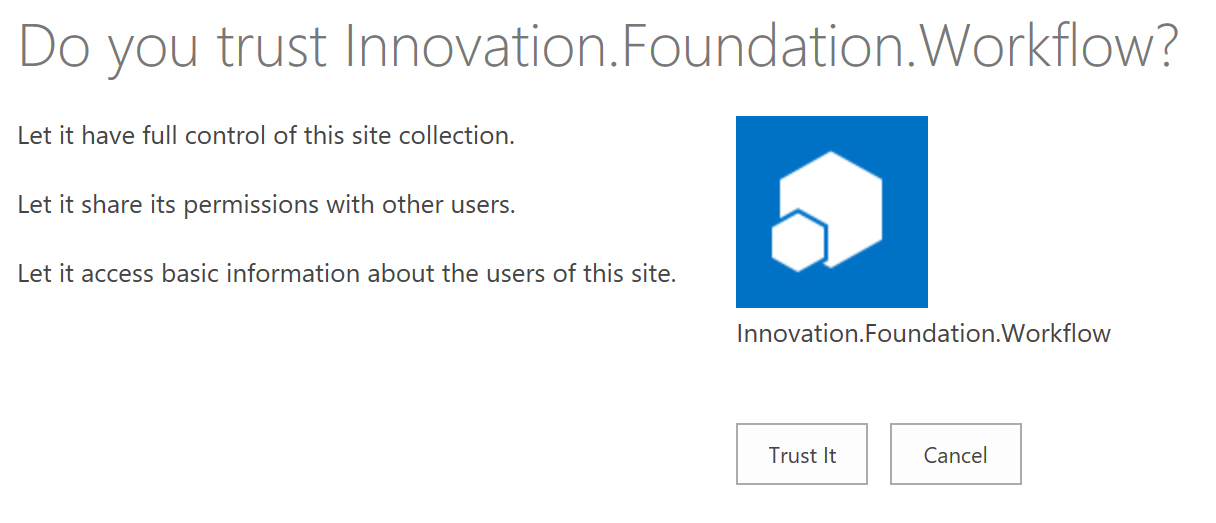
1. Click **Create**.
2. You will then be asked to trust the Innovation.Foundation, as shown in the figure. Click **Trust It**. 
3. If you are not installing the Workflow, skip to 6.3.
4. Get ClientIdAppOnly (**NOT** ClientIdApp) from **Log\<OrgName>-output.xml**
5. Enter the add-in's ID, also called the client ID, in the **Add-in Id** box and click **Lookup**. The other boxes on the form are then populated with information about the add-in.



1. Paste the following **Permissions Request** XML to grant full control permission.

|  |
| --- |
| <AppPermissionRequests AllowAppOnlyPolicy="true">  <AppPermissionRequest Scope="http://sharepoint/content/sitecollection" Right="FullControl" />  </AppPermissionRequests> |

1. Click **Create**.
2. You will then be asked to trust the Innovation.Foundation.Workflow, as shown in the figure. Click **Trust It**



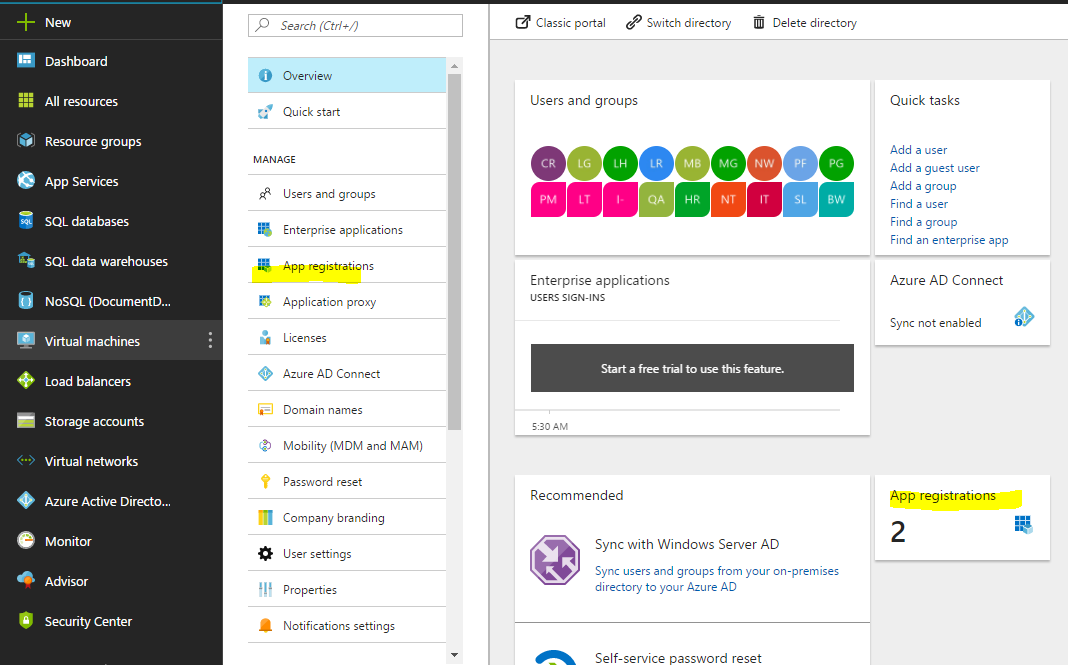
## Permissions & Registering the App (optional)

The script to create Azure resources will assign the necessary permissions to the Azure AD application registration. However, it does not grant the permissions on behalf of all users in the organization. If you do not want the user to be prompted to grant permissions, you can grant on behalf of them with the following steps. Keep in mind, **this is not recommended for most scenarios.**

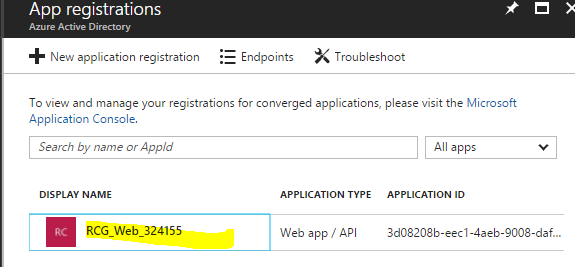
1. Go to the [Azure portal](http://connect.microsoft.com/site1164/Downloads/DownloadDetails.aspx)
2. Click on **Azure Active Directory** Menu



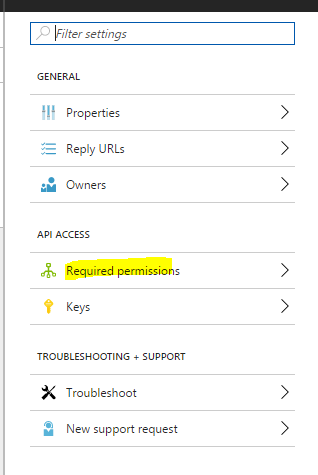
1. Click on **App Registration**



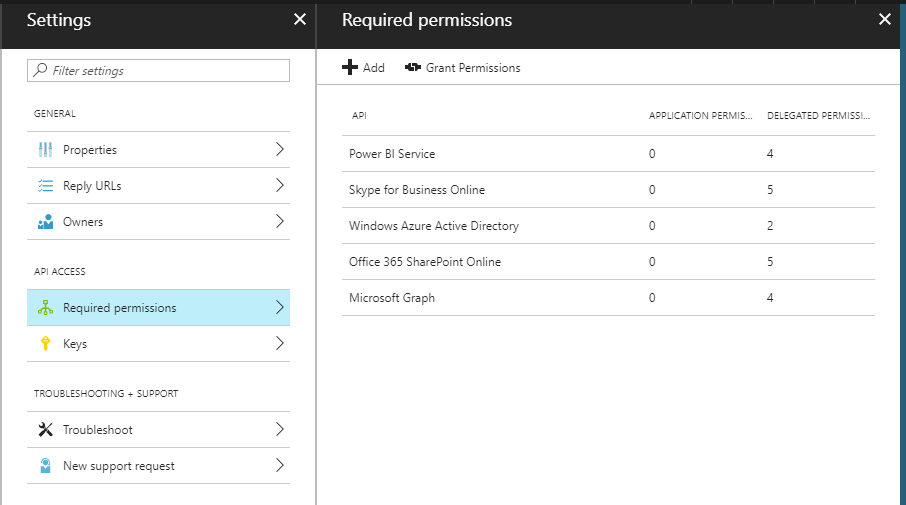
1. Click on **InnFo\_Web\_<instancename>** App



1. Click **Required Permissions**



1. Click on the **Grant Permissions** button



1. Select **Yes**.

## Deploy SharePoint host web resources

This step will deploy all the necessary host web resources (lists, permissions, app package, pages, and page layouts to SharePoint.

* 1. Run **Deployment-PowerShell\Post\Innovation.Foundation.DeploymentHelper.exe** to create SharePoint lists and insert dummy data.
  2. Run **Deployment-PowerShell\Workflow\Innovation.Foundation.Workflow.exe** to test workflow functionality (if enabled)

**Note:** If you run into an issue attempting to connect to Key Vault to get credentials (i.e. Access Denied). Run the following in PowerShell:

Login-AzureRmAccount

Set-AzureRmKeyVaultAccessPolicy -VaultName <VaultName> -ServicePrincipalName <Azure AD App Client ID>

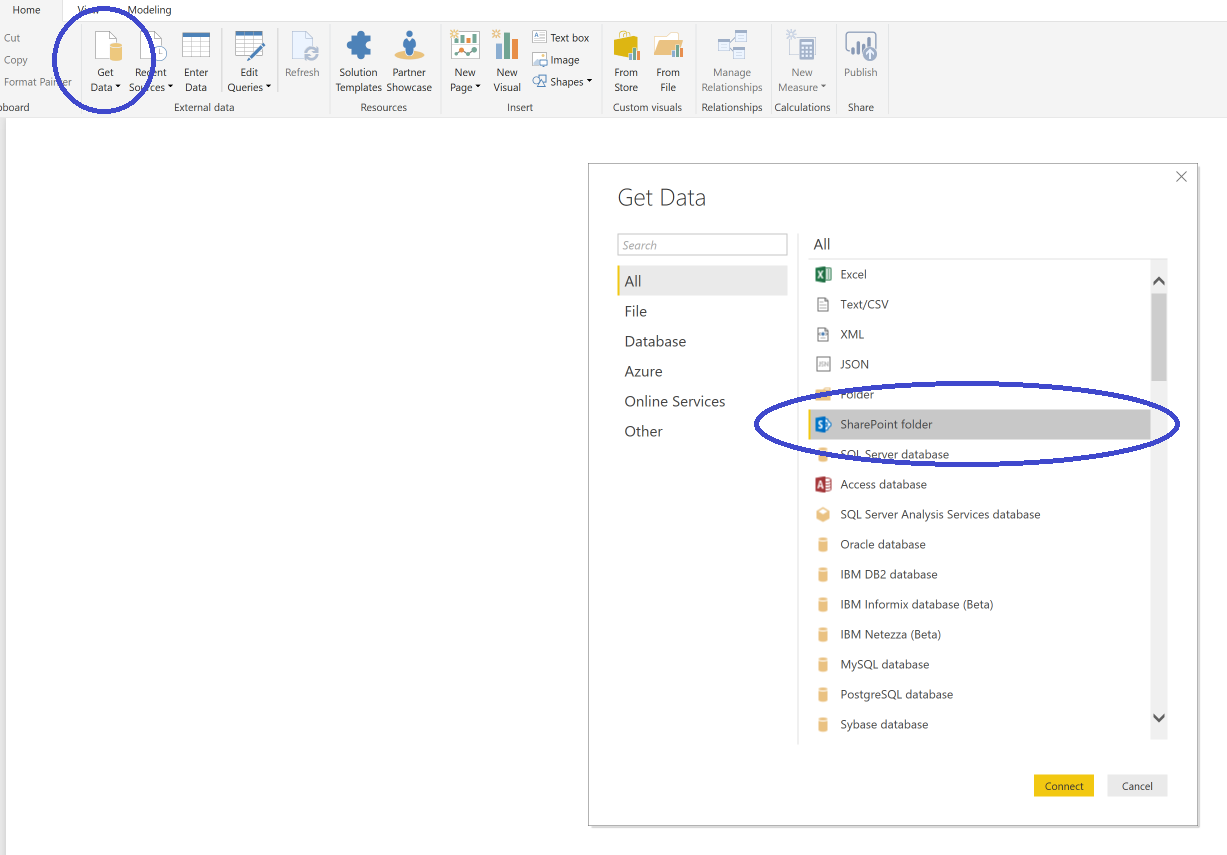
Where the Vault Name is typically kv<thisInstance>.

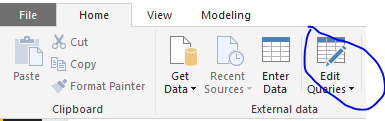
# PowerBI

PowerBI offers reporting functionality for this application. This feature is only available when deployed in SharePoint Online (requiring an Azure Active Directory on the backend).

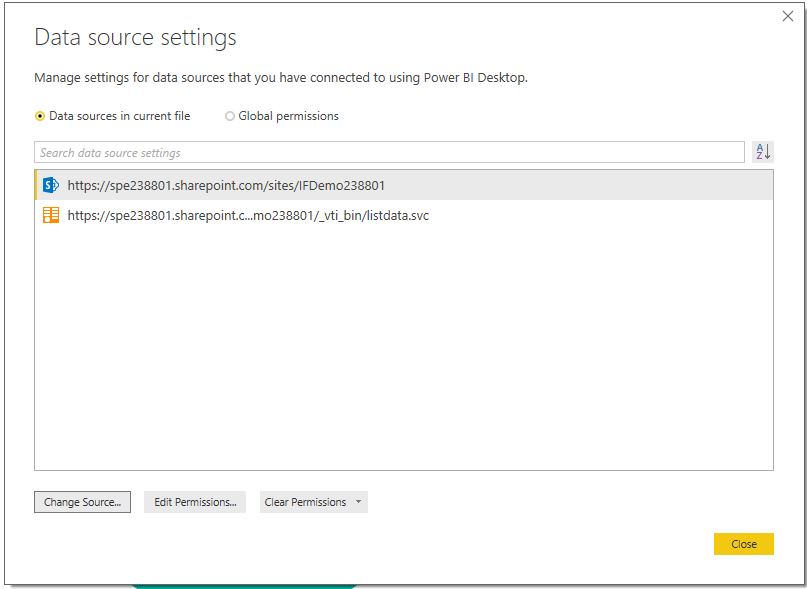
All resources (Azure and SharePoint host web lists) must be deployed prior to this stage

## Configuring Data Sources

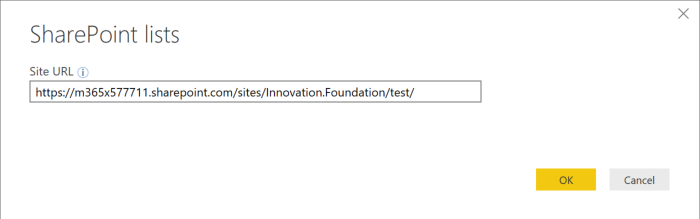
1. In a blank report in PowerBI (Desktop), start a new report. Select **Get Data**, select **More**, and then select **SharePoint Folder**. Enter the URL for your SharePoint site (e.g. *https://contoso.sharepoint.com/sites/demo*).  
     
   
2. **Select Edit Queries.**



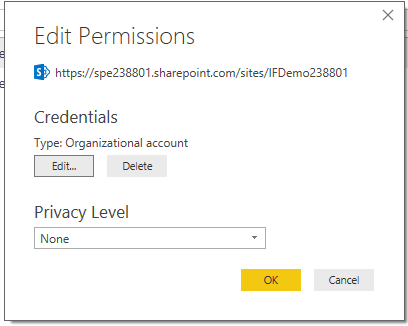
1. Open **Data Source Settings**.



1. Click **Change Source** and enter your site URL. Hit **OK**



1. Click **Edit Permission** followed by **Edit**



1. Login with your Organizational account.
2. Hit **Apply Changes**

If you see an error like the one below, go to **Edit Queries** and click **Go to Error**. In the **Edit Settings** button, you’ll need to specify the resource to navigate to (i.e. Ideas). Lastly, hit **Close & Apply**.

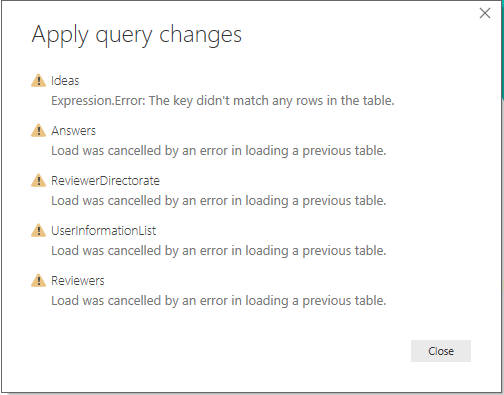


Figure 1 Error example

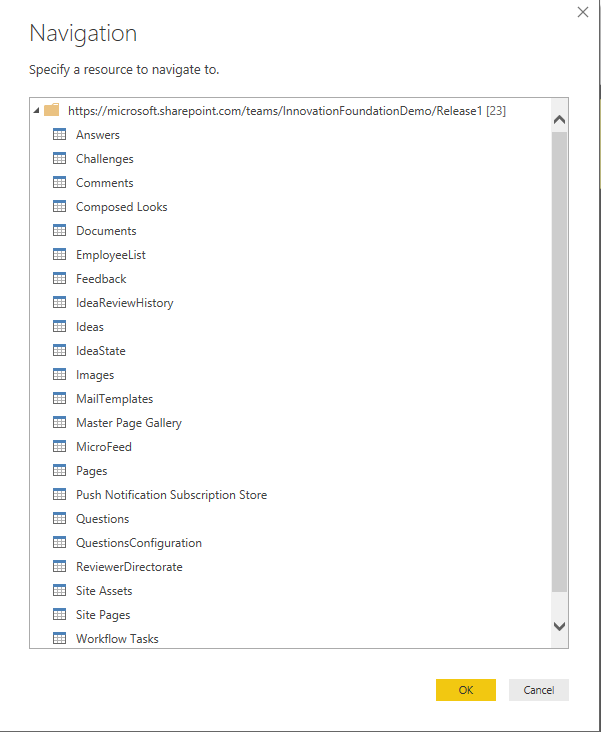
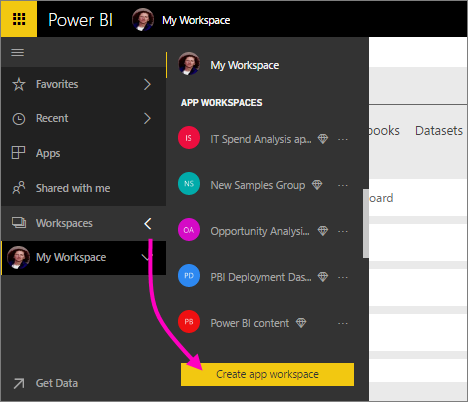


Figure 2 Edit Settings

## Create an App Workspace.

If you utilize PowerBI online as part of an Office 365 subscription, you can create an app workspace to collaborate with a team on the reports.

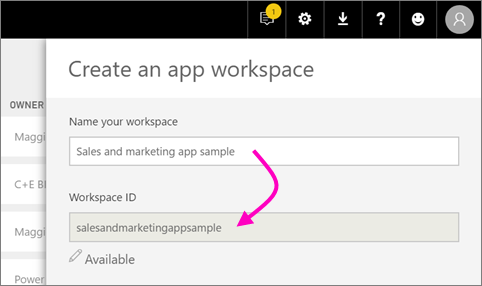
1. Start by creating the workspace in PowerBI Online. Select **Workspaces** > **Create a workspace**.



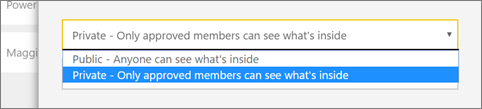
This will be the place to put content that you and your colleagues collaborate on.

1. Give the workspace a name. If the corresponding **Workspace ID** isn't available, edit it to come up with a unique ID.

This will be the name of the app, too.



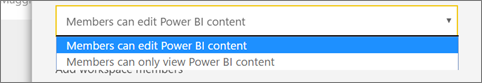
1. You have a few options to set. If you choose **Public**, anyone in your organization can see what’s in the workspace. **Private**, on the other hand, means only members of the workspace can see its contents. Keep in mind that this will influence the ability for administrators to view reports within the app.



##### **Note:**

  You can't change the Public/Private setting after you've created the group.

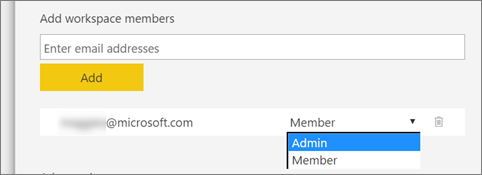
1. You can also choose if members can **edit** or have **view-only** access



##### **Tip:**

  If you're adding someone to the app workspace, it should be so they can edit the content. If they're only going to view the content, don't add them to the workspace. You can include them when you publish the app.

1. Add email addresses of people you want to have access to the workspace, and select **Add**. You can’t add group aliases, just individuals.
2. Decide whether each person is a member or an admin.

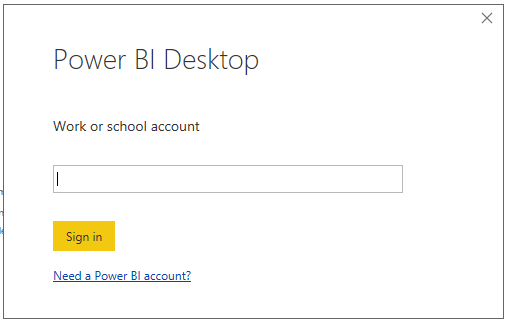


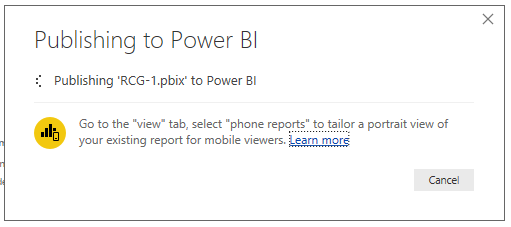
Admins can edit the workspace itself, including adding other members. Members can edit the content in the workspace, unless they have view-only access. Both can publish the app.

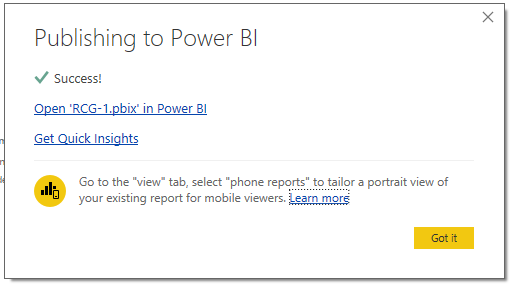
1. Select **Save**.

### Publishing Reports

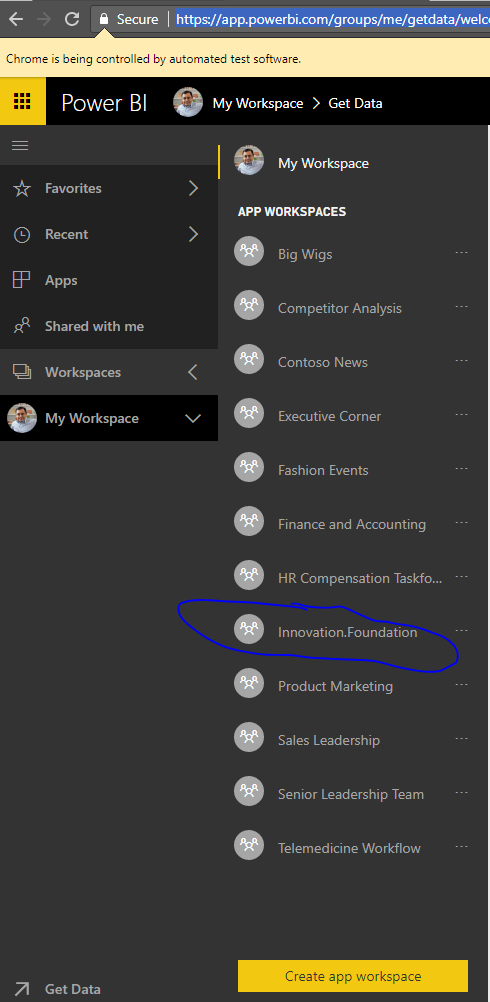
1. In Power BI Desktop > **File** > **Publish** > **Publish to Power BI** or click **Publish** on the ribbon. And publish in the workplace that you just created.  
   https://dpspowerbi.blob.core.windows.net/powerbi-prod-media/powerbi.microsoft.com/en-us/documentation/articles/powerbi-desktop-upload-desktop-files/20170302124441/pbid_publish_publishbutton.png
2. Sign in to Power BI.



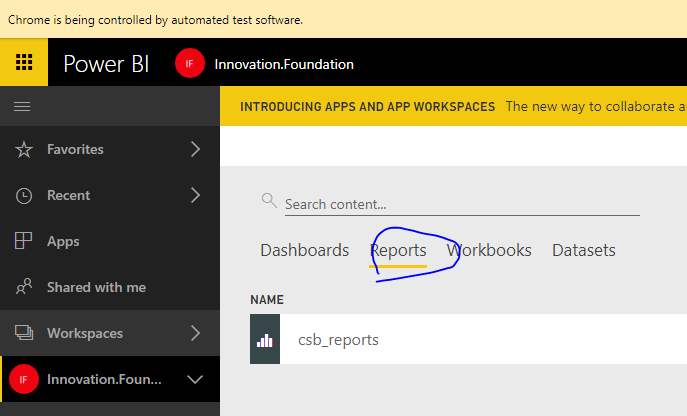


1. When complete, you'll get a link to open your report in your Power BI site.  
   

In PowerBI go to your report - <https://app.powerbi.com/groups/me/getdata/welcome>



Clicks on Reports



Click on the report; the URL should be relatively similar to this:

https://app.powerbi.com/groups/bc9f6286-f8e2-4744-b46a-d0817ec0cbb3/reports/ec0686c7-a09a-41bb-a819-593e419e10d9/ReportSection   
  
  
Update PowerBI.ReportID (reports/**ec0686c7-a09a-41bb-a819-593e419e10d9** just GUID from URL) and PowerBI.WorkspaceID (groups/**bc9f6286-f8e2-4744-b46a-d0817ec0cbb3** just GUID from URL) key value in **Config\ConfigData.xml** and **Config\PostConfigData.xml** and **in Web.config**.

# Update Azure Web App Configuration

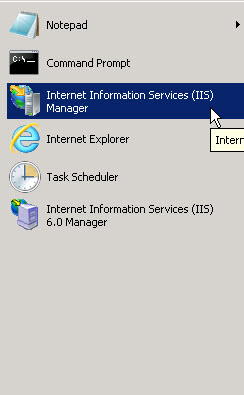
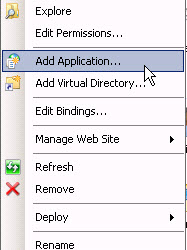
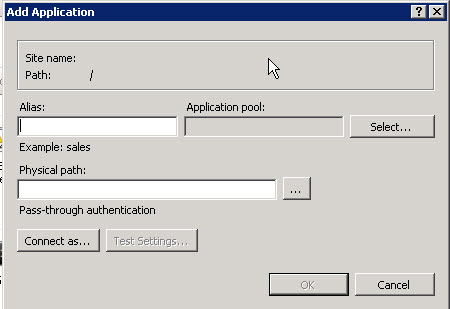
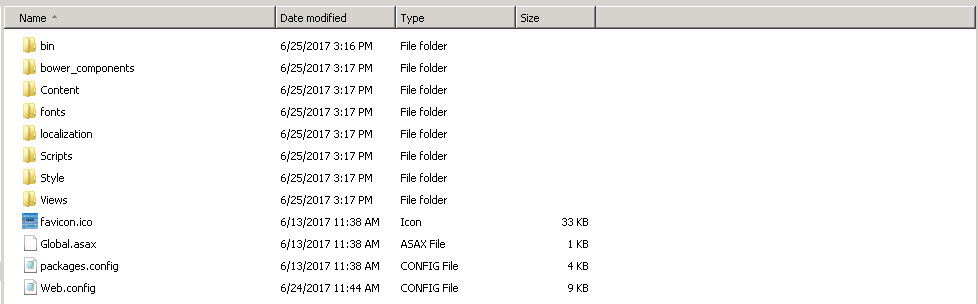
Run **AzureUpdateWebConfigAppSettings.ps1** to update the app settings in web.config and in the Application Settings on Azure for the web application.

# Subsite Master Page

If deploying to a subsite, currently the application deployment process does not have the ability to set the master page to the InnHome.aspx page to force users in the site collection to the app. The following steps will solve this issue. This is unnecessary if deployed in a site collection.

* 1. Go to **Site Settings** using the cog in the upper right on the top bar
  2. Click **Master Page**
  3. Set the master page to “InnHome.aspx”

# On-Premises Deployment

1. Open the IIS Web Manager.  
     
     
   
2. Set up a site for the application and then add the application to the new site.  
     
     
     
     
     
     
   
3. Add the **web.config** from **Deployment-PowerShell\Config** to the site contents.  
     
   
4. Edit **Web.config** – under <appSettings>, add the following line:  
     
   