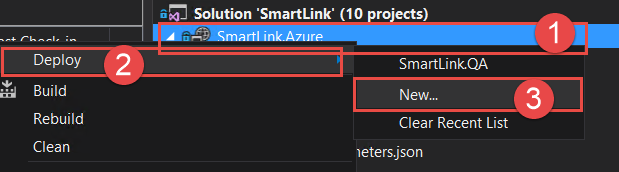
# SmartLink Deployment Guide

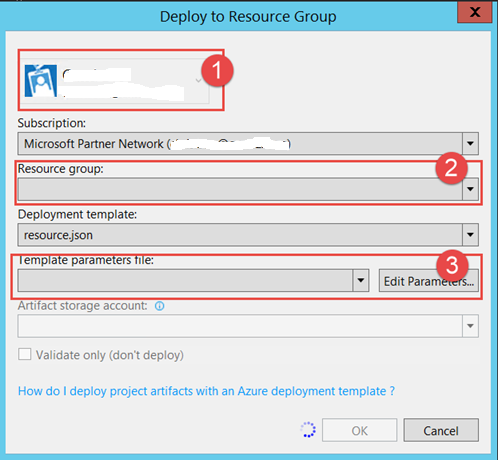
## Create Azure Resources

### Create Azure resources via ARM Template

1. Right click on the Azure Resource Group project “SmartLink.Azure” and click Deploy | New.

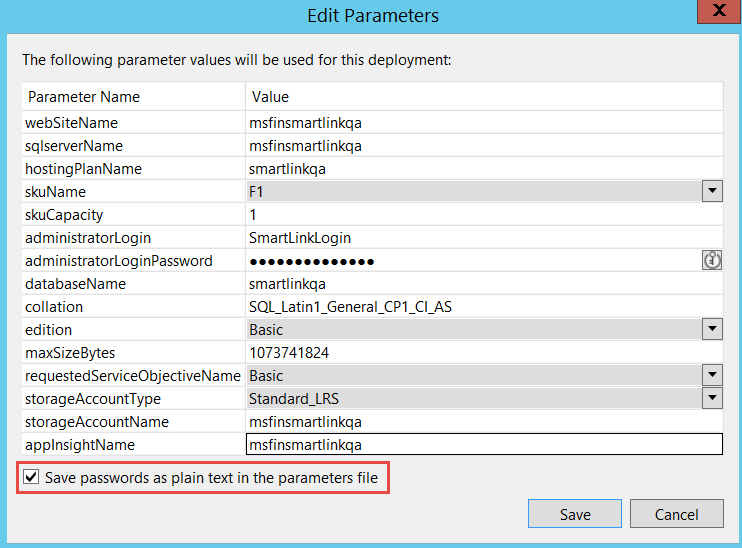


1. Fill the Azure login account and create a new resource group (**For example:** SmartLink.QA) then select the resource parameter json file (**For example:** “resource.qa.parameters.json”) & edit the parameter.



1. After clicking the Edit Parameters and please fill the parameters below

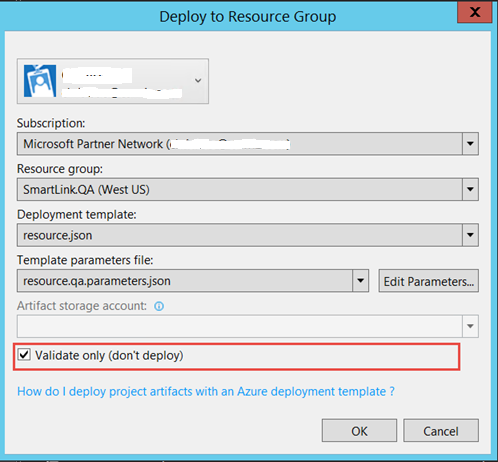
**Note: For the production deployment, the web app/database needs to consider to select higher price plan and please remember to select the ‘alwayson’ in the web app.**



**Note:** Please select the checkbox to store the password as plain text in the parameters file.

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Value** | **Note** |
| webSiteName | <WebSiteName>  **For example:** SmartLinkQAWebApp |  |
| sqlserverName | <SQLServerName>  **For example:** SmarkLinkQASQL | The SQL server name |
| hostingPlanName | <hostingPlanName>  **For example:**  SmartLinkQAHostPlan | The name of the App Service plan to use for hosting the web app. |
| skuName | <skuName>  **For example:**  F1 |  |
| skuCapacity | <skuCapacity>  **For example:**  1 | sets number of workers for this App Service plan SKU |
| administratorLogin | <administratorLogin>  **For example:**  SmartLinkLogin | This login is used to login to the SQL database. |
| administratorLoginPassword | < administratorLoginPassword >  **For example:** |  |
| databaseName | <databaseName>  **For example:**  SmartLinkQA | The database name hosted on the SQL server |
| collation | Leave it as is. |  |
| edition | <Edition>  For example: basic | Specifies the edition for the database. Valid values are:  -- Default  -- None  -- Premium  -- Basic  -- Standard |
| maxSizeBytes | Leave it a is |  |
| requestedServiceObjectivename | <requestedServiceObjectivename> | performance level |
| storageAccountType | <storageAccountType>  **For example:**  Standard\_LRS |  |
| storageAccountName | <storageAccountName>  **For example:**  SmartLinkQA |  |
| appInsightName | <appInsightName>  **For example:**  msfinsmartlinkqa |  |

1. Validate and hit Ok.



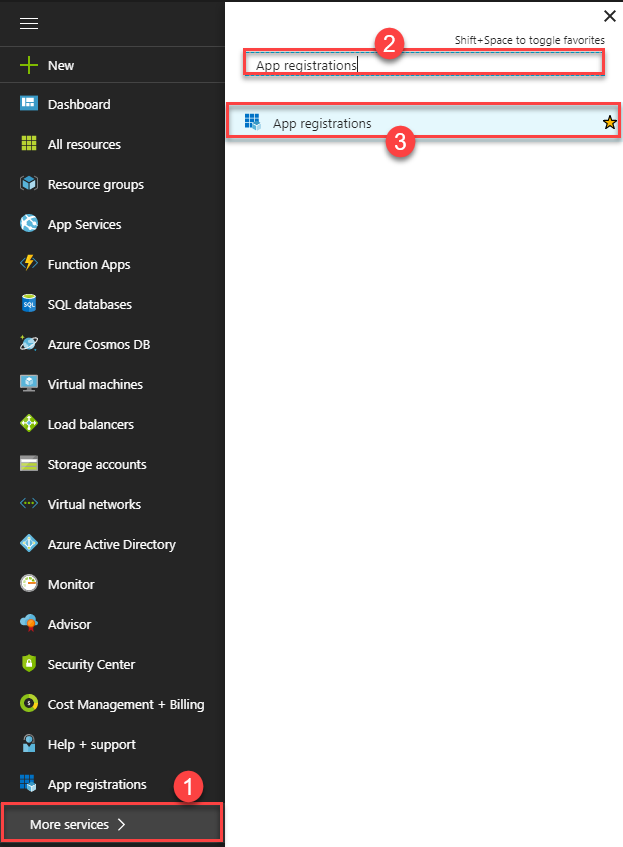
1. Uncheck the checkbox (Validate only) and then hit OK to create the Azure resources.

|  |  |  |  |
| --- | --- | --- | --- |
| **Resource Name** | **Resource Type** | **Pricing Level** | **Resource Group** |
| <WebSiteName> | App Service | F1 | Resource group created in the step 3 above. |
| <SQLServerName> | SQL Server | N/A |
| <databaseName > | SQL database | basic |
| <storageAccountName> | [Storage account](https://azure.microsoft.com/en-us/pricing/details/storage/blobs/) | LRS |
| <WebSiteName> | Application insight | basic |

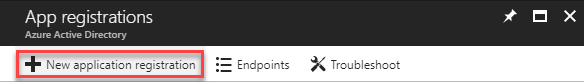
**Note:** The DEV & QA environments are configured with lowest pricing level and UAT/PROD configuration might need to change to have better performance.

## Register the application in AAD for MVC WEB APP.

1. Login the Azure AD using the O365 account. (<https://portal.azure.com> )
2. Click **More services** on the left navigation, search **App registrations** and then select **App registrations.**



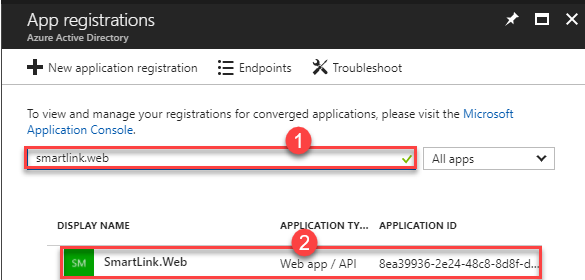
1. Click the **New application registration**



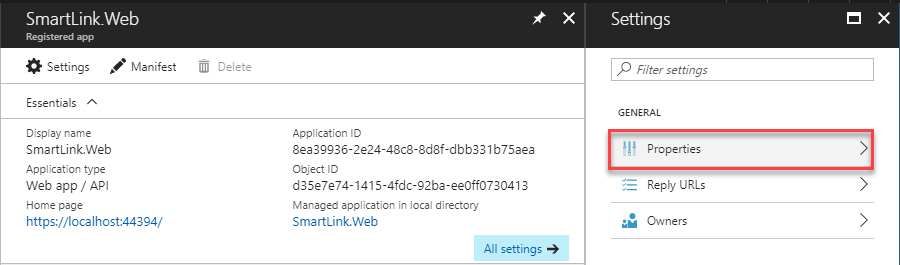
1. In the Add Application wizard, enter a name of **SmartLink.Web** and choose the type **Web Application and/or Web API**. Click the arrow to the next page of the wizard.
2. In the **App Properties** page, enter a **SIGN-ON URL** https://<websitename>.azurewebsites.net

For example: <https://msfinsmartlinkqa.azurewebsites.net>

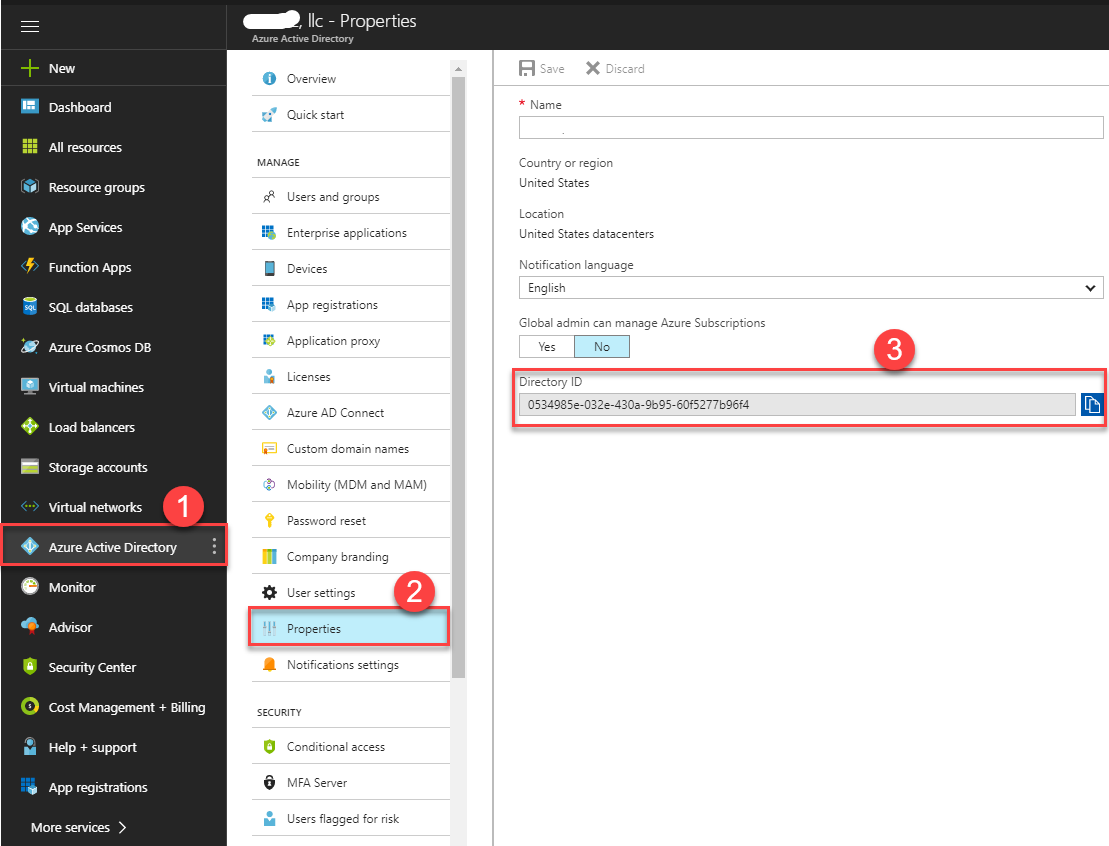
1. Click **Create** to create the registration.
2. Go back the **App registrations** page when the application created successfully.
3. Search the application name and then click it.



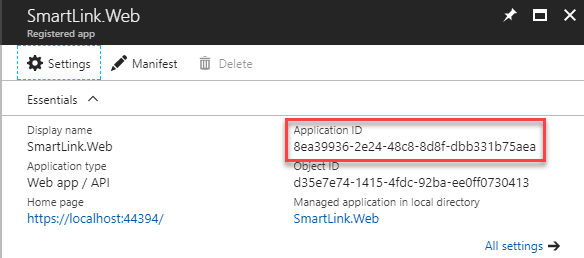
1. Click **Properties**.



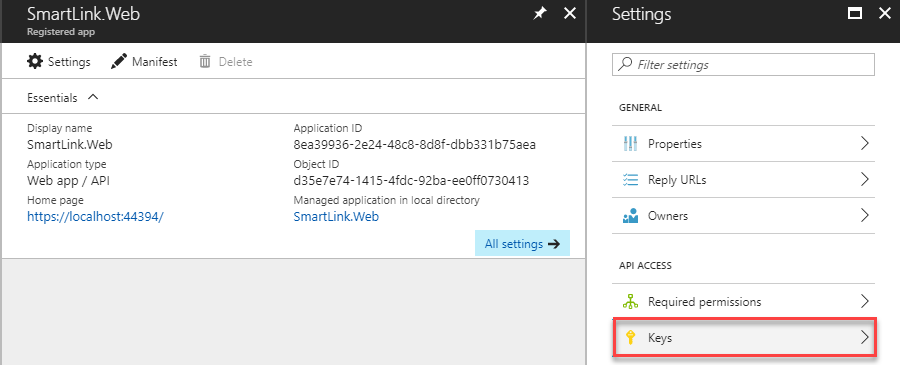
1. Enter an **App ID Uri** https://[your-domain].onmicrosoft.com/<websitename>
2. Hit **Save** to save the properties.
3. Obtain and store the Azure AD tenant ID.
4. Click **Azure Active Directory** in the left menu, and the click **Properties** in the middle menu.
5. The GUID in **Directory ID** is tenant ID and store it.



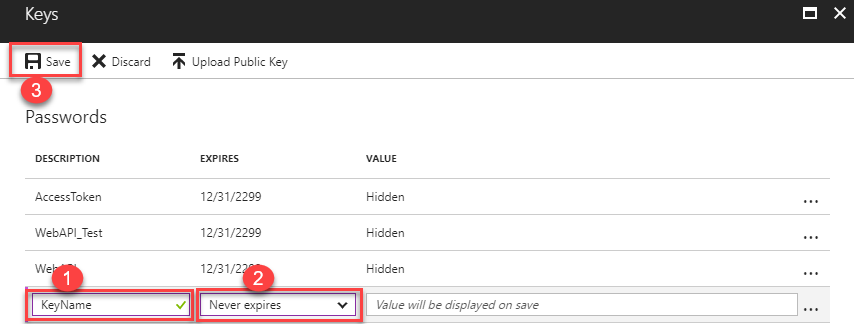
1. Obtain and store the application client ID.
2. On the application, the GUID in **Application ID** is client ID and store it.



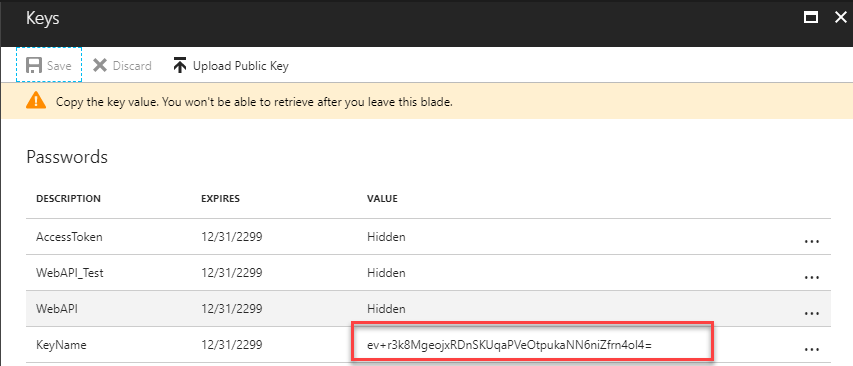
1. Obtain and store the application client secret.
2. On the application, click **Keys**.



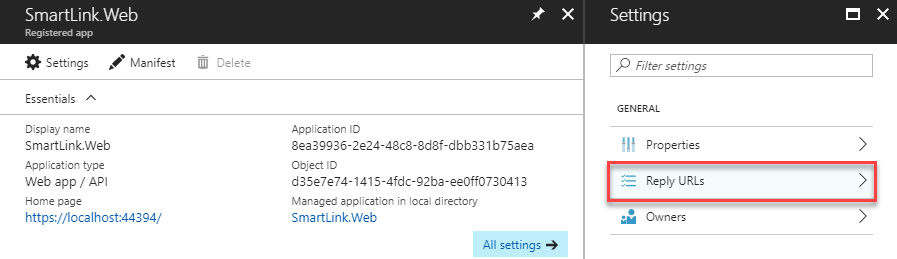
1. Enter **Key Name** and select **Expires** and then click **Save**.



1. After saved successfully copy and story the client secret.



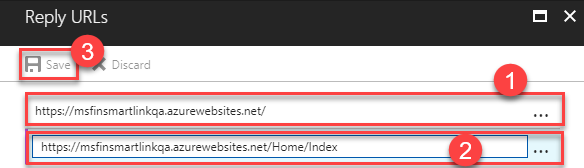
1. Store the copied client secret.
2. Configure the reply URLS.
3. On the application, click **Reply URLs**.



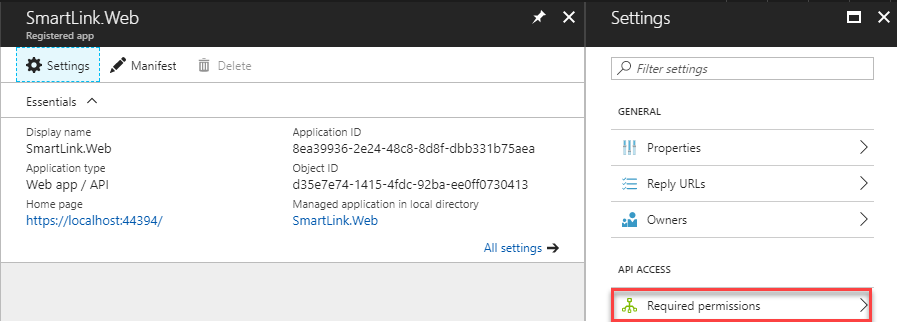
1. Enter the following URLS:
   1. https://<websitename>.azurewebsites.net/auth
   2. https://<websitename>.azurewebsites.net/auth/end
   3. https://<websitename>.azurewebsites.net/Word/Point
   4. https://<websitename>.azurewebsites.net/Excel/Point
   5. https://<websitename>.azurewebsites.net

and then click **Save**.

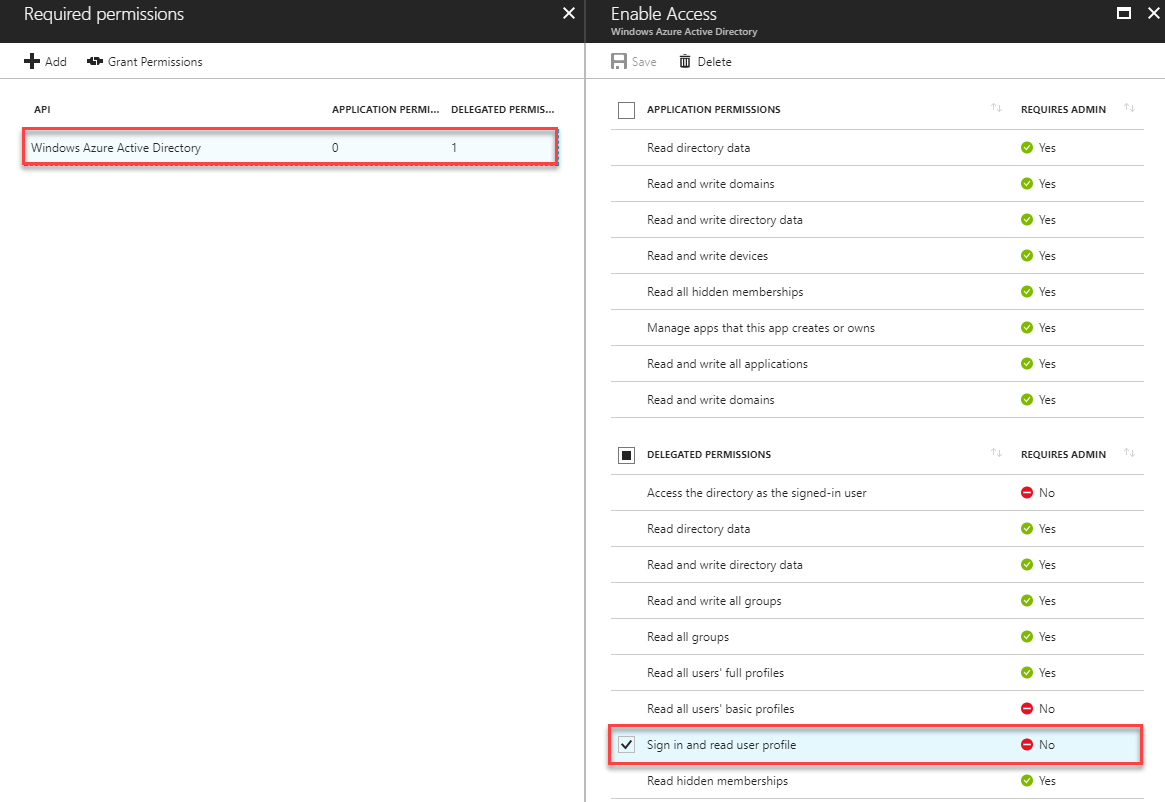
**Note**: You can find the web site name in the “**Create Azure Resources**” section above.



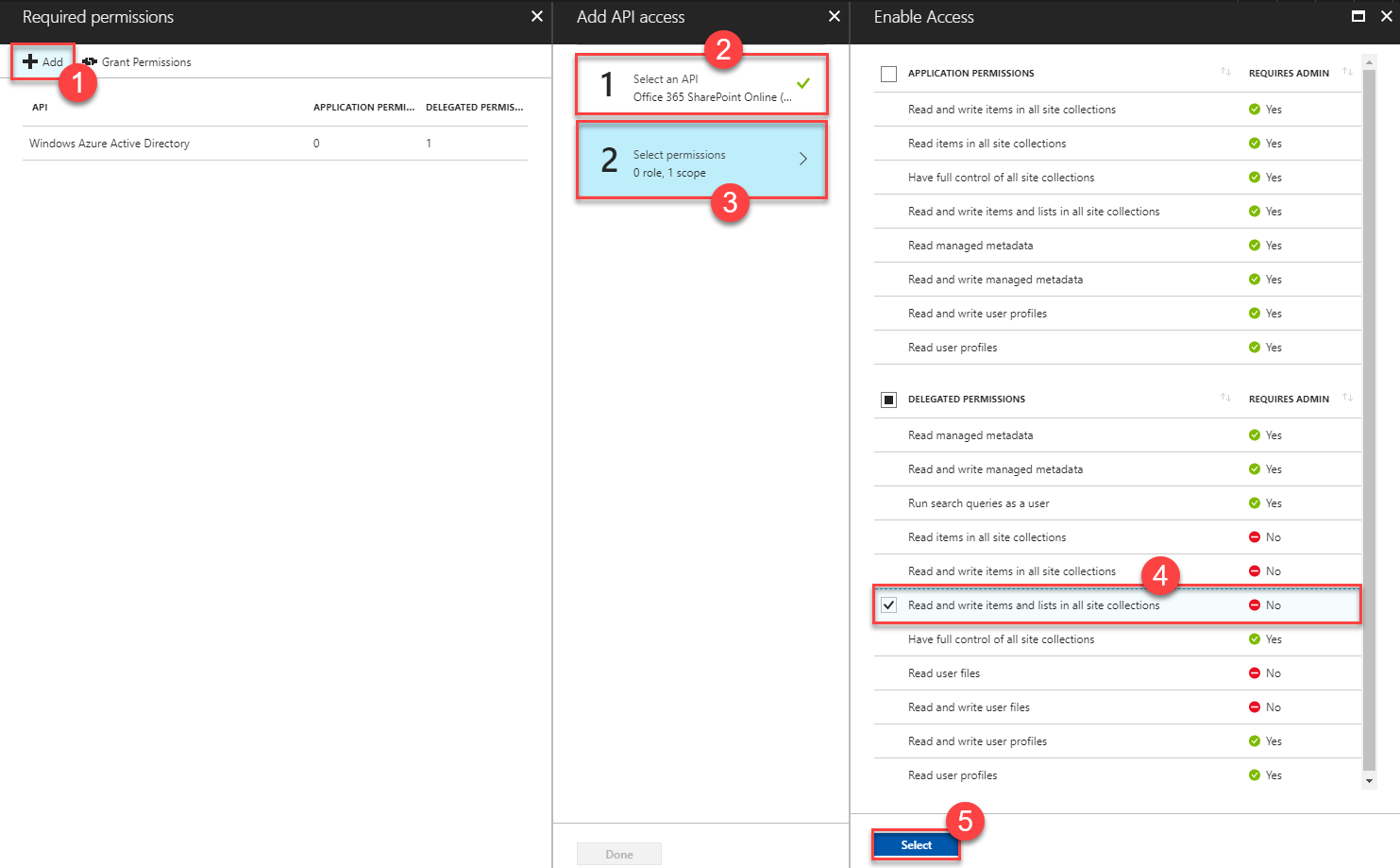
1. Configure the application permissions.
2. On the application, click **Required permissions**.



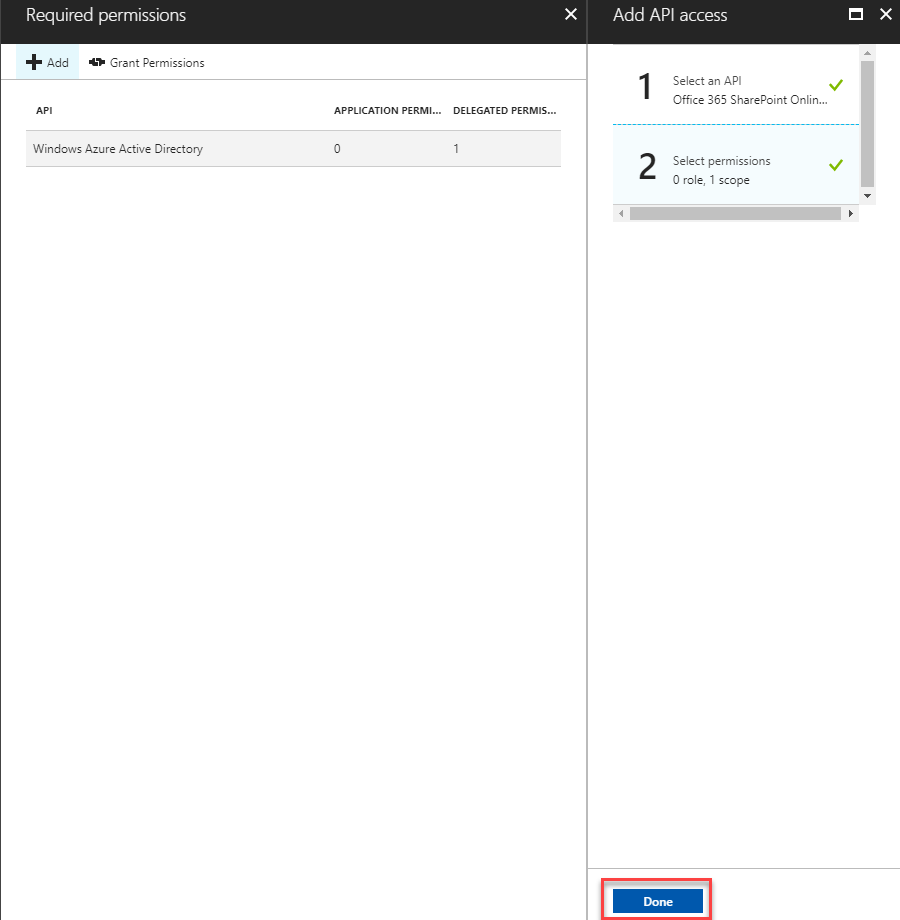
1. Make sure the **Sign in and read user profile** is selected of **Windows Azure Active Directory**.



1. Click **Add**, select **Office 365 SharePoint Online**, select **Read and write items and lists in all site collections** and then click **Select**.



1. Click **Done**.

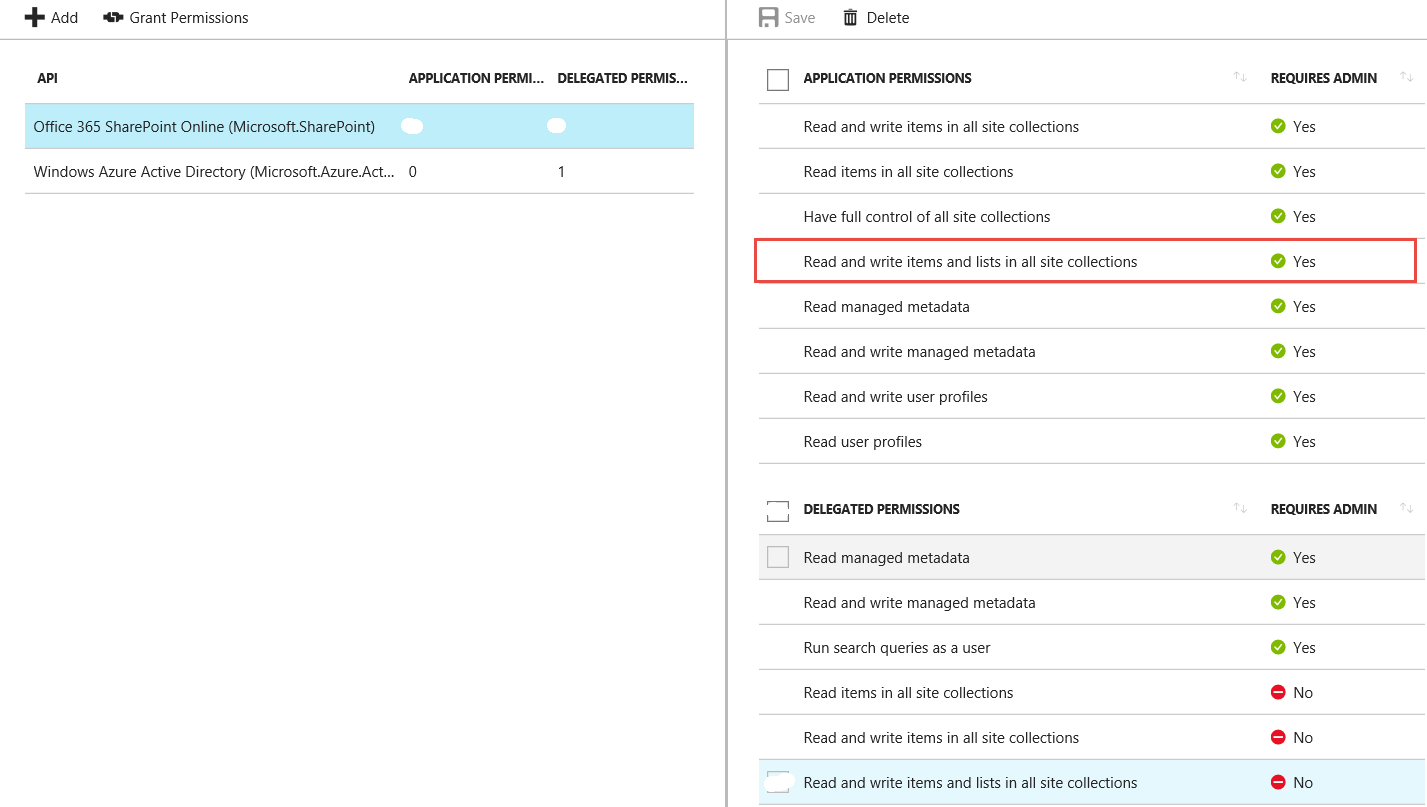


## Register the application in AAD for WEB JOB.

Follow “Register the application in AAD for MVC WEB APP” section to register another app.

|  |  |
| --- | --- |
| SIGN-ON URL | https://<websitename>.azurewebsites.net |
| App ID Uri | https://[your-domain].onmicrosoft.com/<webjobname> |

Here are the app permissions needed for the Azure AD app (please select the highlighted app permission)



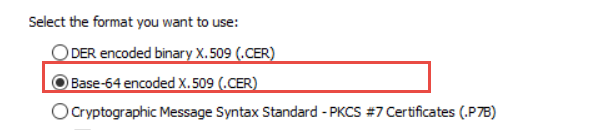
## Configure the communication between WEBJOB and O365 site.

### Create self-certificate

1. Execute the following command to create the self-certificate.

*Makecert -r -pe -n "CN=MyCompanyName MyAppName Cert" -ss my -len 2048*

1. Go to the personal store under current user (my current account).
2. Export the CER certificate with base 64 encoded X.509



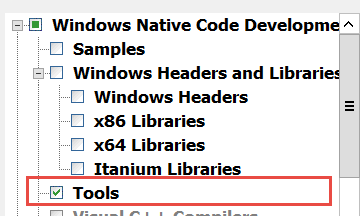
1. Execute the following PowerShell command.

|  |
| --- |
| $cer = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2  $cer.Import("<absolute path of the CER file> Example c:\mycert.cer>")  $bin = $cer.GetRawCertData()  $base64Value = [System.Convert]::ToBase64String($bin)  $bin = $cer.GetCertHash()  $base64Thumbprint = [System.Convert]::ToBase64String($bin)  $keyid = [System.Guid]::NewGuid().ToString() |

1. Store the base64Value; $base64Thumbprint; keyid that will be used in other places.

**Note:** **If MAKECERT is not available, please download the windows SDK and install the tool only in the screenshot below.**

<https://www.microsoft.com/en-us/download/confirmation.aspx?id=8279>



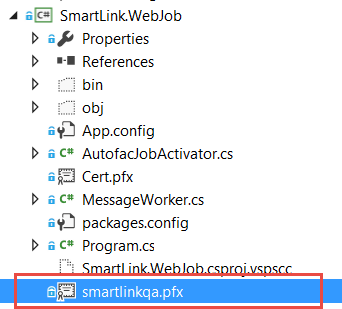
### Export PFX certificate.

1. Export the PFX certificate from the self-certificate (remember to export with the private key)
2. Store the password when export the PFX certificate.
3. Store the PFX certificate.

### Update the team project to include this certificate

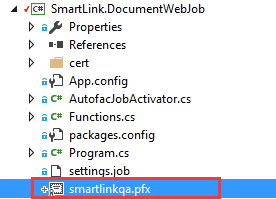
1. Open the solution.
2. Include the PFX certificate in SmartLink.WebJob project.

**For example:** smartlinkqa.pfx.



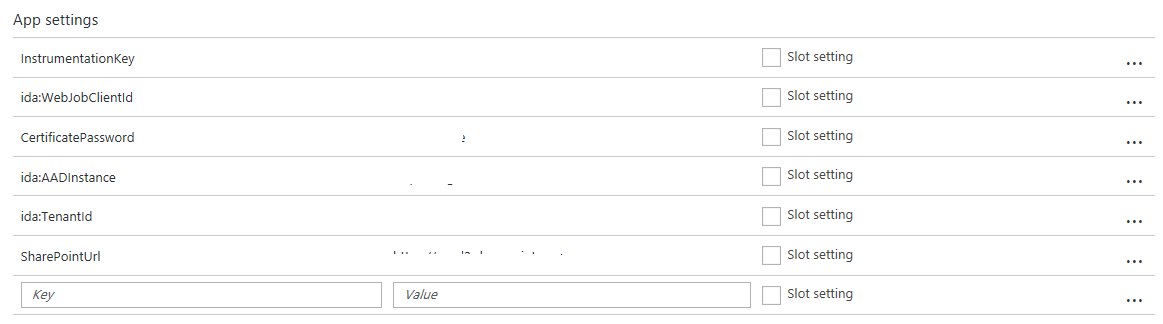
1. Include the PFX certificate in SmartLink.DocumentWebJob project.

**For example:** smartlinkqa.pfx.



### Configure the app settings in Azure portal.

1. Go to the smart link QA website create in this section.
2. Go to the application settings.
3. Locate the App settings node



1. Fill the following app settings.

|  |  |  |
| --- | --- | --- |
| **App setting** | **Value** | **Notes** |
| InstrumentationKey | Application Insight Instrumentation key. | Find it in the application insights |
| ida:WebJobClientId | The application ID of the AAD App for WEB JOB | Find it in the AAD App for web job. |
| CertificatePassword | The password when export the certificate | Step b in this section. |
| ida:AADInstance | <https://login.microsoftonline.com/> | Fixed value |
| ida:TenantId | The Azure Tenant ID | Step 10 in this section |
| SharePointUrl | The root site collection of the O365 site | For example:  https://<tenant>.sharepoint.com |
| Key | 23, 7, 19, 11, 24, 226, 85, 45, 114, 184, 27, 162, 37, 112, 222, 209, 241, 24, 175, 144, 176, 53, 196, 29, 24, 26, 17, 218, 131, 236, 53, 209 | Fixed value |
| ida:ClientId | The application ID of the ADD App for MVC WEB APP | Find it in the AAD APP for MVC WEB APP |
| ida:ClientSecret | The secret key of the ADD App for MVC WEB APP | Find it in the AAD APP for MVC WEB APP |

### Update the manifest file.

1. Download the manifest file from the AAD app for the web job.
2. Edit XXX following the template below

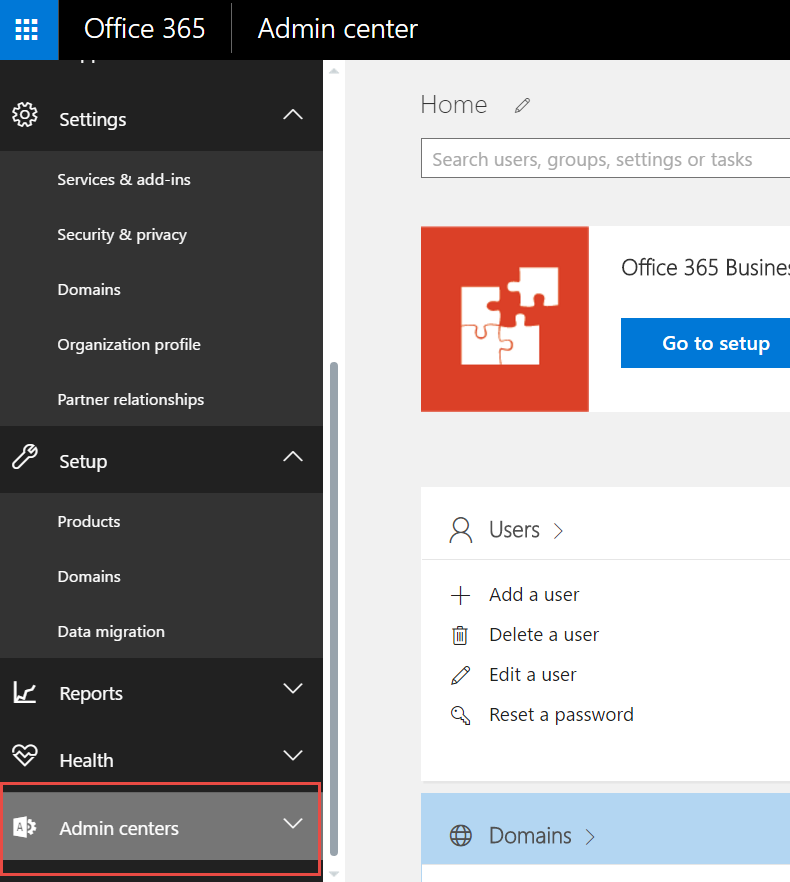
|  |
| --- |
| "keyCredentials": [  {  "customKeyIdentifier": "$base64Thumbprint\_from\_above",  "keyId": "$keyid\_from\_above",  "type": "AsymmetricX509Cert",  "usage": "Verify",  "value": "$base64Value\_from\_above"  }  ], |

|  |  |  |
| --- | --- | --- |
| **Key** | **Value** | **Notes** |
| customKeyIdentifier | $base64Thumbprint | Step e in this section. |
| keyId | $keyid | Step e in this section. |
| value | $base64Value | Step e in this section. |

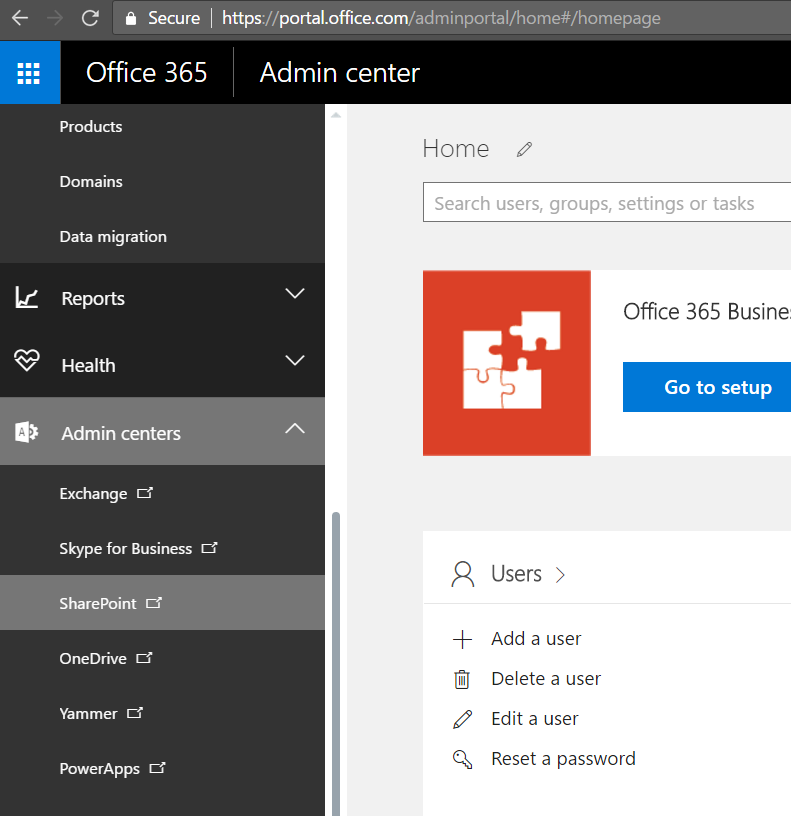
1. Upload the manifest file.

## Configure SharePoint Catalog

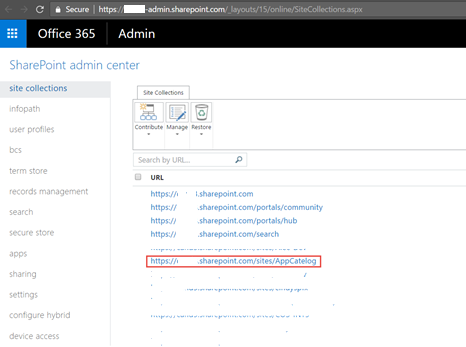
1. Open <https://portal.office.com> , sign in with your account.
2. Expand Admin centers on the left navigation.



1. Click on SharePoint on the left navigation



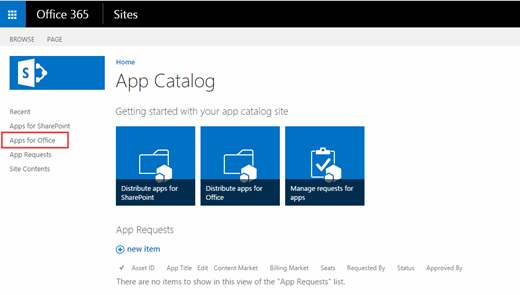
1. A new page will be displayed, find the catalog URL and copy it.



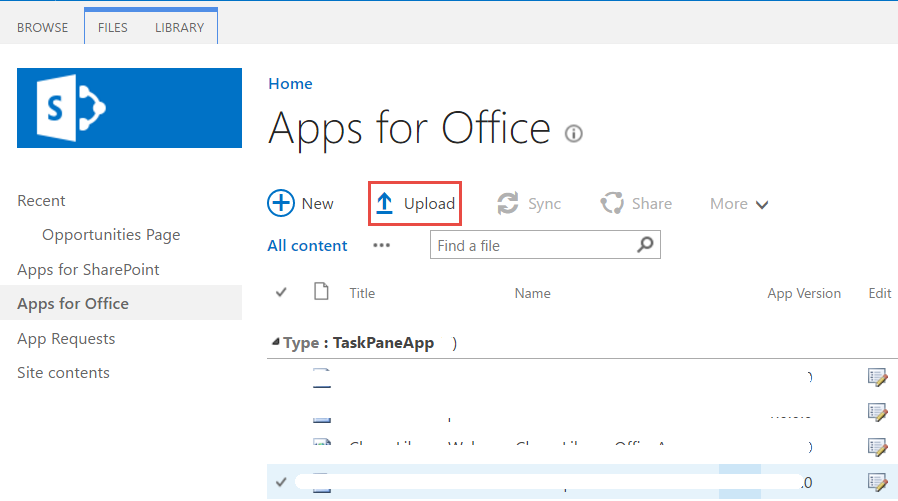
1. Go to the catalog page



1. Click on Apps for Office



1. Click on Upload



1. Find the app manifest file in:

***src/SmartLink/SmartLinkExcel/SmartLinkExcelManifest/SmartLinkExcel.xml***

1. Update the manifest file.
   1. Update the Id with a new generated GUID and you could refer to the link below on how to generate it. <https://guidgenerator.com/online-guid-generator.aspx>
   2. Update the value of display name and description to SmartLink.
   3. Update the IconUrl default value to the image path.
   4. Update the hosts to the following elements.

<Host Name="Workbook" />

<Host Name="Document" />

<Host Name="Presentation" />

* 1. Update the source location default value to the one provisioned in Azure.



1. Upload word app manifest file to SharePoint catalog.

## Configure SharePoint Document ID Service

**NOTE**:  In order enable the Document ID feature you must be a site collection administrator.

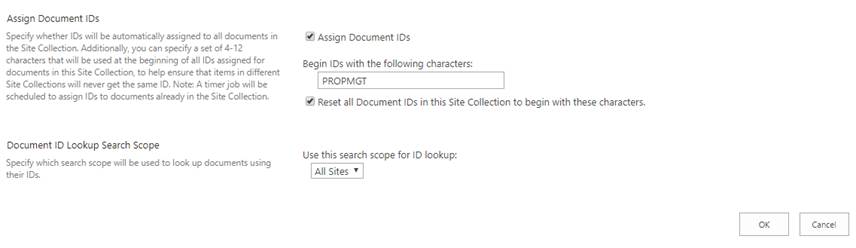
1. Go to the created Opportunity SharePoint site.

**For Example:** https://<tenant>.sharepoint.com/sites/<OpportunityName>.  
If you created an Opportunity with a name including a space, for example “My Opportunity”, then you’ll need to remove the space. It should look like “MyOpportunity”.

1. Click **Site Contents** and then click **Site Settings**.
2. Under Site Collection Administration, click **Site collection features**.
3. Next to Document ID Service click **Activate**.

cid:image002.jpg@01D410BC.49A4CC20

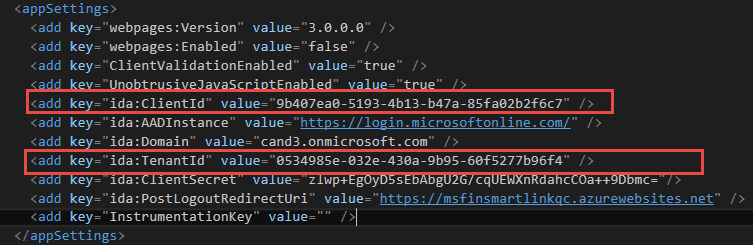
1. Go back to site settings page.
2. Under Site Collection Administration, click **Document ID settings**.
3. On the Document ID settings page, check the **Assign Document IDs** check box. Input the begin characters (Such as PROPMGT) in **Begin IDs with the following characters** text box, check **Reset all Document IDs in this Site Collection to begin with these characters** check box and select All Sites in **Use this search scope for ID lookup** dropdown.



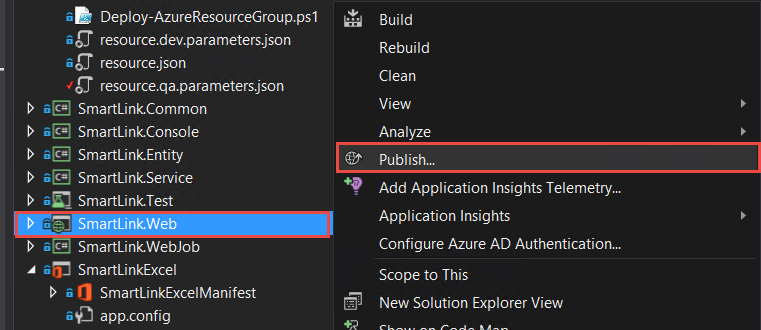
1. Click **OK.**

## Publish website

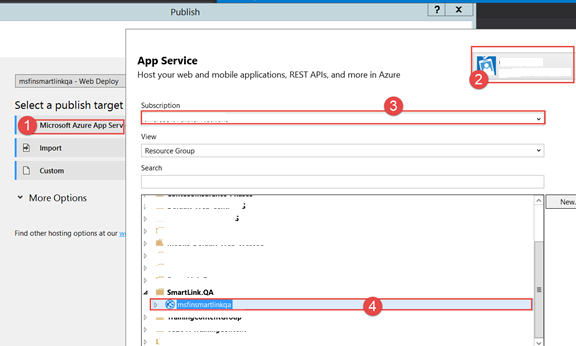
1. Open Visual Studio with administrator permission
2. Open the SmartLink project
3. Update the ida:clientID, ida:clientSecret and ida:TenantID in the web.config with the values generated in step 10, step 11 & step 12 in “Register the application in AAD” section.



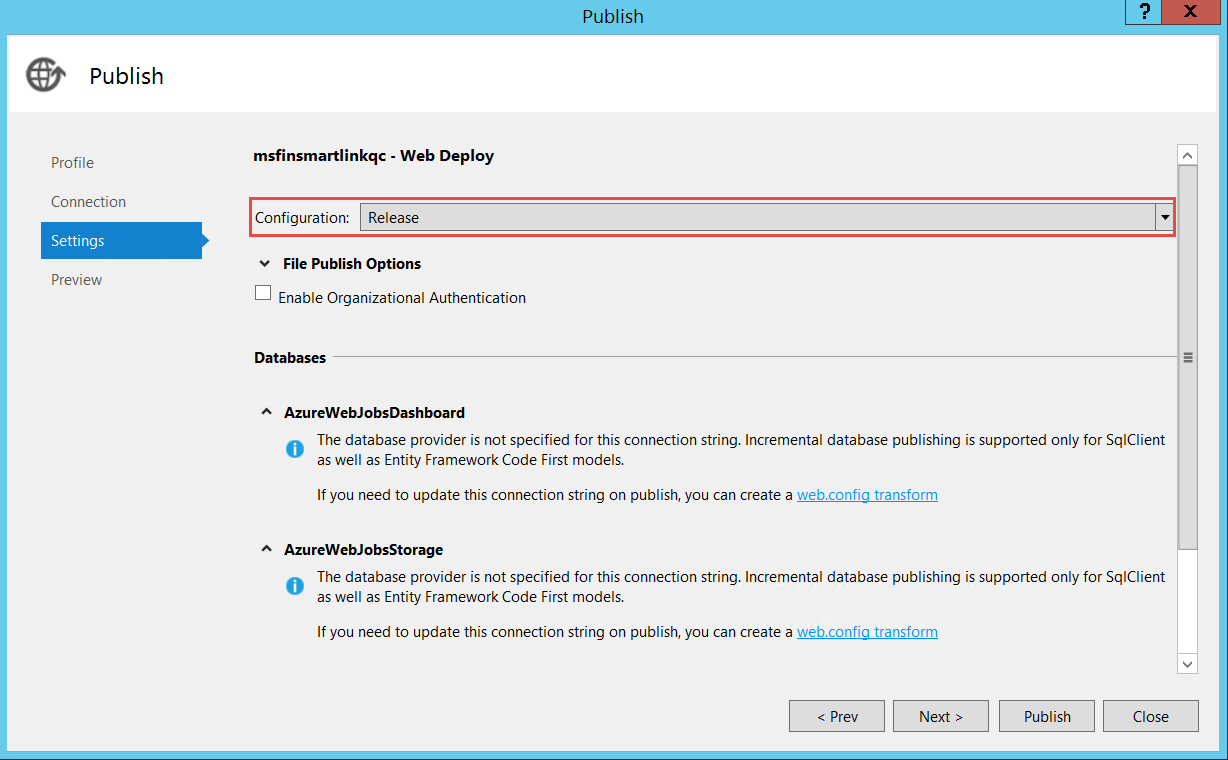
1. Right click on SmartLink.Web and then select Publish



1. You will see the Publish Web popup
2. Click on Microsoft Azure App Service.

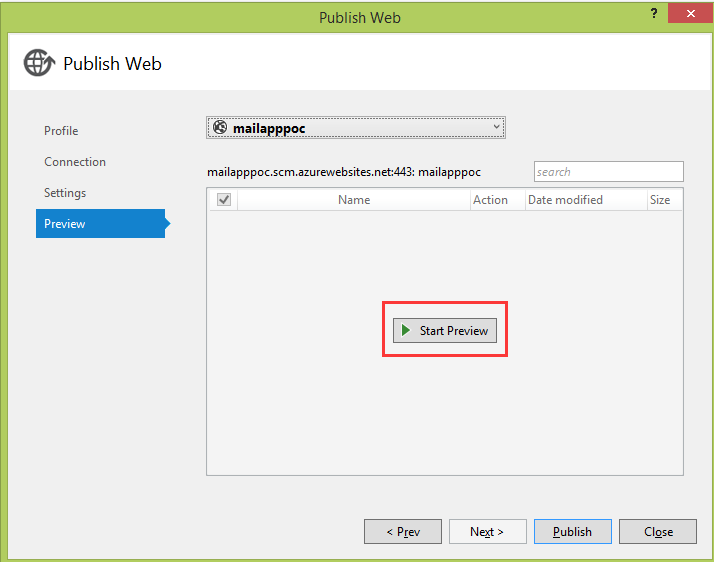


1. Sign in with Azure account and select your azure website, then click Next
2. Select Web Deploy and then click Next
3. Select Release and click Next



1. Click on Start Preview

**Note:** an authentication window may appear, and you need to fill the O365 account (user name & password)



1. Click on Publish when there is no error in preview window

## Encrypt the Azure connection string in application settings

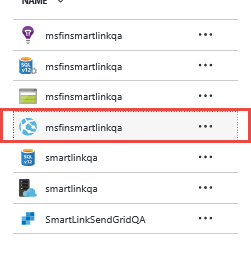
### Login the Azure portal.

a. Login the site <https://portal.azure.com>

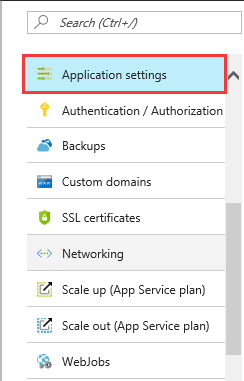
### Go to the web site.

Select website created in this section.

**For example:** QA environment web site is **msfinsmartlinkqa**

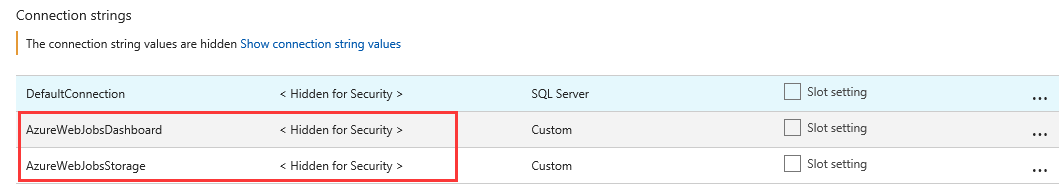


### Go to the application settings



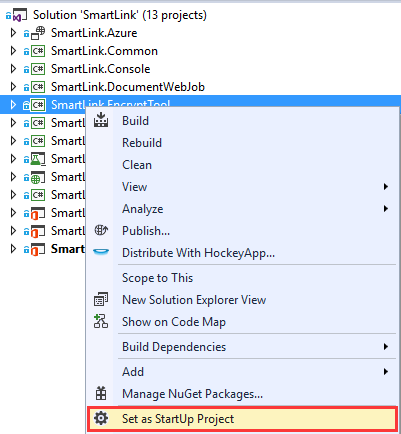
### Copy the connection strings

Copy the value of AzureWebJobsDashboard and AzureWebJobsStorage under Connection strings



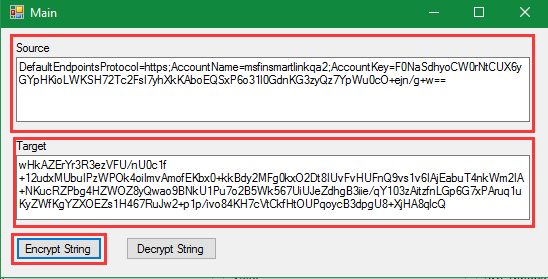
### Open SmartLink project

* 1. Switch to Visual Studio and make sure you are opened the SmarkLink Project.
  2. Open Encrypt tool
  3. Set SmartLink.EncryptTool as StartUp Poject and Run it

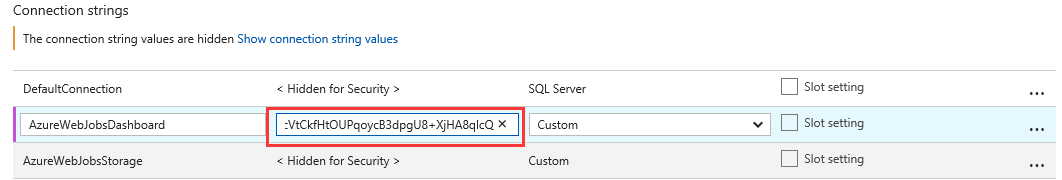


### Encrypt connection strings

* 1. Copy the AzureWebJobsDashboard value into Source and click on Encrypt String



* 1. Copy the value from Target and paste it to AzureWebJobsDashboard



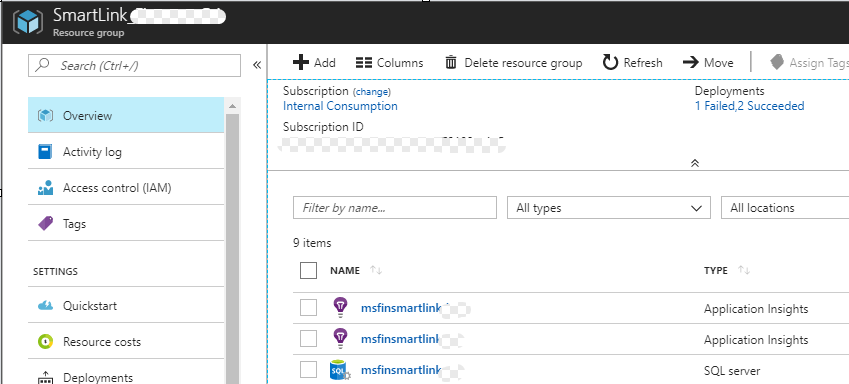
* 1. Repeat step a and b to encrypt the AzureWebJobsStorage

## Configure always encrypted feature for database

**IMPORTANT: The application you registered and the azure resource group you created must be in same domain.**

### Create Key Vault.

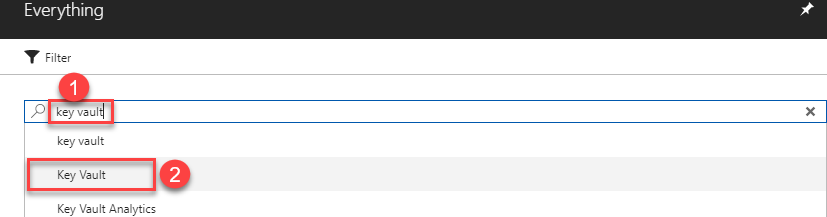
* 1. Login to the [Azure portal](https://portal.azure.com).
  2. Go to the resource group page you created in “Create Azure Resources” section.



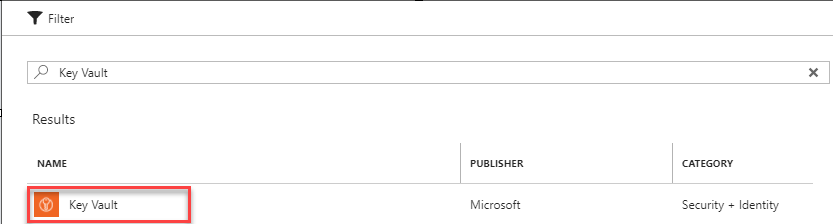
* 1. Click Add on the top menu.



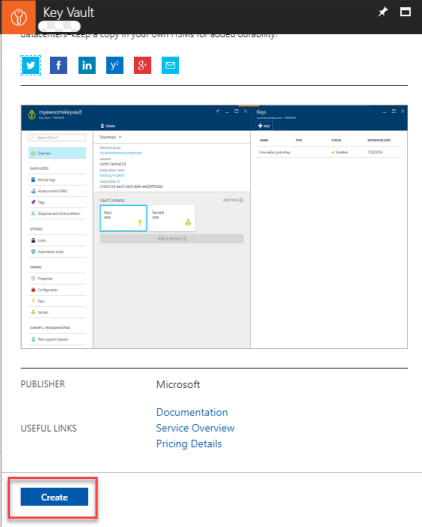
* 1. In the new page, type “key vault” and then select Key Vault.



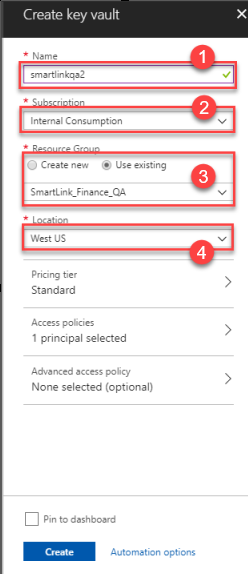
* 1. Select Key Vault.



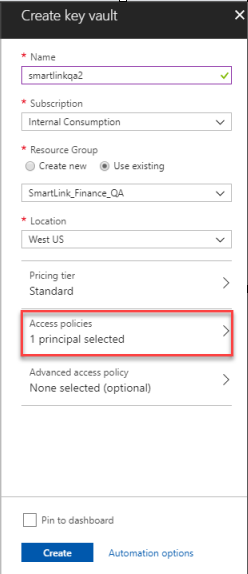
* 1. Click on Create.



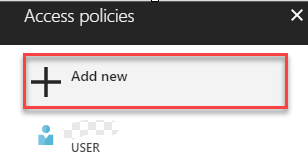
* 1. Fill the form.
     1. Type the key vault name.
     2. Select subscription.
     3. Select the existing group.
     4. Select location.



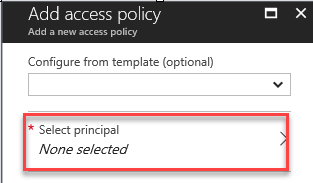
* 1. Click on Access policies.



* 1. Click on Add new.

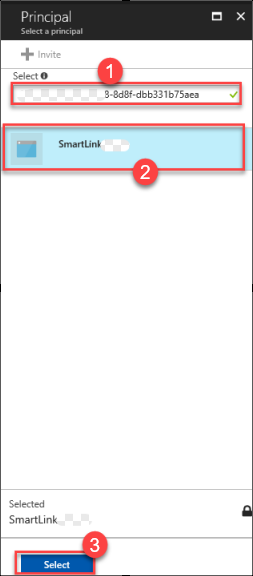


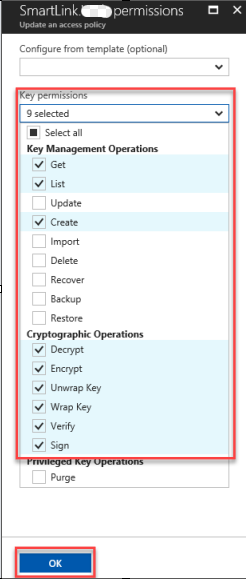
* 1. Click Select principal.



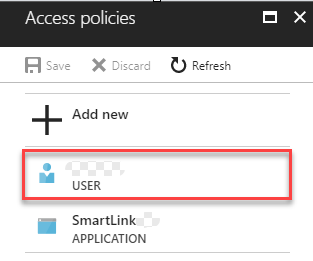
* 1. Type the client ID then select the application, and then click Select.

**NOTE: You can find the client ID in the previous section.**

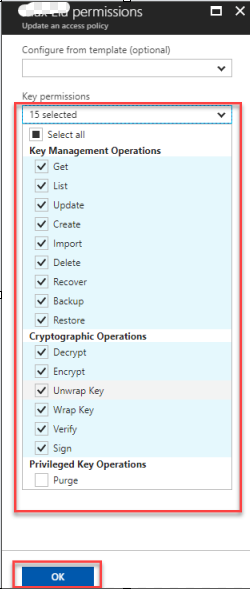
* 1. 
  2. Select the following key permissions and then click OK.



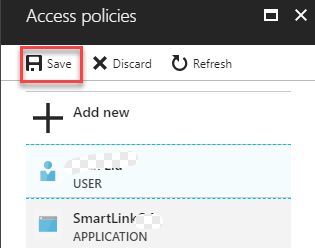
* 1. Click OK.
  2. Go back to Access policies page and click the user name you signed in.



* 1. Select the following key permissions and then click OK.



* 1. Click Save in Access policies page.



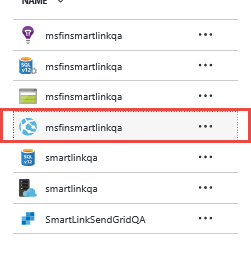
## Encrypt the database connection string in application settings

### Login to the [Azure portal](https://portal.azure.com/).

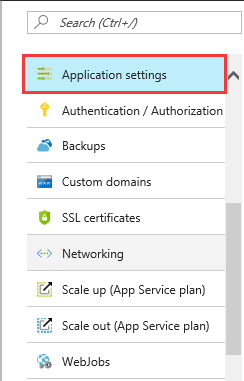
### Go the web site.

Select website created in this section.

**For example:** QA environment web site is **msfinsmartlinkqa**



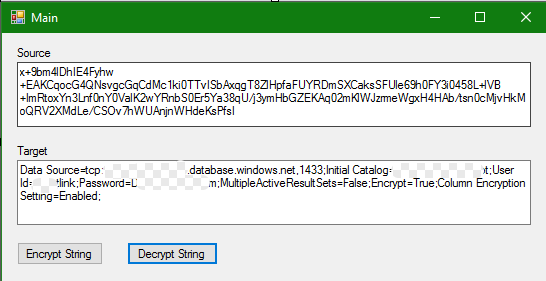
### Go the application settings.



### Copy the value of DefaultConnection. If it is encrypted, please decrypt it first.

**NOTE: If you don’t know how to decrypt it please following the steps below.**

1. Open SmartLink in Visual Studio.
2. Set SmartLink.EncryptTool as StartUp Poject and Run it.
3. Copy the encrypted string into Source and then click Decrypt String. The value in Target is the unencrypted string.



### Please make sure the unencrypted connection string looks like the format below.

**Data Source={server address};Initial Catalog={database name};User Id={user name};Password={password};MultipleActiveResultSets=False;Encrypt=True;Column Encryption Setting=Enabled;**

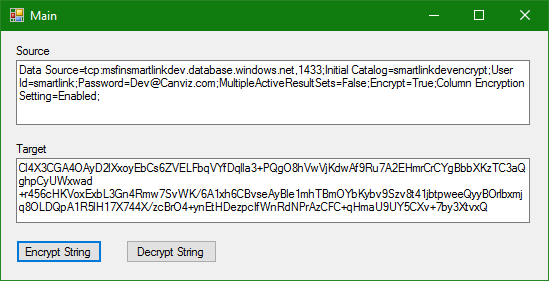
**For Example**: The QA environment connection string is:

Data Source=tcp:msfinsmartlinkdev.database.windows.net,1433;Initial Catalog=smartlinkdevencrypt;User Id=smartlink;Password=<Password>;MultipleActiveResultSets=False;Encrypt=True;Column Encryption Setting=Enabled;

### Encrypt the connection string.

**NOTE: If you don’t know how to encrypt it please following the steps below.**

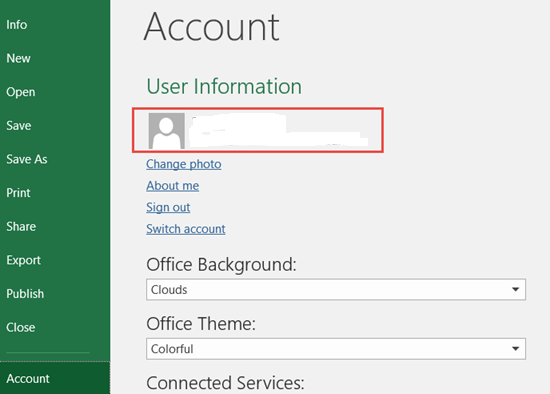
1. Open SmartLink in Visual Studio.
2. Set SmartLink.EncryptTool as StartUp Poject and Run it.
3. Copy the unencrypted string into Source and then click Encrypt String. The value in Target is the encrypted string.



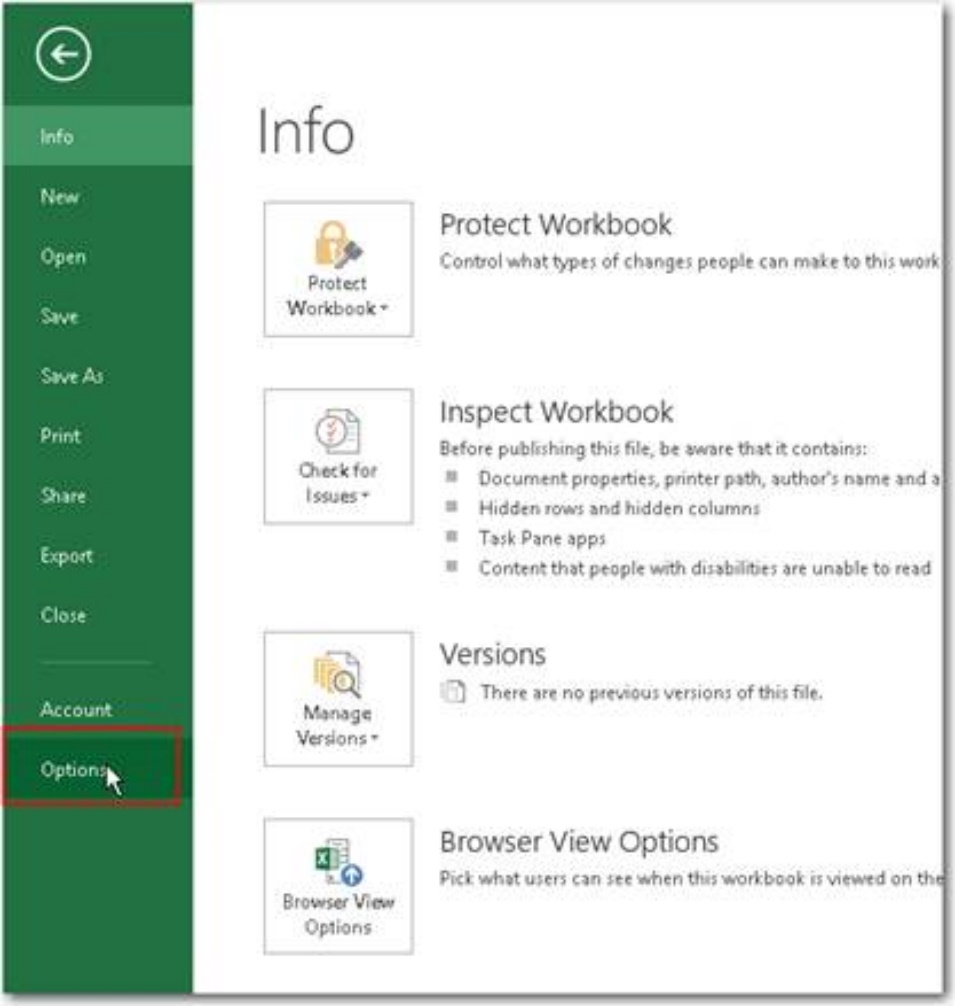
### Copy the encrypted connection string and paste it to DefaultConnection in Azure and then click Save.

## Install Excel Add-in

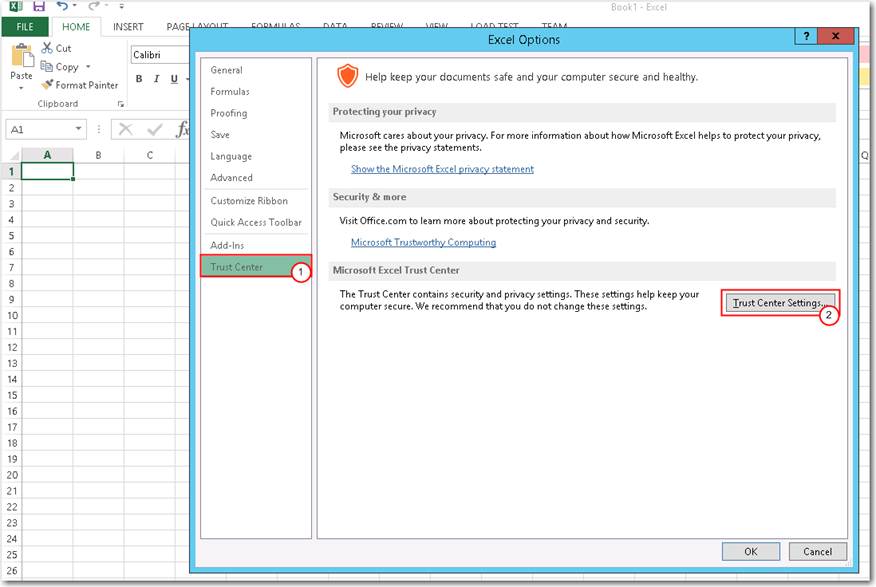
1. Open Excel 2016
2. Sign in with the your O365 account

****

1. Click File | Options

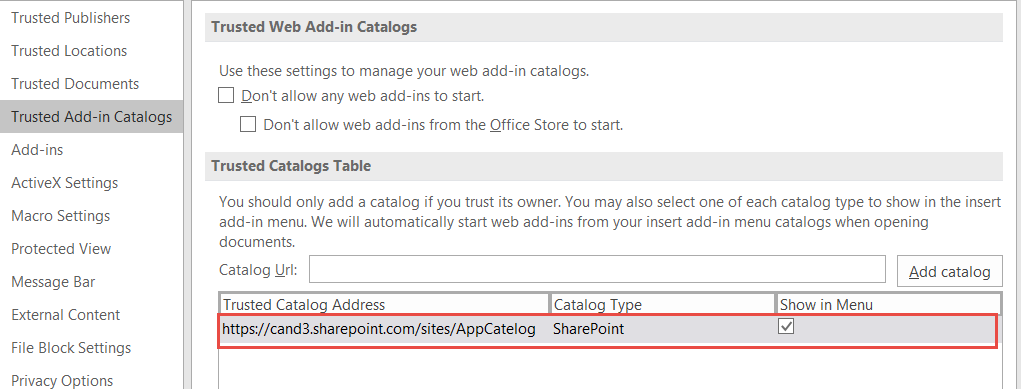


1. Click Trust Center | Trust center settings

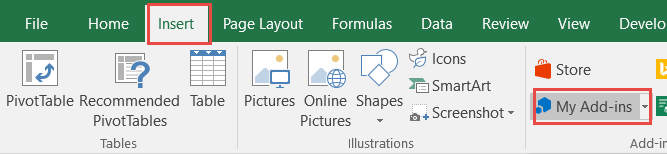


1. Click Trust App Catalogs | Add catalog with the link you found in above

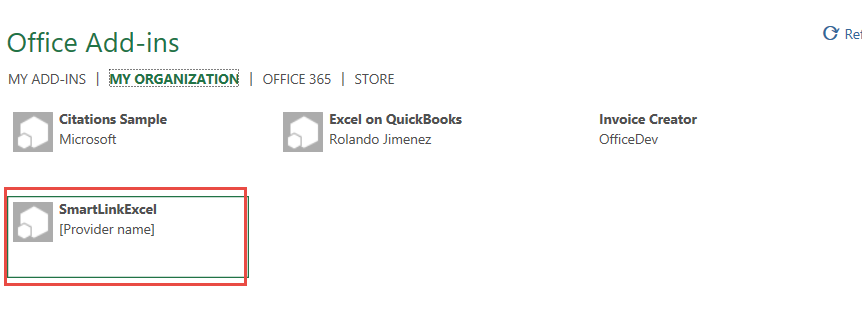
For example: https://<tenant>.sharepoint.com/sites/AppCatelog



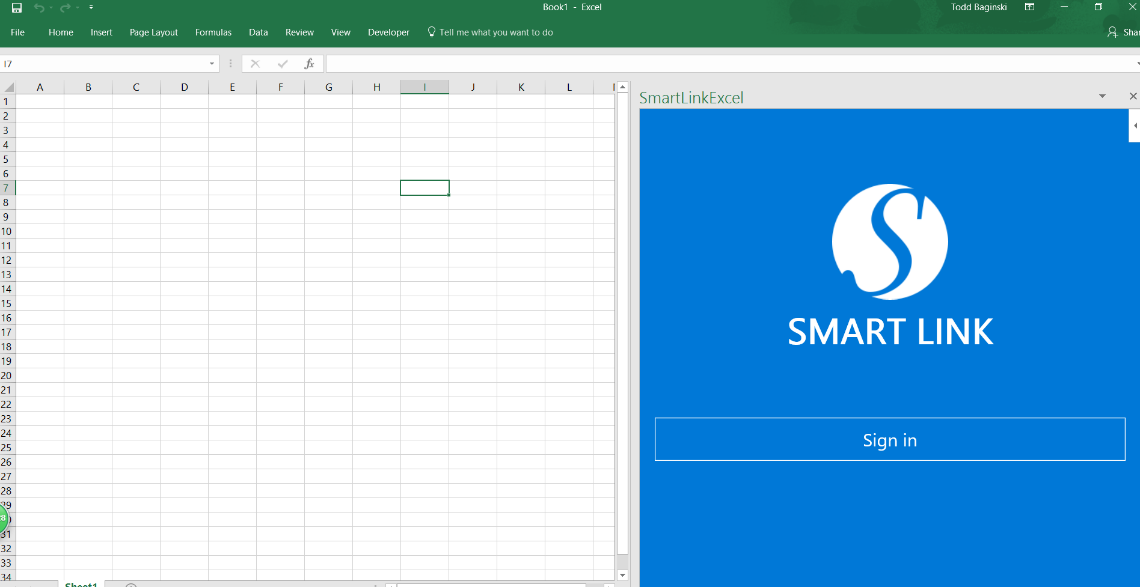
1. Restart the excel.
2. Click Insert and My Add-ins.



1. Click ‘MY ORGANIZATION’ and insert the SmartLinkExcel.

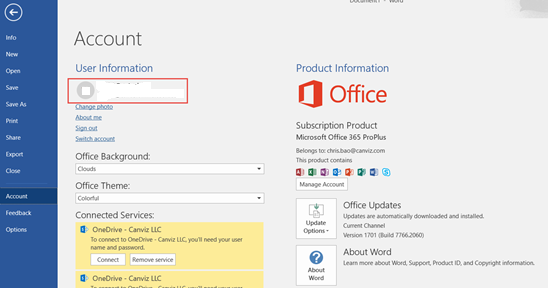


1. Then the Excel Add-in would be shown on the right.

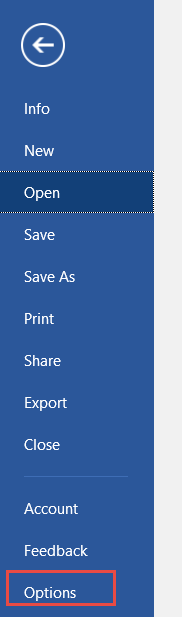


## Install Word Add-in

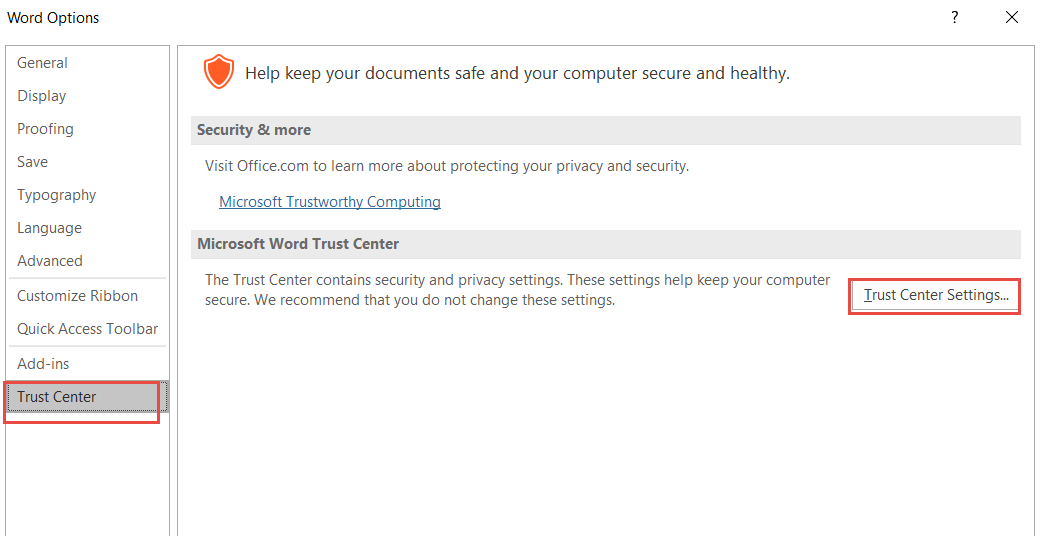
1. Open Excel 2016
2. Sign in with the your O365 account



1. Click file | Options.



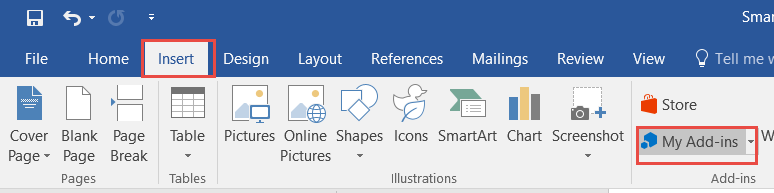
1. Click Trust Center | Trust center settings



1. Click Trust App Catalogs | Add catalog with the link you found in above

For example: https://<tenant>.sharepoint.com/sites/AppCatelog

1. Restart the Word.
2. Click Insert and My Add-ins.



1. Click ‘MY ORGANIZATION’ and insert the SmartLinkWord.
2. Then the Word Add-in would be shown on the right.