

Dynamics CRM

IMS APP Deployment Document

|  |  |
| --- | --- |
| Author(s) | AppSource Team |
| Date | 16/07/2018 |
| Version | 1.0 |

# Contents

[Contents 2](#_Toc522726283)

[1.Summary 3](#_Toc522726284)

[1.1 Overview 3](#_Toc522726285)

[1.2 Document Overview 3](#_Toc522726286)

[2. Prerequisites 3](#_Toc522726287)

[2.1 Power Auto Number 4](#_Toc522726288)

[2.2 Data Export Service 5](#_Toc522726289)

[2.3 Email Lab Alerts 7](#_Toc522726290)

[2.4 Voice of Customer (VOC) 9](#_Toc522726291)

[2.5 Portal Installation and Configuration 11](#_Toc522726292)

[2.6 System Settings 17](#_Toc522726293)

[2.7 Group Policy Set up 19](#_Toc522726294)

[2.8 Windows PowerShell 21](#_Toc522726295)

[2.9 Azure Setup 23](#_Toc522726296)

[3. Microsoft Dynamics CRM Package Deployer 25](#_Toc522726297)

[3.1 Download Package Deployer 26](#_Toc522726298)

[4.Download the IMS App Package 28](#_Toc522726299)

[5.TroubleShooting Details 42](#_Toc522726300)

[5.1 Azure Applications 42](#_Toc522726301)

[5.1.1 Web Service Plan 42](#_Toc522726302)

[5.1.2 Web Jobs 42](#_Toc522726303)

[5.1.3 Web Applications 47](#_Toc522726304)

[5.2 FN Portal 54](#_Toc522726305)

[5.2 Questionnaire Portal 54](#_Toc522726306)

[6.Post Deployment Configuration 55](#_Toc522726307)

[6.1 Azure Configurations 55](#_Toc522726308)

[6.2 CRM Configurations 55](#_Toc522726309)

[7.Uninstalling Immigration AppSource System 56](#_Toc522726310)

# 1.Summary

## Overview

Microsoft AppSource is our new destination to enable business users to explore and use line-of-business SaaS apps from Microsoft and our partners. Users can also discover add-ins for Microsoft business apps, like Dynamics 365, Power BI, Office and more. Through AppSource, ISVs (app publishers) and other partners can reach a broad range of customers right where the users are either through the AppSource portal or within the context of the Microsoft services they’re already using.

## 1.2 Document Overview

This document describes the components of Immigration Management System ( referred as “IMS” in the document) App and its related information including the prerequisites to be made available in the system before proceeding with the IMS App Solution installation and steps to download and import the solution.

This is a living document that will be used to support the CRM application and its associated portal.

# 2. Prerequisites

As a part of prerequisites to install the IMS App, the below solutions must be installed or imported into the user’s CRM organization. The user must login with Office 365 Administrator account to install the below apps.

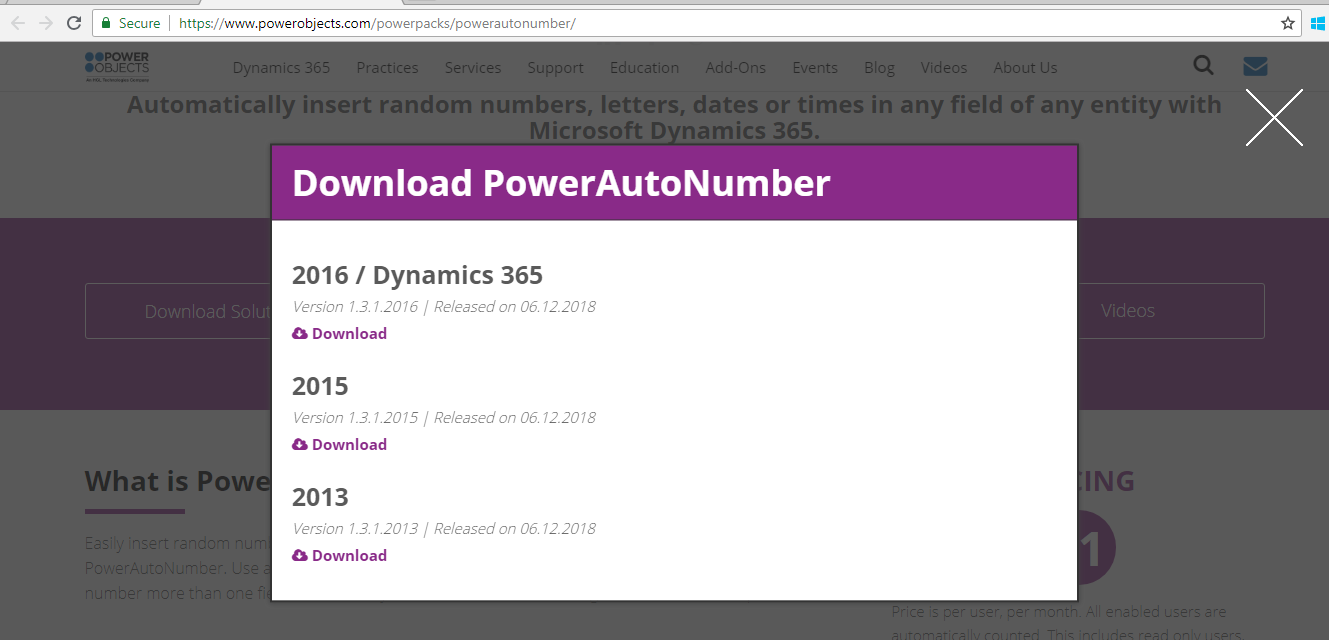
* Power Auto Number
* Data Export service
* Email Lab Alerts
* Voice of Customer
* Portal Installation and Configuration

## 2.1 Power Auto Number

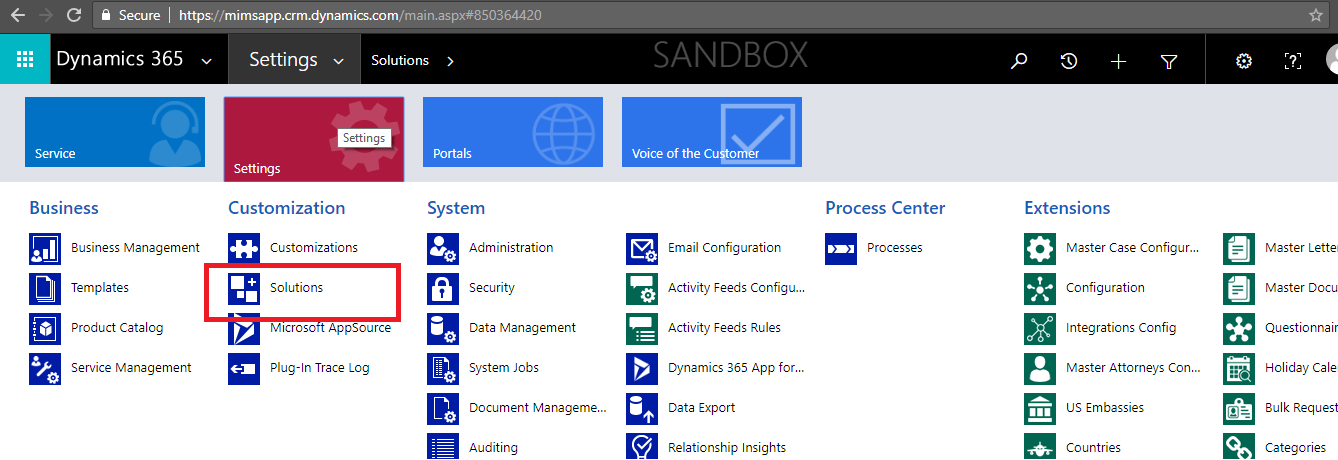
Power Auto Number is an app that enables automatic insertion of random numbers, letters, dates or times in any field of any entity in CRM. The app enables the use of combination of numbers, letters and dates in the CRM fields and provide the ability to number more than one field in each entity. Random numbers are guaranteed to be unique.

The power Auto number solution can be downloaded from the below **web site**.

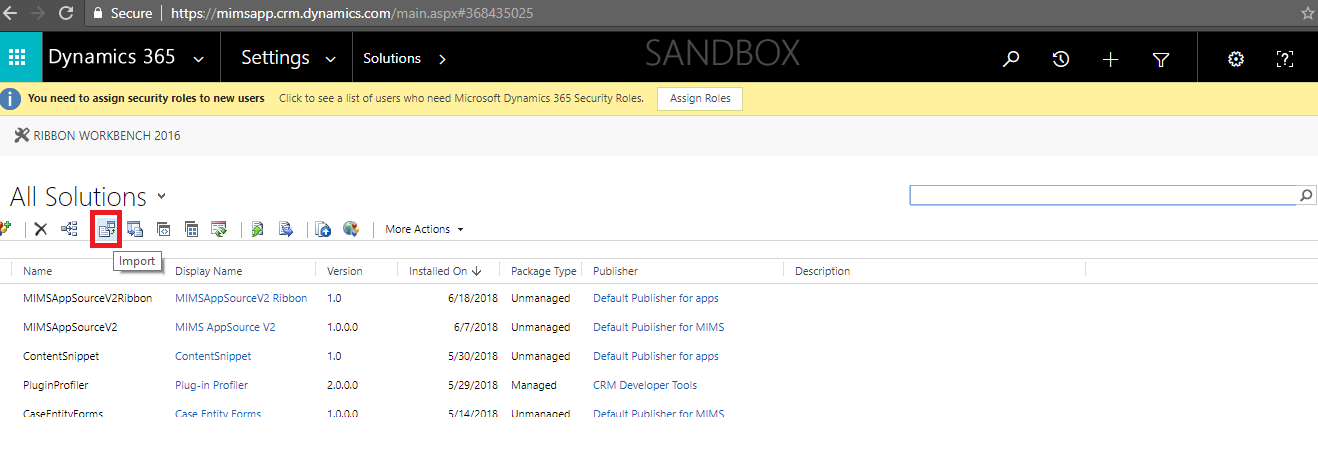
<https://www.powerobjects.com/powerpacks/powerautonumber>



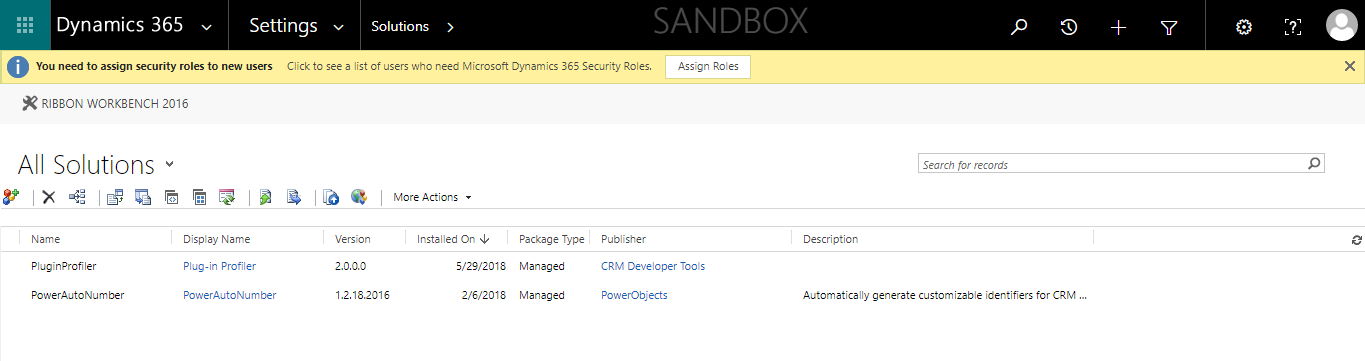
Import the same into the CRM system by navigating to **Settings - > Solutions.**



Click on **Import** button in the solutions page.



After importing the Power auto number, the same is available in the below imported list.

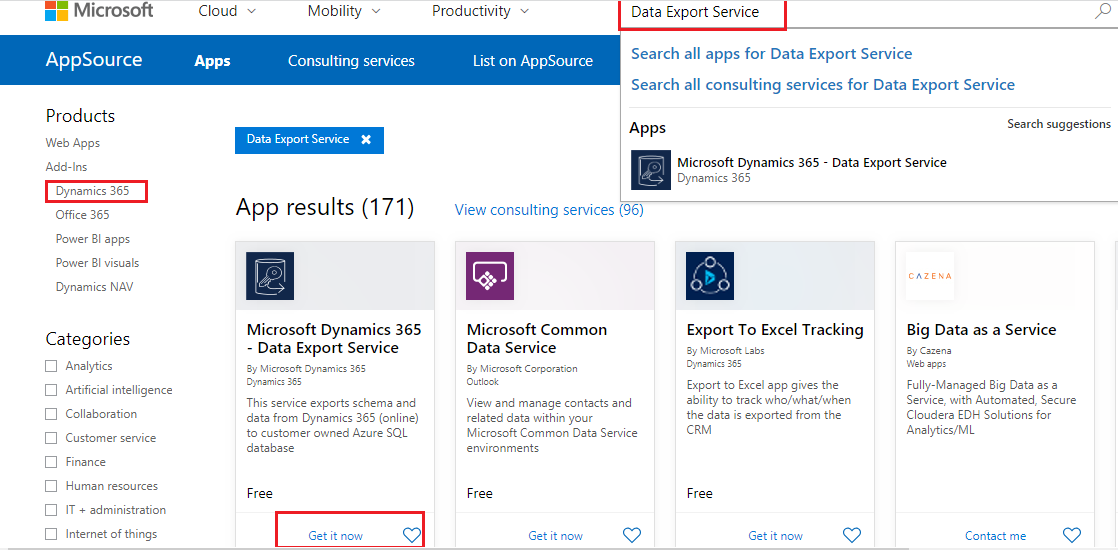


## 2.2 Data Export Service

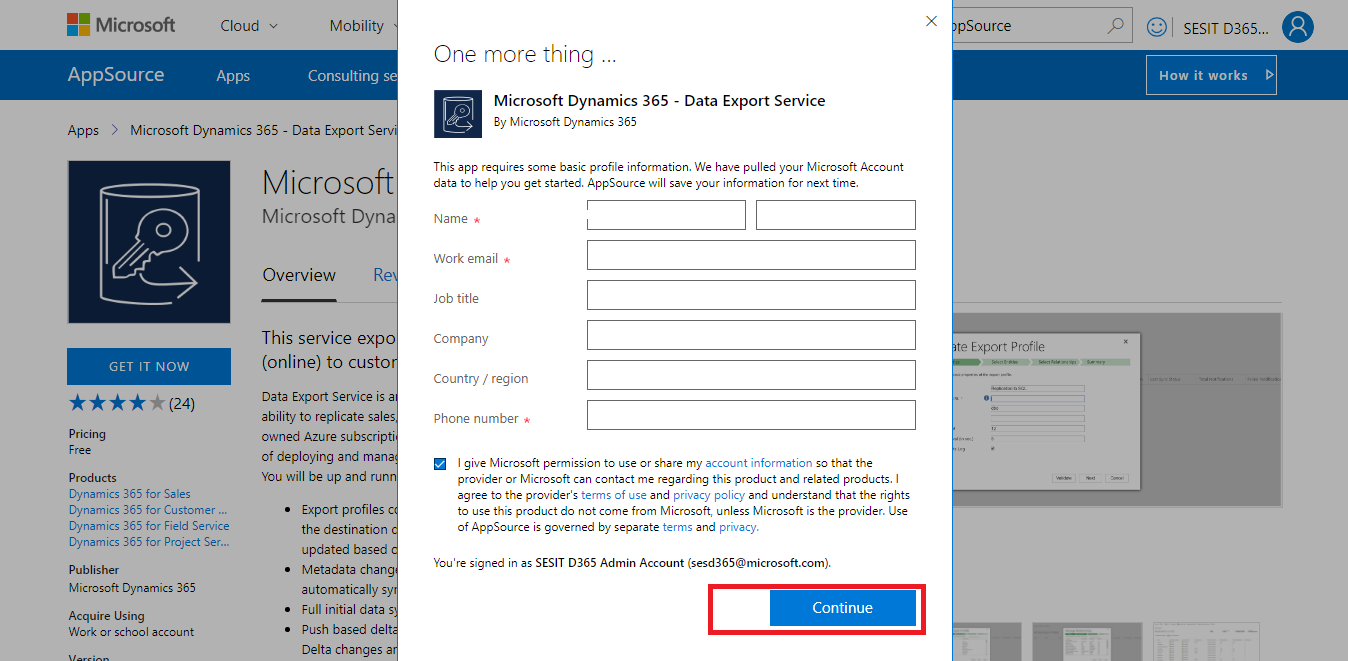
Data Export Service is an add-on service for Dynamics 365 (online) that provides the ability to replicate sales, service and marketing data to a SQL store in a customer-owned Azure subscription. It simplifies the technical and administrative complexity of deploying and managing a data export solution managing schema and data.

Navigate to the below **URL** , go to “Dynamics 365” Add-Ins and search with the name of “**Data Export Service**” as shown below

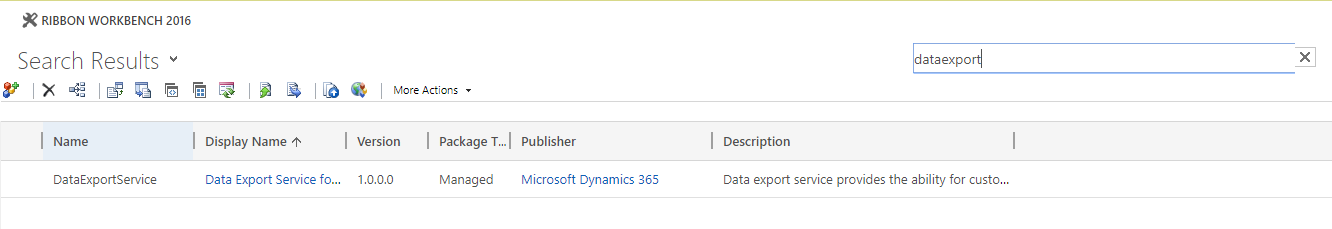
URL: <https://appsource.microsoft.com/en-us/marketplace>



The system will request the connection details like user name, organization name and other details as shown below. Enter the appropriate details and click on **Continue**.



The solution is imported into the system and the same is available in the Imported list as shown below.

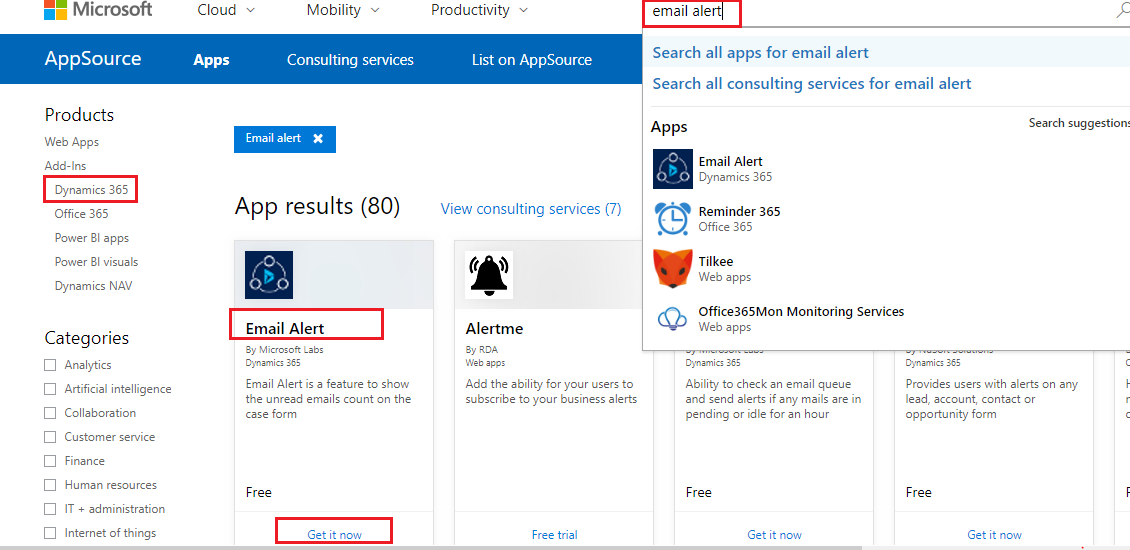


## 2.3 Email Lab Alerts

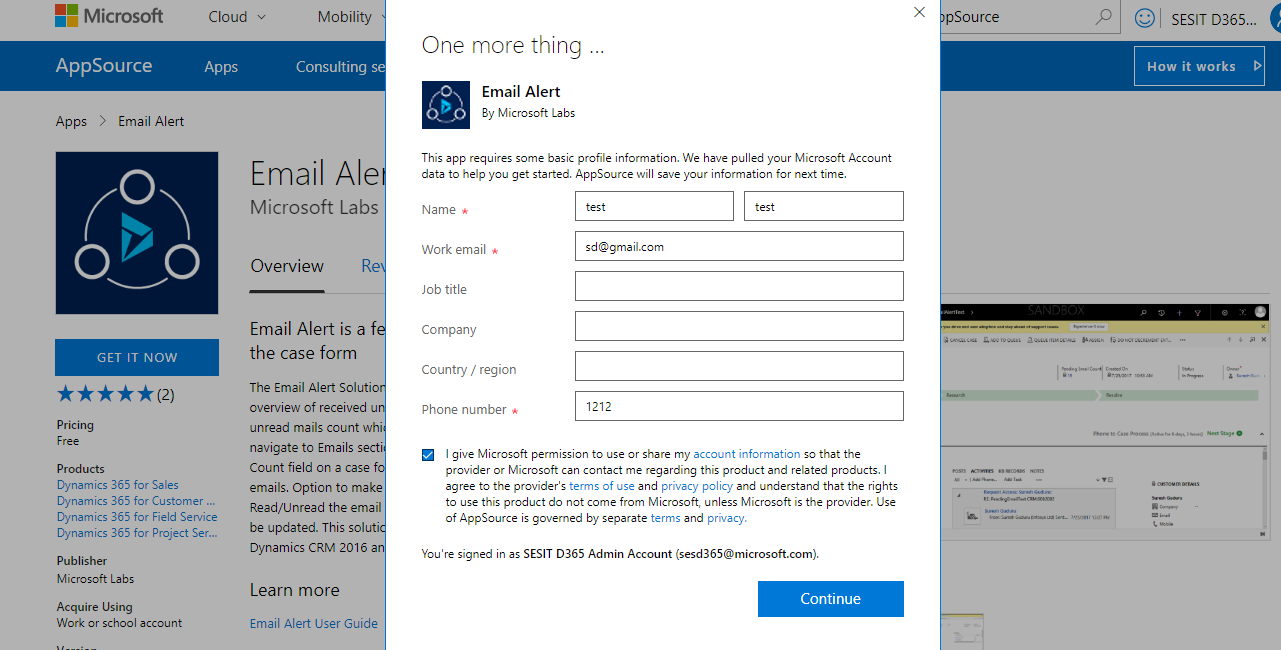
The Email Alert Solution is an add-on feature to dynamics 365 CRM which provides an overview of received and unread CRM emails on a Case. It enables users to get the unread mails count which were received on a case.

Navigate to the below **URL** , go to “Dynamics 365” Add-Ins and search with the name of “**Email Alert**” as shown below

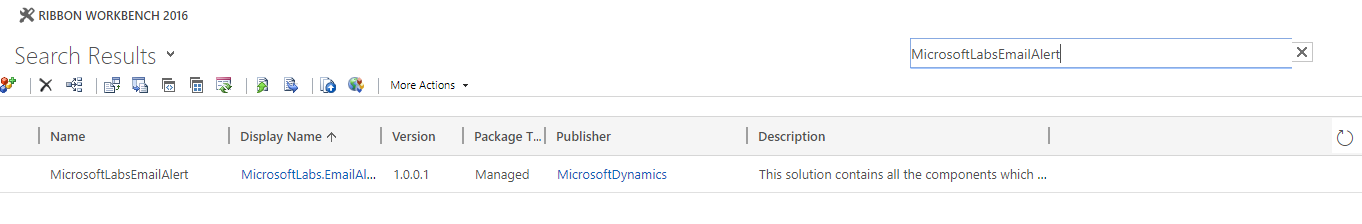
URL: <https://appsource.microsoft.com/en-us/marketplace>



The system will request the connection details like user name, organization name and other details as shown below. Enter the appropriate details and click on **Continue**.



The solution is imported into the system and the same is available in the Imported list as shown below.

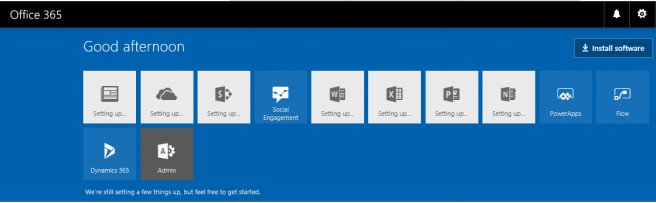


## 2.4 Voice of Customer (VOC)

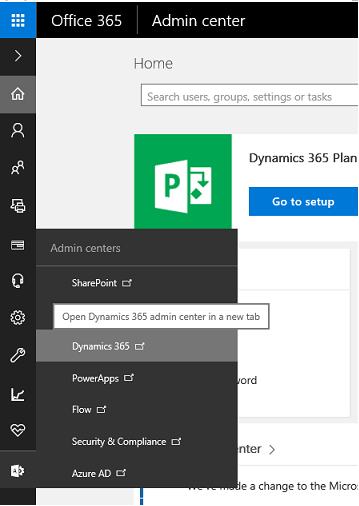
Voice of the Customer is a new Dynamics CRM 2016 feature that was created to capture feedback regarding the product or service provided by the CRM. With Voice of the Customer, enables setting up of surveys, survey distribution to contacts via workflows, and capture responses. These captured data can be utilized for generating reports and making appropriate changes to the product/service.

User can take advantage of the feature by installing Voice of the Customer solution. The Solution is available globally for Microsoft Dynamics CRM Online subscriptions.

1. Access the link, <https://portal.office.com/> and click on the ‘**Admin’** tile. User must be logged into the Office 365 Administrator account.



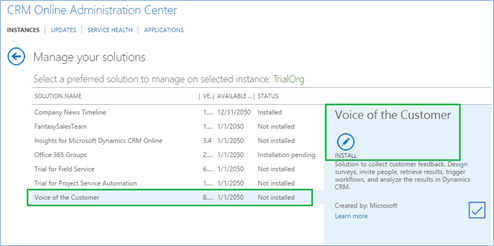
1. Click on **Admin Center** and select **Dynamics 365**.



1. Select the instance (organization) for which you want to install **Dynamics 365 Portal** and click on the **Edit** button, next to Solutions.



1. Select the **Voice of the Customer** and click on **Install.**

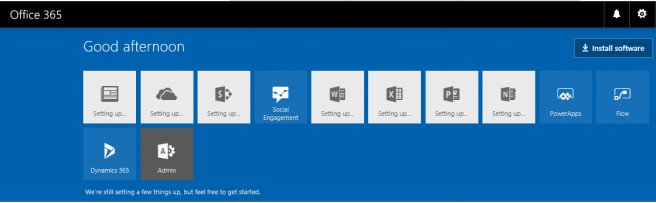


**Note**: When a solution is installed in CRM, the CRM Online site goes into maintenance mode and will be offline for a short period of time. It is advisable to install the solution in outside office hours.

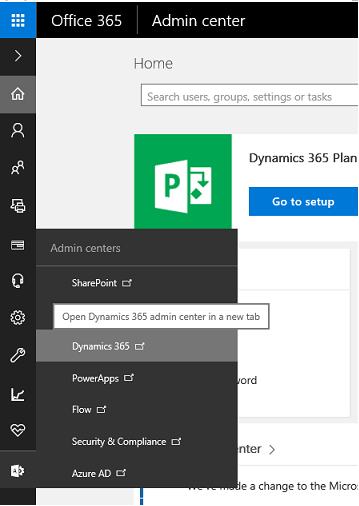
## 2.5 Portal Installation and Configuration

A portal is an independent running application that exposes specific CRM data and functionality to the respective users. It can be leveraged to model business processes that aren’t easily modeled inside CRM or to provide a higher level of integration between your CRM and your clients. Users can range from full CRM users to external clients with minimal rights and access. Below is the step by step process to install and configure the portal:

1. Go to <https://portal.office.com/>  and click on the ‘**Admin’** tile. User must be logged into the Office 365 Administrator account.



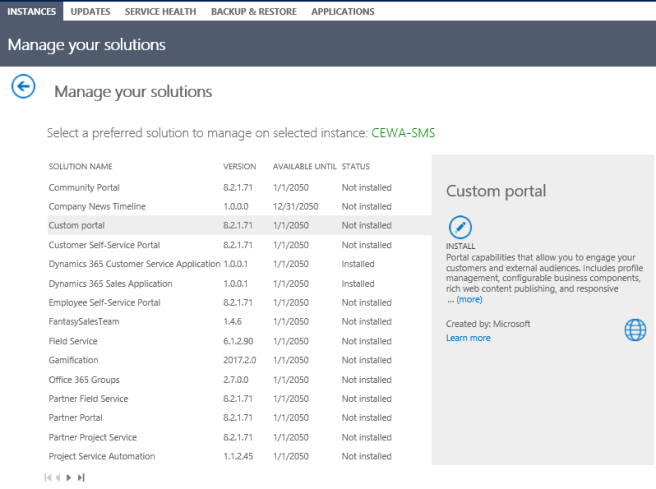
1. Click on ‘Admin Center’ and select ‘Dynamics 365’.



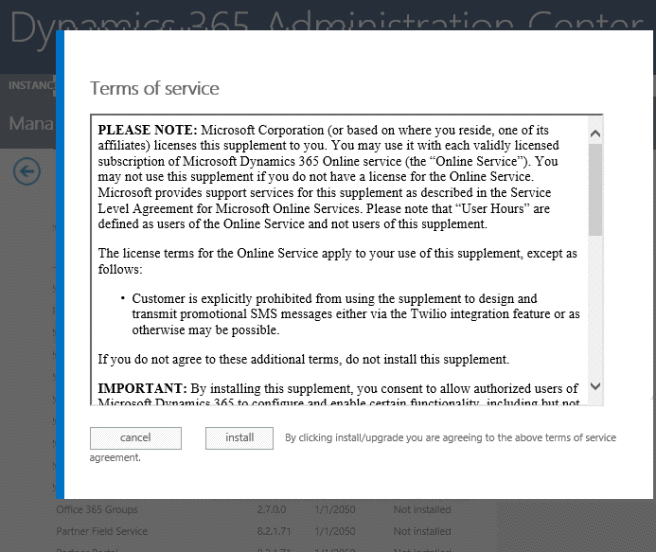
1. Select the instance (organization) for which you want to install ‘Dynamics 365 Portal’ and click on the ‘**Edit’** button, next to Solutions.



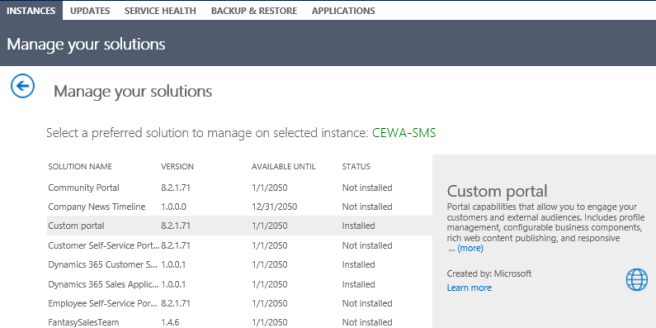
1. Select **‘Custom Portal’** from the list and click on the ‘**Install’** button.



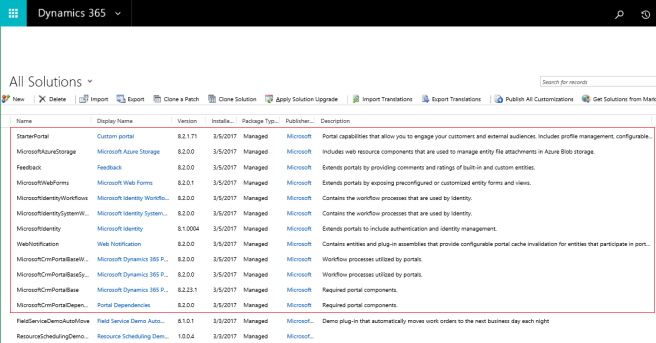
1. Click on ‘**Install’** button in the pop-up.



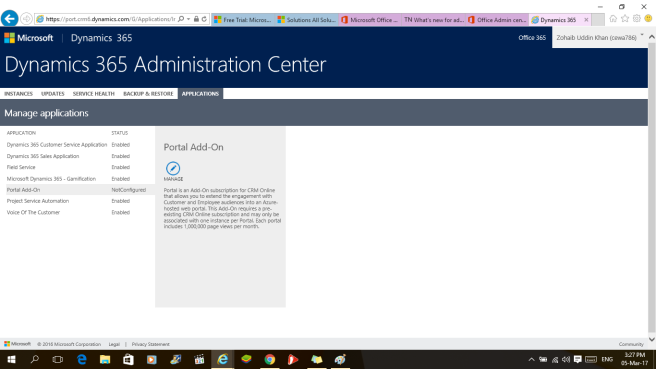
1. The Status changes to **‘Installed’.**



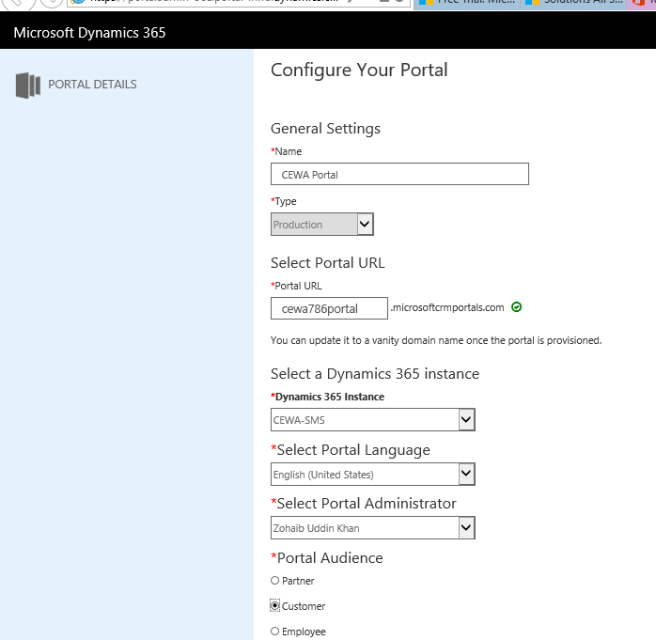
1. Navigate to Dynamics 365 CRM Organization to verify and validate all the Managed solution for the Custom portal installed with the above steps.



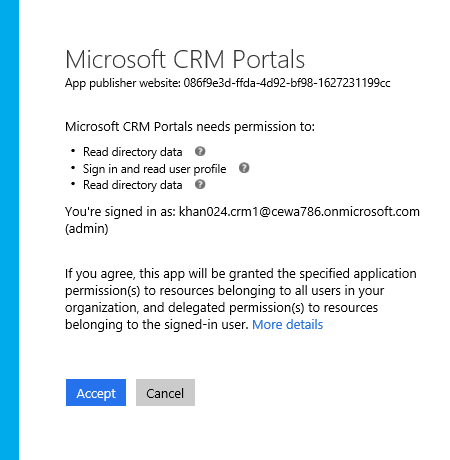
1. Access the website [https://portal.office.com](https://portal.office.com/), select **‘Admin Center’** and **‘Dynamics 365’** and click on ‘**Application’** tab.



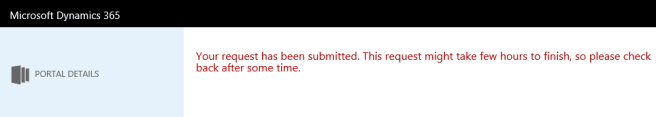
1. On the ‘**Configure your Portal’**, enter the necessary information. Enter the below details:
   1. **Portal Audience**: Customer
   2. **Portal Binding**: Custom



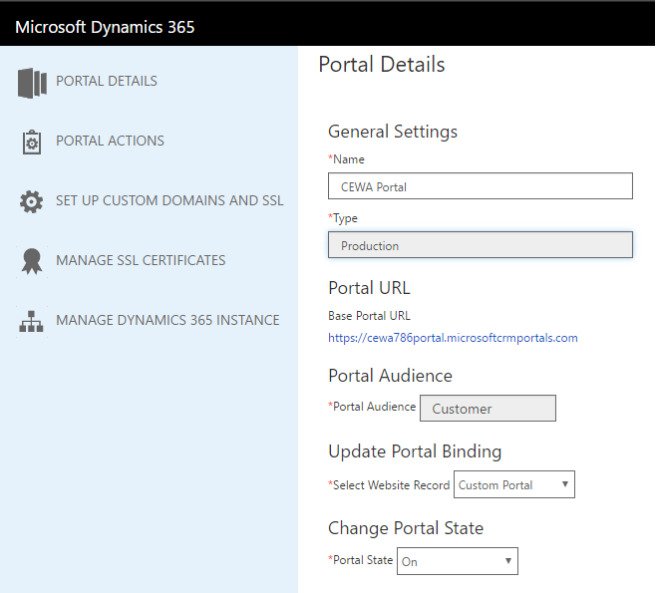
1. Click on ‘**Accept’** in the Permissions pop up for Dynamics 365 Portal.

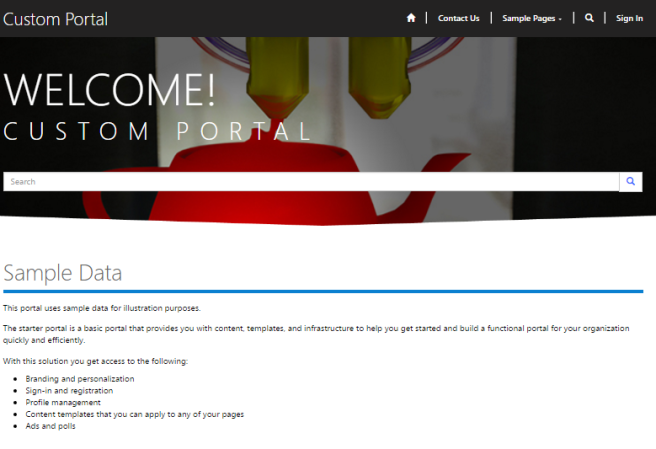


1. Navigate to ‘Portal Details’. As appearing in the below message, it might take time to configure the portal, check back in some time and the ‘Custom Portal’ will be ready for use.



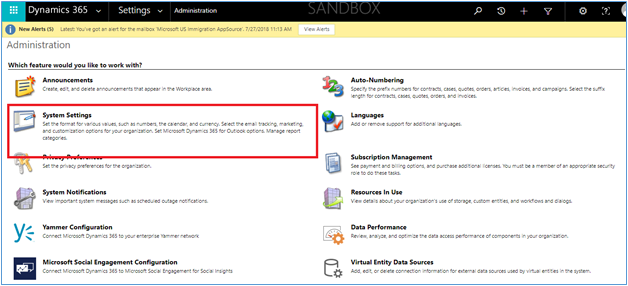
1. Once the portal set up is complete the page wil display the below set of options.



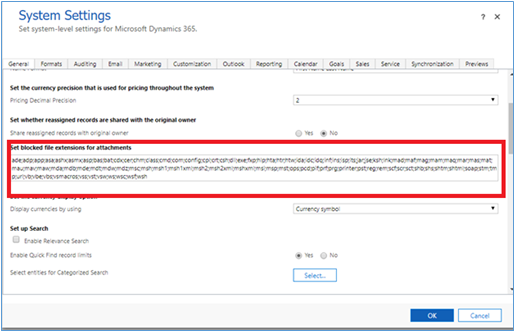
1. Click on the ‘Base Portal URL’, a new window will open and the Dynamics 365 Custom Portal is ready for Branding.

## 2.6 System Settings

As a part of deployment process, the following changes are required in the general tab of system setting. To access system settings, navigate to Setting - > Administration - > System Settings:



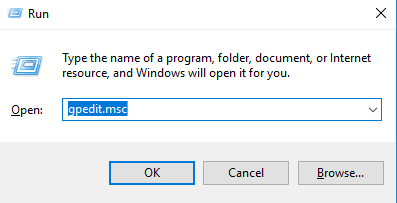
Navigate to general tab - > Set blocked extensions for attachments. Remove JS from the extensions.



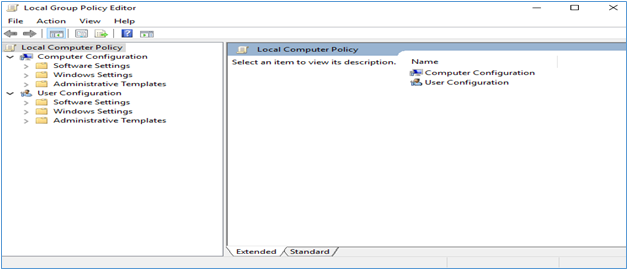
## 2.7 Group Policy Set up

Below settings are required to run the script and power shell commands for setting up IMS Apps from Package deployer

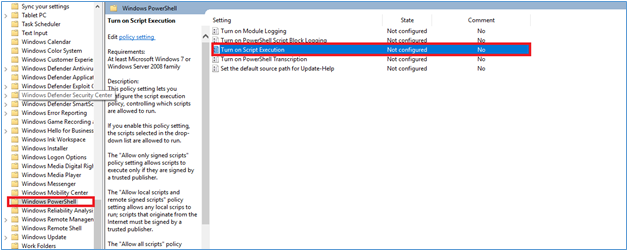
* Execute the “**gpedit.msc**” in windows run and open the Group Policy Settings as shown in the below screenshot.



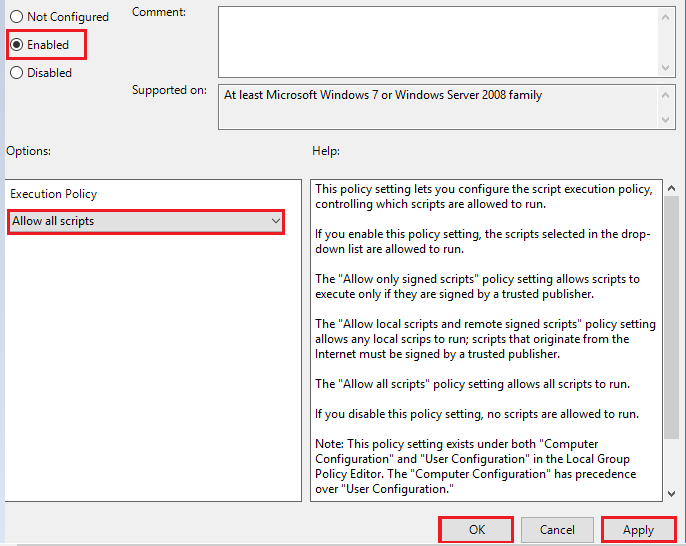
* Click on Ok



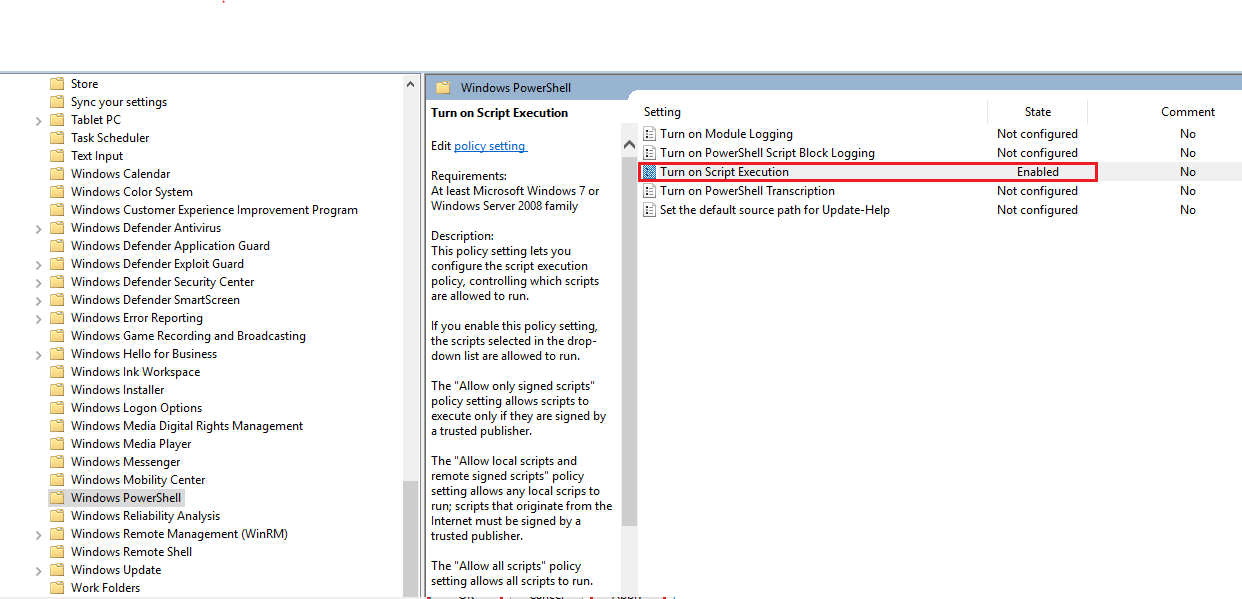
* Navigate to Computer Configuration -> Administrative Templates ->Windows Components -> Windows PowerShell as shown below



* Open “Turn on Script Execution” and change the settings as shown below

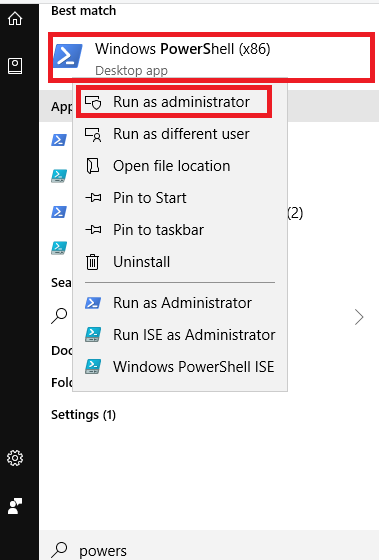


* Now the script execution state changes from Not configured to Enabled as shown below



## 2.8 Windows PowerShell

* Open PowerShell window from windows as shown below

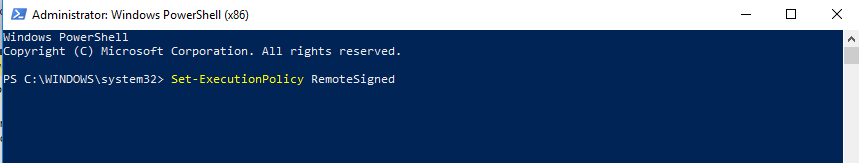


* It opens the below PowerShell window

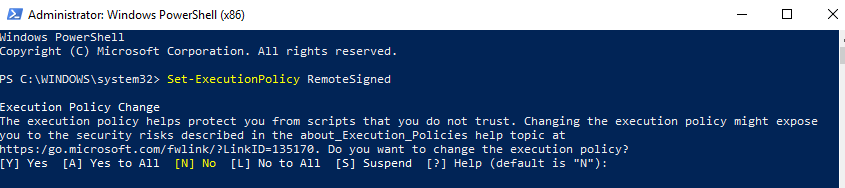


* Run the below Commands

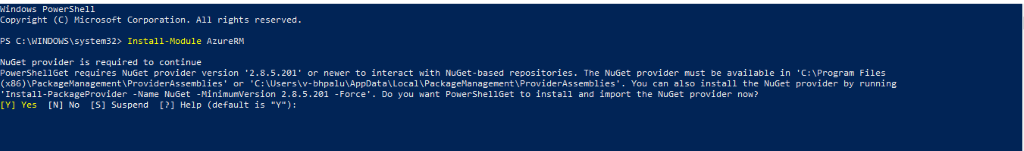
*“***Set-ExecutionPolicy RemoteSigned***” as shown below*



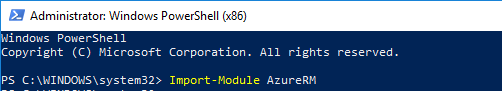
* *The following question appears on screen, “Do you want to change the execution policy” type “Y” and click enter*



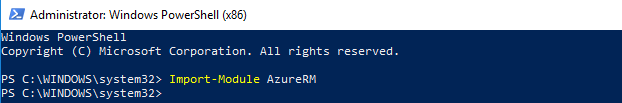
* *Run the command “****Install-Module AzureRM****” as shown below*



* *Type “****Y****” and click enter*
* *Run the command “***Import-Module AzureRm**” as shown below



* Wait to execute the import module azureRM until shows the next line as shown below



## 2.9 Azure Setup

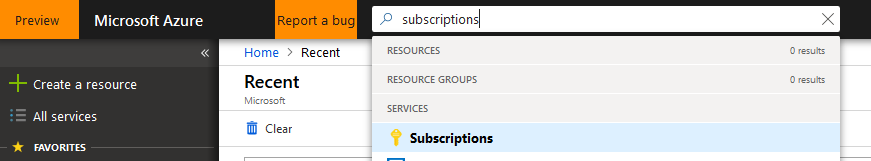
Use the below link to get more information about the azure subscription

<https://azure.microsoft.com/en-in/features/azure-portal/>

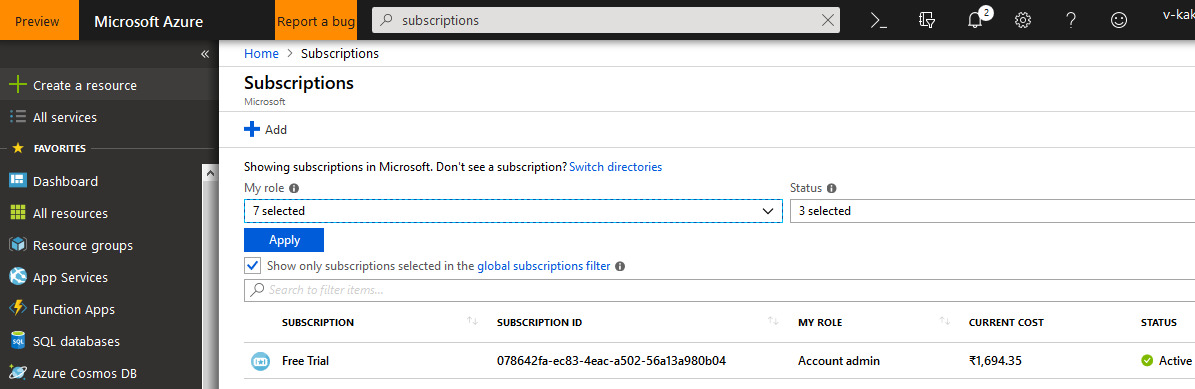
* Login to azure portal using <https://portal.azure.com>
* Sign-In to portal using service admin/co-admin role

**How to get the Subscription Id** **?**

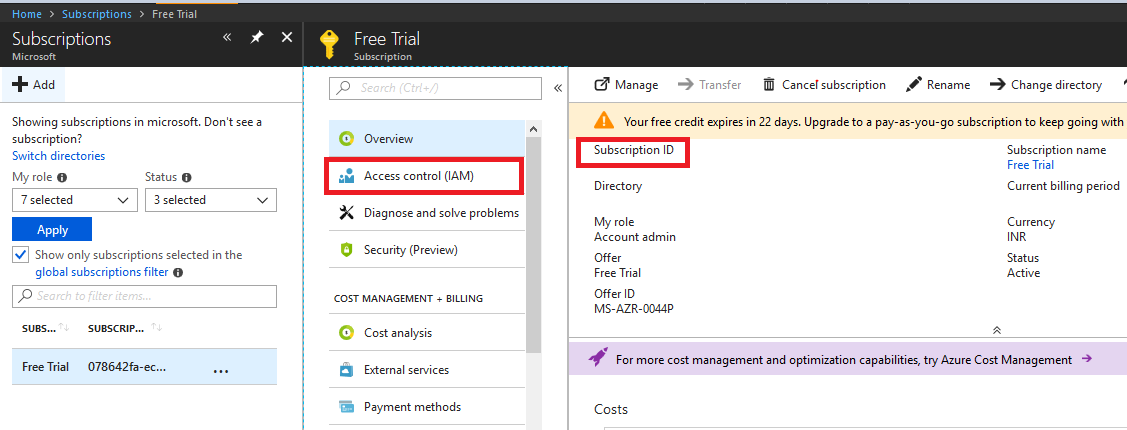
* Search for Subscriptions like showing below .



* Select the required subscription from the list below

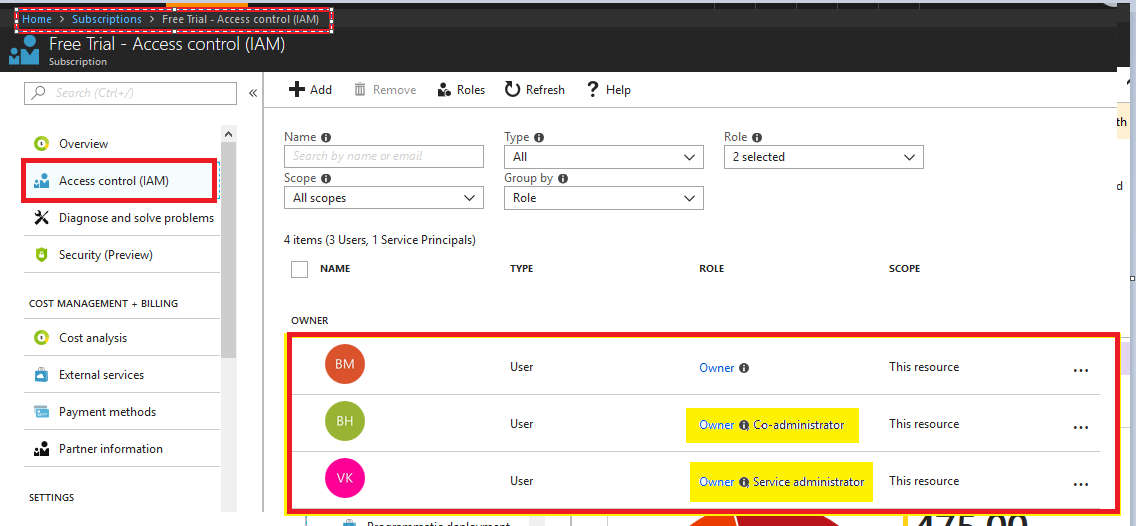


* On click of the subscription we can find the subscription id



**How to find Login User role/Powershell script executing user role ?**

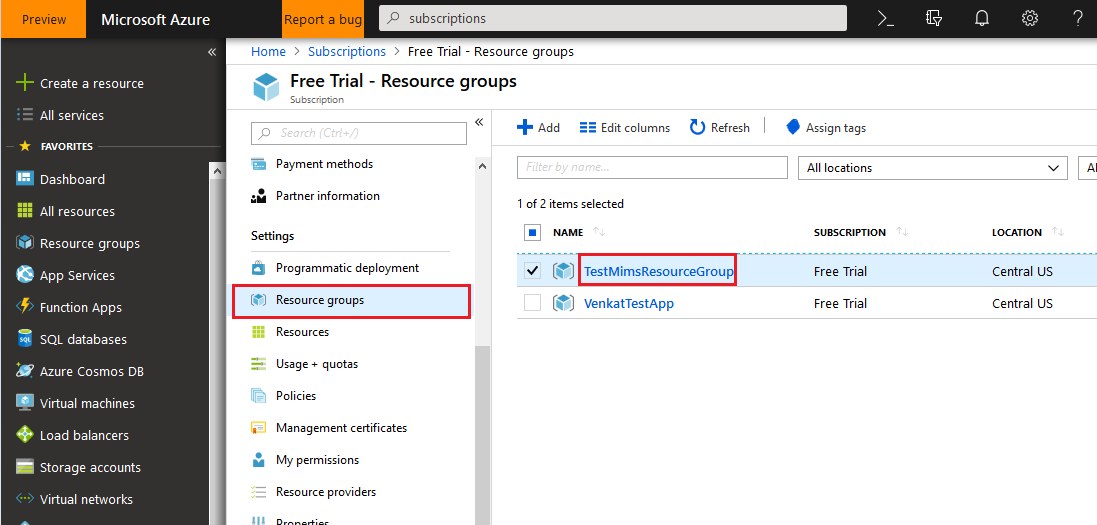
* Click on Subscription name and select AccessControl(IAM) from left navigation links



* Find the user and his roles like above (powershell script executing user has to be under co-admin/service admin role )

**How To find the Resource Group ?**

* Click on Subscription name and select ResourceGroups from left navigation links



# 3. Microsoft Dynamics CRM Package Deployer

Microsoft Dynamics CRM Package Deployer lets administrators deploy packages on Microsoft Dynamics 365 (online) and Dynamics 365 (on-premises) instances. A “package” can consist of any or all of the following:

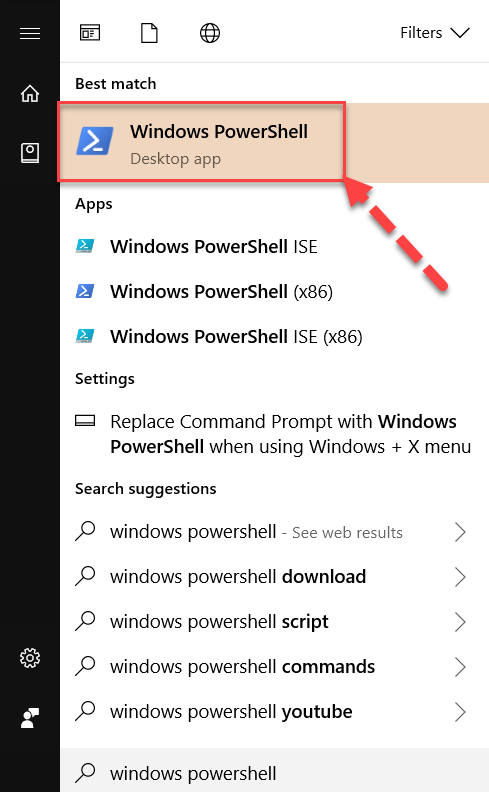
1. One or more **Dynamics 365 solution files**.
2. **Flat files** or exported configuration data file from the Configuration Migration tool.
3. **Custom code** that can run before, while, or after the package is deployed to the Dynamics 365 instance.
4. **HTML content** specific to the package that can display at the beginning and end of the deployment process. This can be useful to provide a description of the solutions and files that are deployed in the package.

## 3.1 Download Package Deployer

Follow the below steps to download the Tools,

**Step 1**: Create a folder in any drive and name it as “Dynamics\_365\_Development\_Tool “

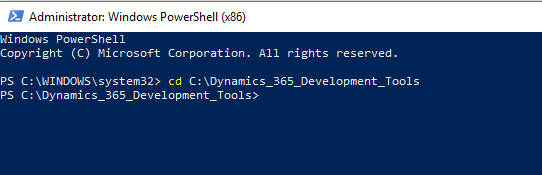
**Step 2**: Search for Windows PowerShell and open it.



**Step 3**: Change the working directory to the downloaded folder in step #2.

cd <drive>:\Dynamics\_365\_Development\_Tools

Example : created folder with name ‘Dynamics\_365\_Development\_Tools’ in ‘C’ drive as shown example below



**Step 4**: Copy & Paste the below PowerShell script in PowerShell Window to download tools from Nuget.

$sourceNugetExe = "https://dist.nuget.org/win-x86-commandline/latest/nuget.exe"

$targetNugetExe = ".\nuget.exe"

Remove-Item .\Tools -Force -Recurse -ErrorAction Ignore

Invoke-WebRequest $sourceNugetExe -OutFile $targetNugetExe

Set-Alias nuget $targetNugetExe -Scope Global –Verbose

**Step 5:** To download Package Deployer, Copy & Paste the below PowerShell script in PowerShell Window to download tools from Nuget.

./nuget install Microsoft.CrmSdk.XrmTooling.PackageDeployment.WPF -O .\Tools

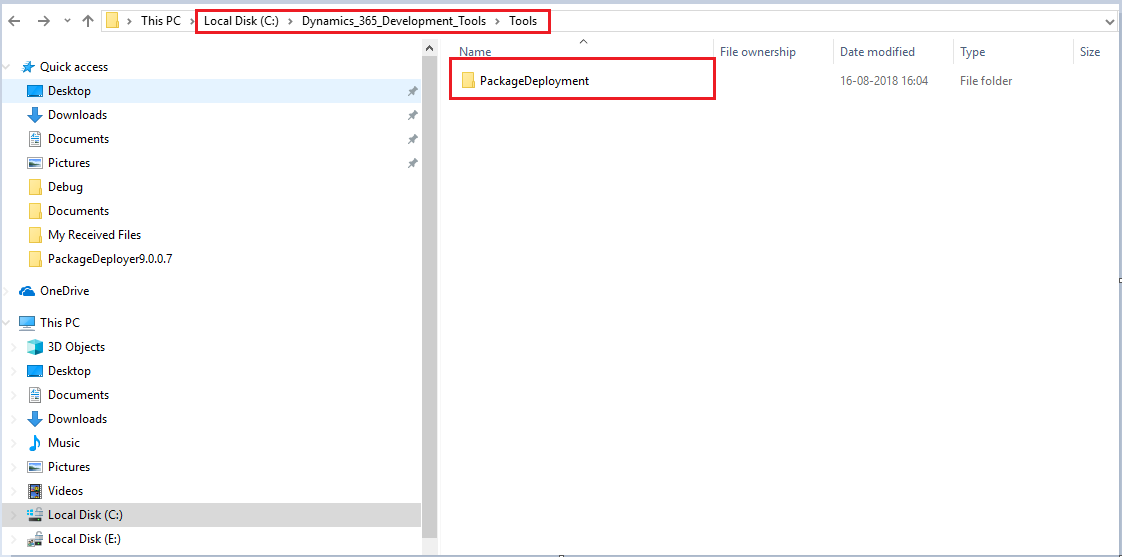
md .\Tools\PackageDeployment

$pdFolder = Get-ChildItem ./Tools | Where-Object {$\_.Name -match 'Microsoft.CrmSdk.XrmTooling.PackageDeployment.Wpf.'}

move .\Tools\$pdFolder\tools\\*.\* .\Tools\PackageDeployment

Remove-Item .\Tools\$pdFolder -Force -Recurse

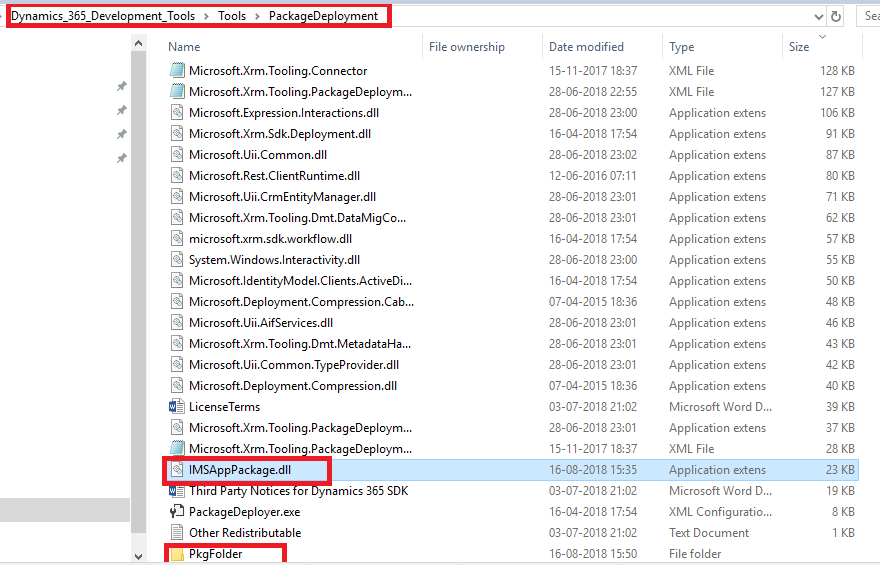
The package will be available as shown below.



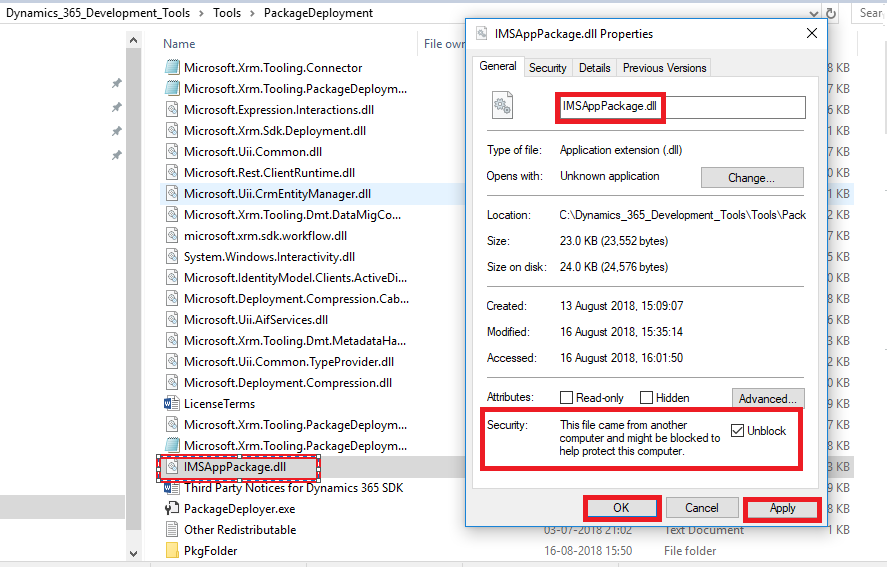
# 4.Download the IMS App Package

Download the IMS App package from Microsoft AppSource portal and Unzip the package which consists of *PkgFolder* and *IMSAppPackage.dll*

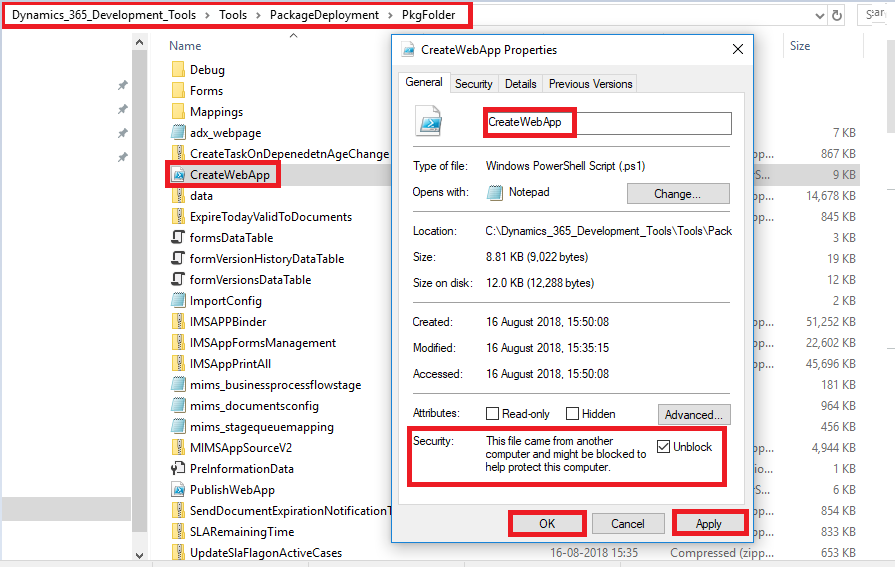
* Copy both the folder by name “PkgFolder” and “IMSAppPackage.dll” to the directory <drive>:\Dynamics\_365\_Development\_Tools\Tools\Package Deployment\ as shown below



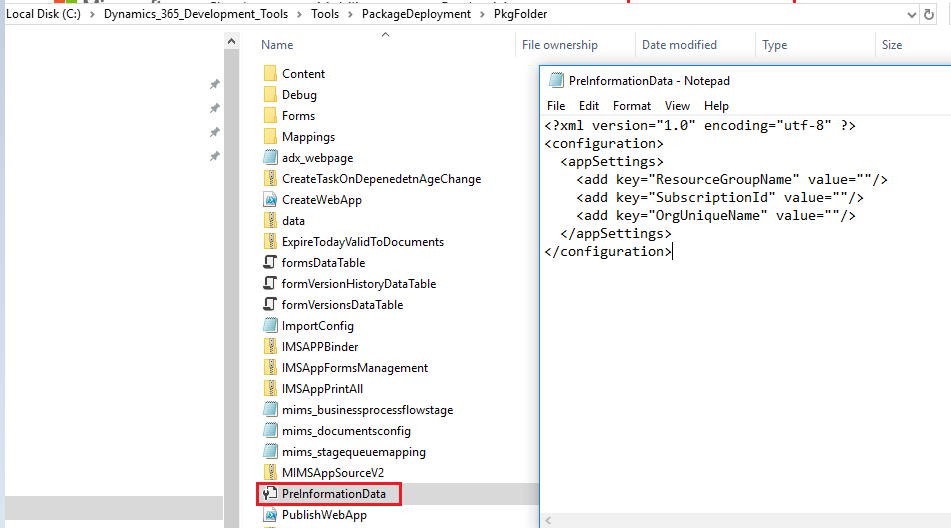
* Unblock the dll by right click on the dll file, check it as shown below and click Apply.



* Open the Package Folder (PkgFolder) and Unblock all the files, as shown below (example is shown for one file, user need to repeat this for all the blocked files under PkgFolder)

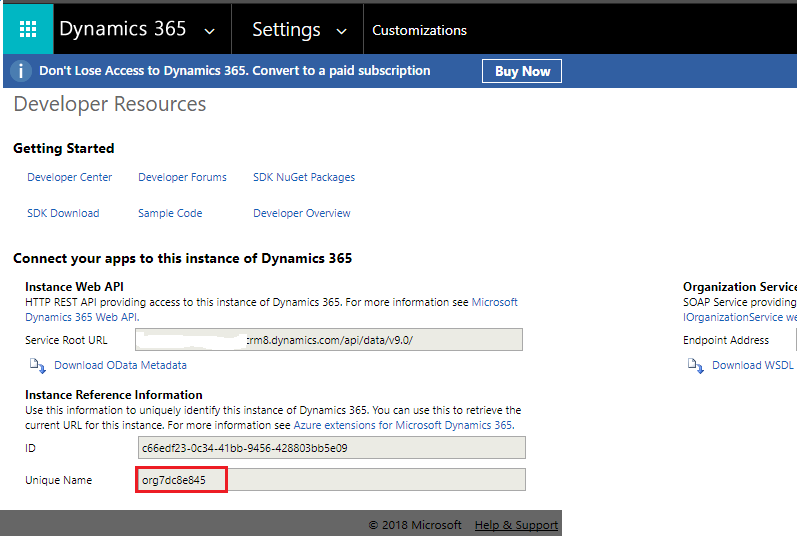


* Open the “PkgFolder” in package Deployer tool and open the file “PreInformationData” as shown below



**How to get Organisation Unique Name ?**

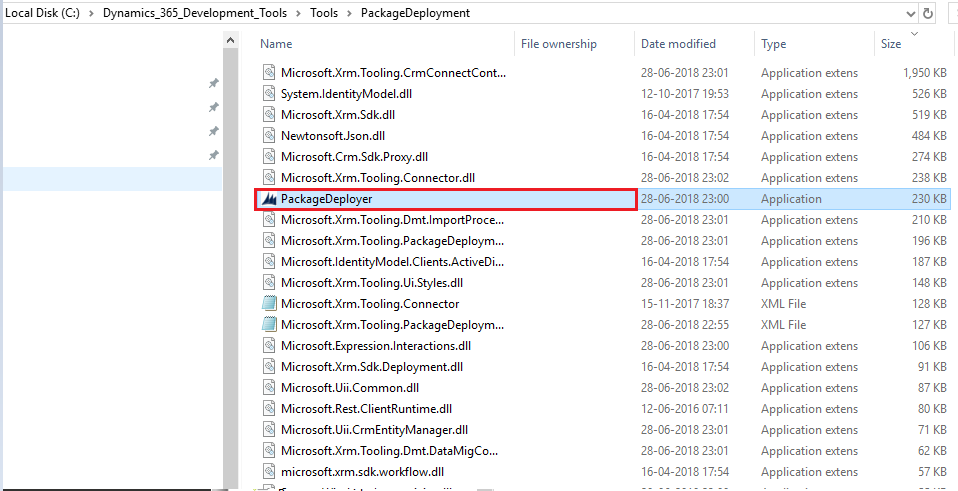
* In CRM Navigate to Settings - > Customizations -> Developer resources as shown below:



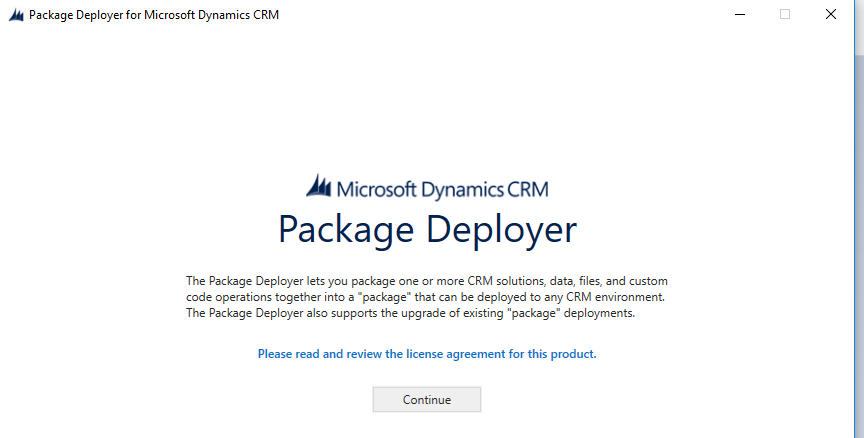
* Resource Group Name and Subscription Id are mentioned in 2.9 Azure Setup
* Mention the “ResourceGroupName” , “SubscriptionId” and Organisation uniqueName as shown below image



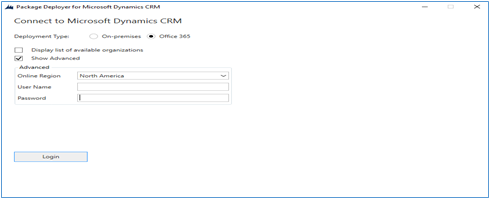
* + Update the above file with Resource Group,SubscriptionId ,OrganizationName values save and close the file
* Go back to “Package Deployment” Folder and run the package deployer as shown below:



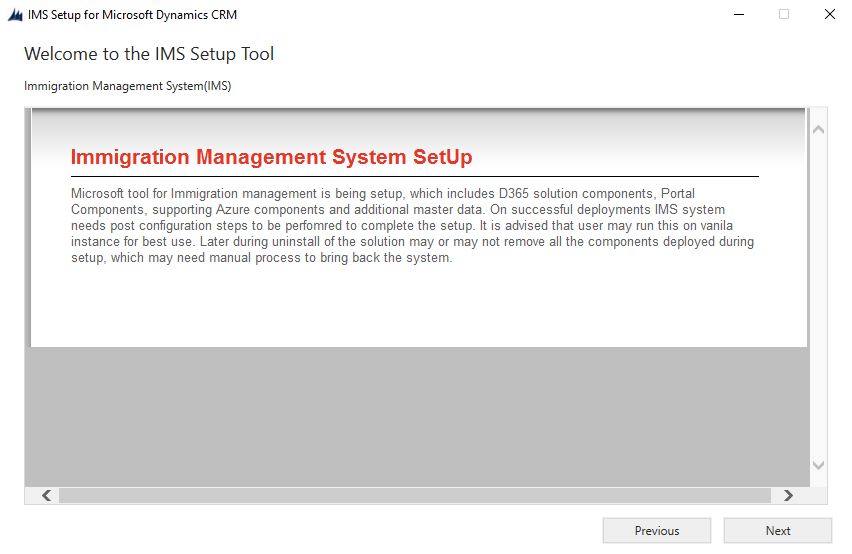
* Click on **Continue**.



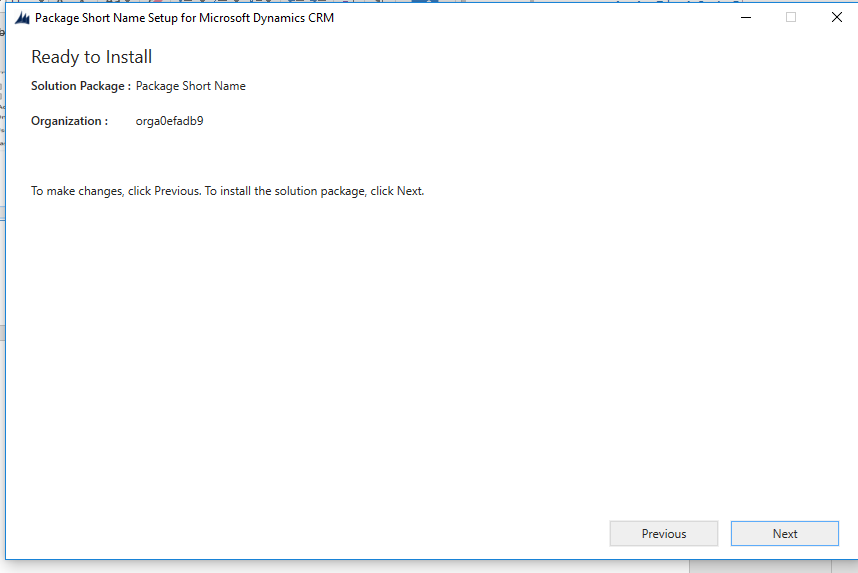
* Enter the CRM **user name** and **password** , Click on **Login**.



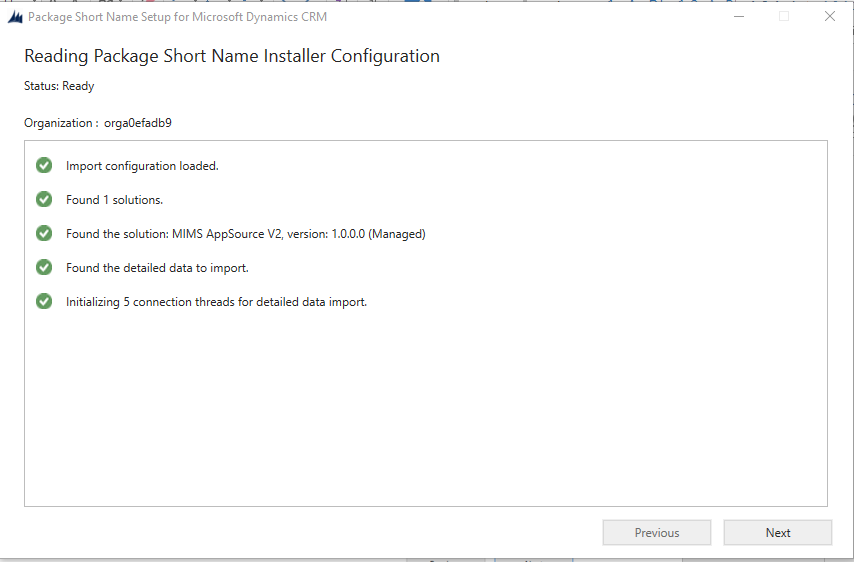
* After Login get a below screen which mention the Immigration Management System setup details and click on **Next**



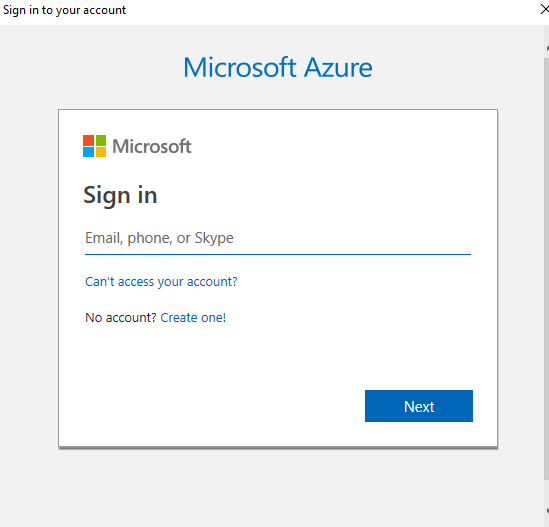
* The organizational details and solution package are displayed.



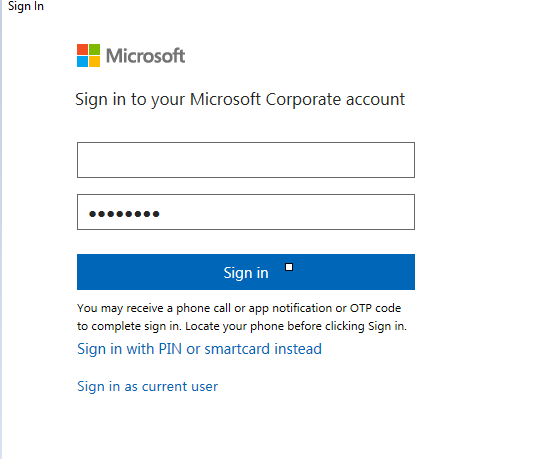
Click **Next** to see the package details:



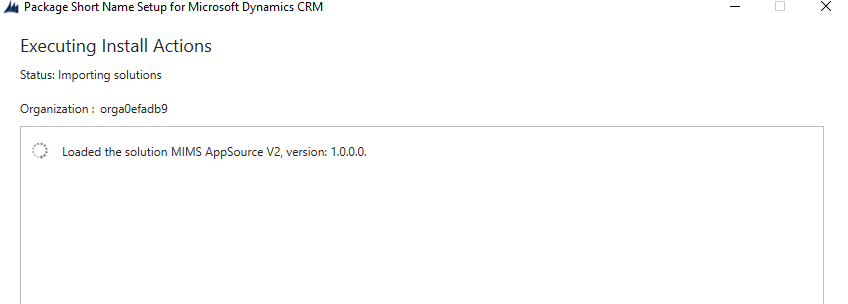
* Login to azure using service admin/co-admin role to deploy web applications



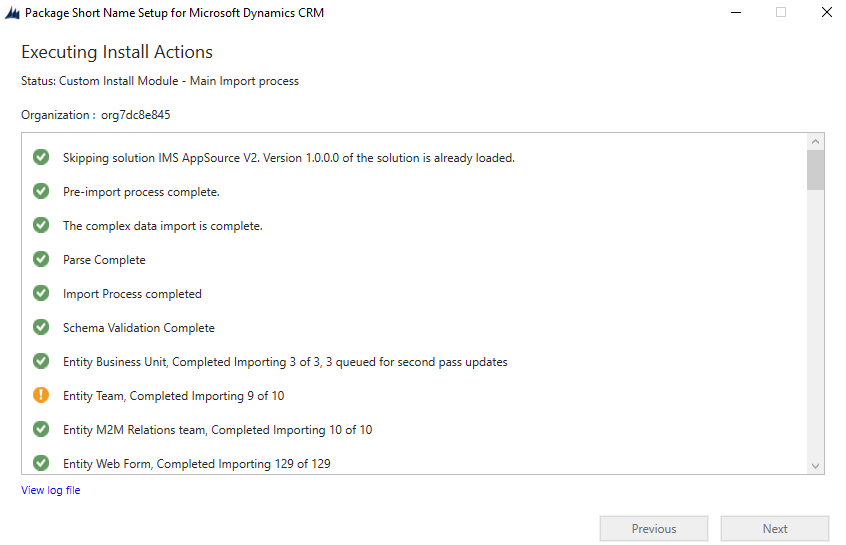
* Enter the azure UserName and click on Next shows below screen



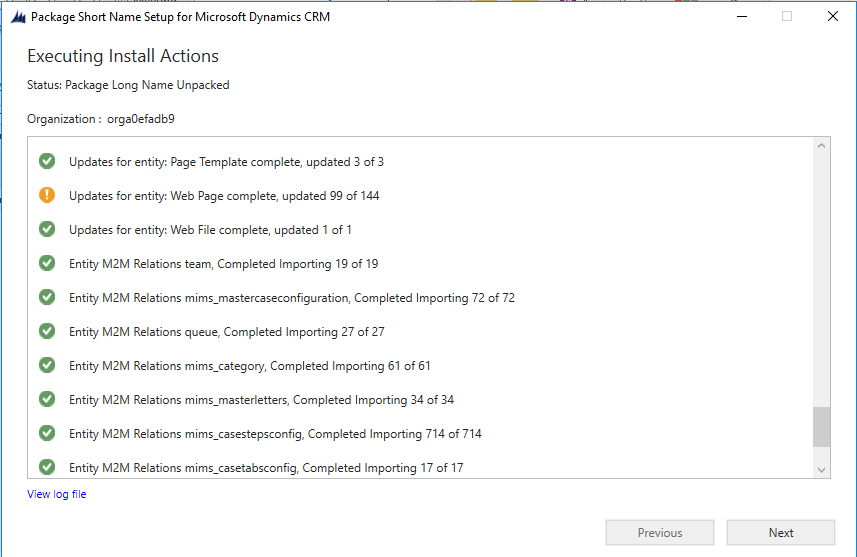
* After Azure login was successful then the solution import started like below



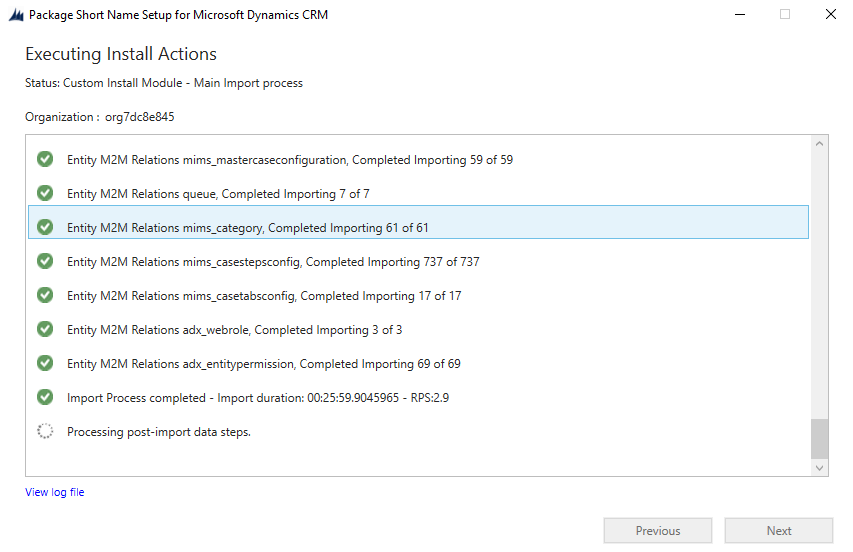
* After the solution is imported, data import will initiate as shown below.



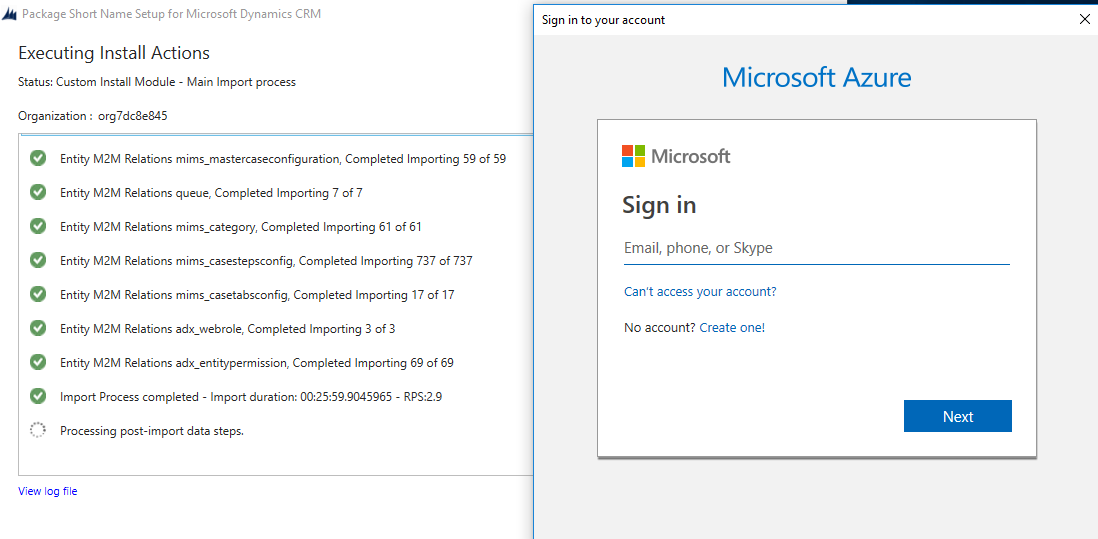
* Updating the many to many relationships as shown below



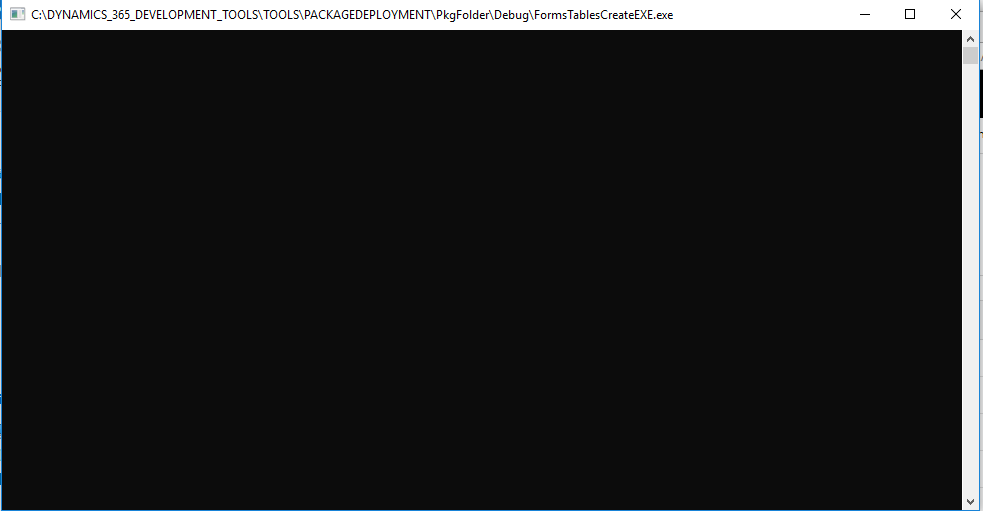
* Post import data steps are started as like below



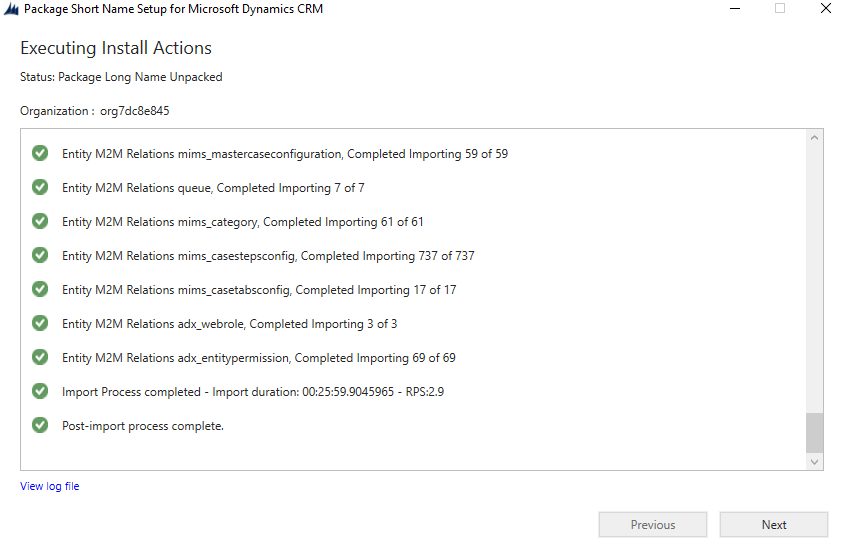
* Login to azure using service admin/co-admin role to Publish the web applications



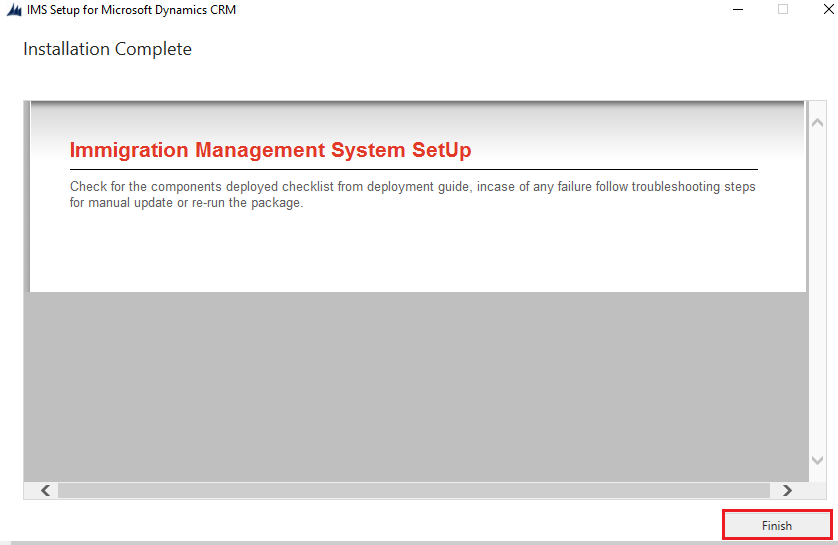
* After login to azure an executable file is running to create forms table in azure



* Post – Import process Completed as shown below



* Click on **Next** and gets the below screen



* Click on **Finish** completes the deployment process**.**

Go to CRM --> Settings --> Solutions --> **PUBLISH ALL**

# 6.Post Deployment Configuration

Below applications will be created in azure

**Note:** If your organization name is “org123456789” then <xxxxx> will <12345>

**App Service plan**

1) IMSAppServicePlan

**WebApp**

2) “IMSAppPrintAll<xxxxx>”

3) “IMSAppBinder<xxxxx>”

4) “IMSAppForms<xxxxx>”

5) “IMSAppWebJob<xxxxx>”

**Storage Account**

6) **“**imsappformstrg<xxxxx>”

7) ”imsappstorageacc<xxxxx>”

**KeyVault**

8) “ImsAppKeyVaults<xxxxx>”

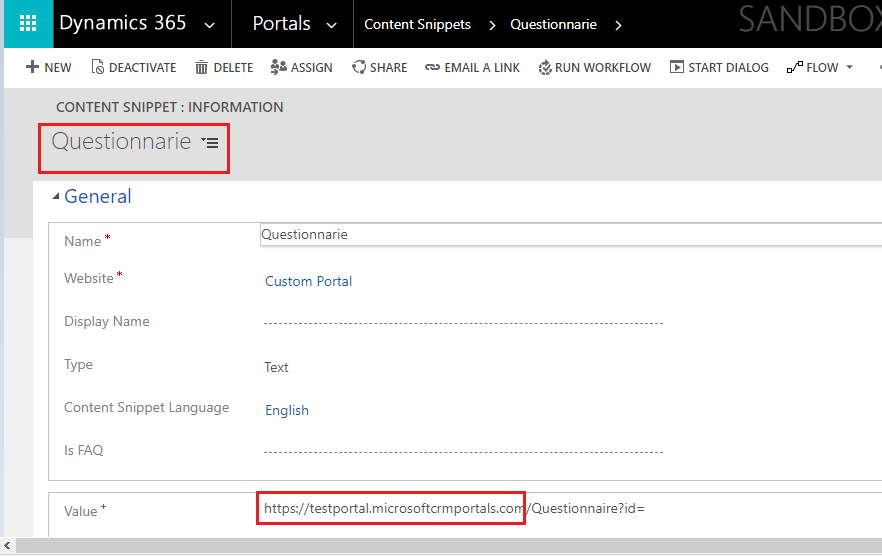
**App Insight**

9) “ImsAppInsight<xxxxx>”

**Note:** Follow troubleshooting steps in case of any of the above applications failed during deployment.

## 6.1 CRM Configurations

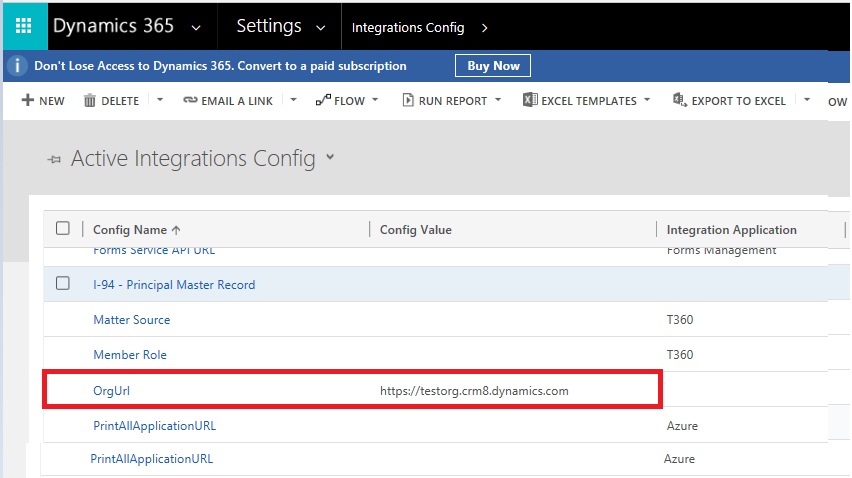
* Adding portal URL in content snippet entity in Questionnaire record as shown below
* Navigate to CRM -> Portals -> Content Snippets
* Open “Questionnarie” record and update the value field as shown below



* Change the URL with your organization configured CRM portal as given below
* **“CRM Portal URL”**/Questionnaire?id=
* **Example**: https://<your portal >.microsoftcrmportals.com/Questionnaire?id=

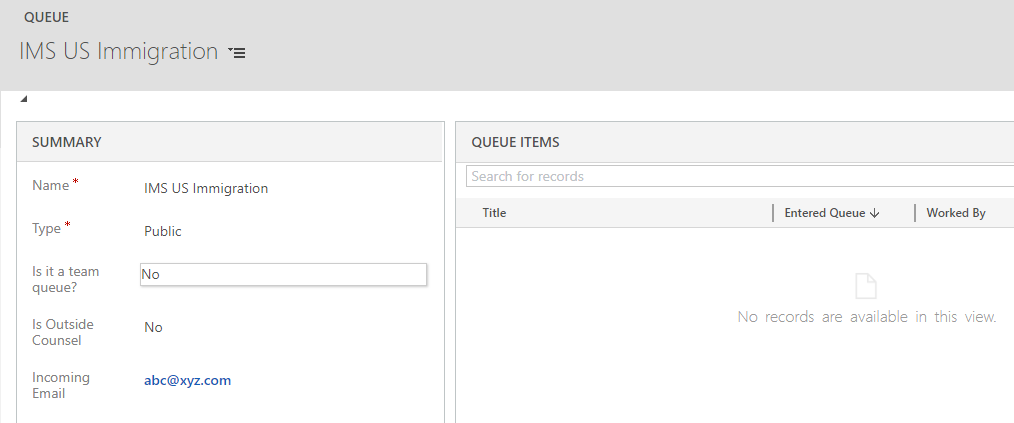
**Integration Configuration**

* Navigate to CRM -> Settings -> IntegrationsConfig
* Open “OrgUrl” record and update the Config Value field with organization URL as shown below



**Queue and MailBox Configuration**

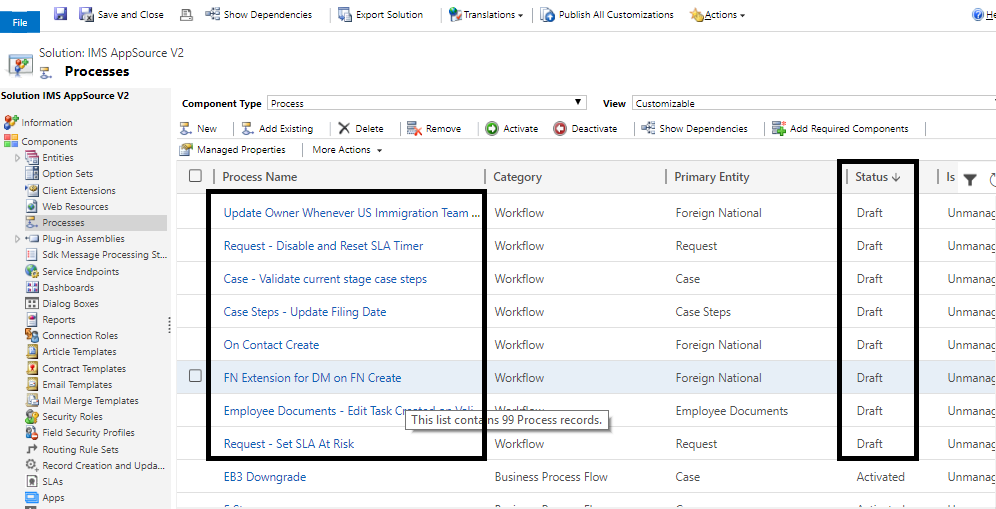
Configure below queue and mailbox for all the communication through IMS CRM instance



**Workflow Activation:**

Go to CRM settings 🡪 solutions 🡪 open “**IMSAppSourceV2”** 🡪 select “*processes”* under components as shown in below screen.

Note all draft processes and now go to settings 🡪 processes and activate them.



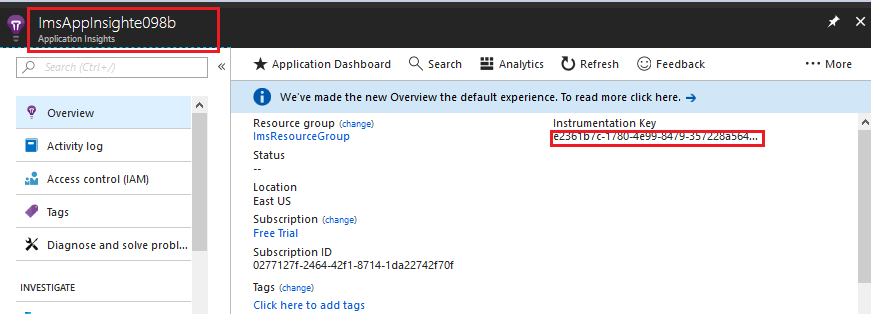
**Documents Storage:**

* Navigate to CRM -> Settings -> Extensions -> Integration Config
* Update the below config name values

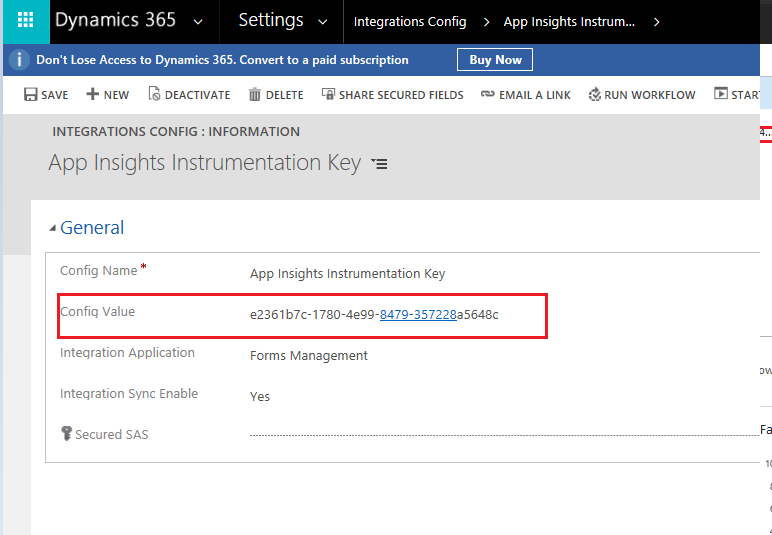
NOTE : Azure component names appear on document are same but on azure portal component names will end with your CRM Organization Unique Name (for ex:your CRM org name:org12345678 , on the Azure portal component names end with ComponentName12345 like : IMSAppPrintAll12345)

1) App Insights Instrumentation Key

* Navigate to Azure
* Open the Resource “IMSAppStorageAppInsight<xxxxx>” as shown below

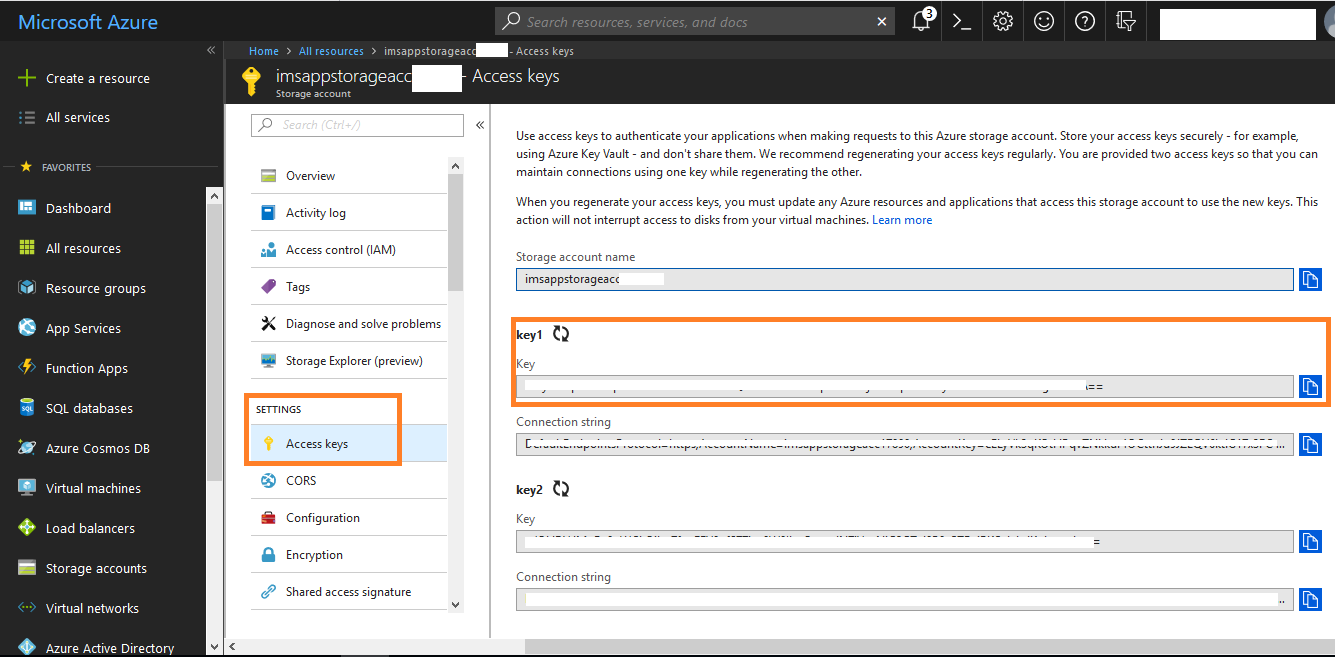


* Copy the Instrumentation Key and update the Config Value in App Insights Instrumentation Key as shown below

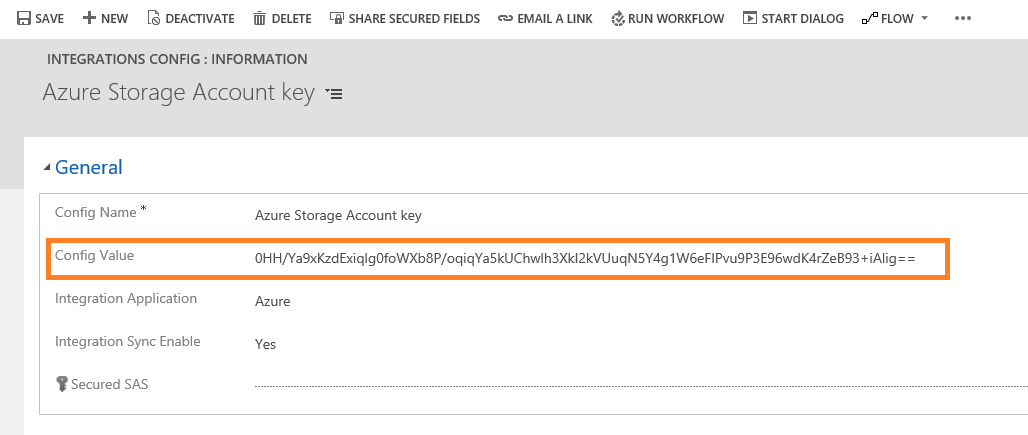


2) Azure Storage Account key

* Navigate to Azure
* Open the Resource “imsappstorageacc<xxxxx>” as shown below
* Go to Access Keys under Settings, Copy **key1 val**

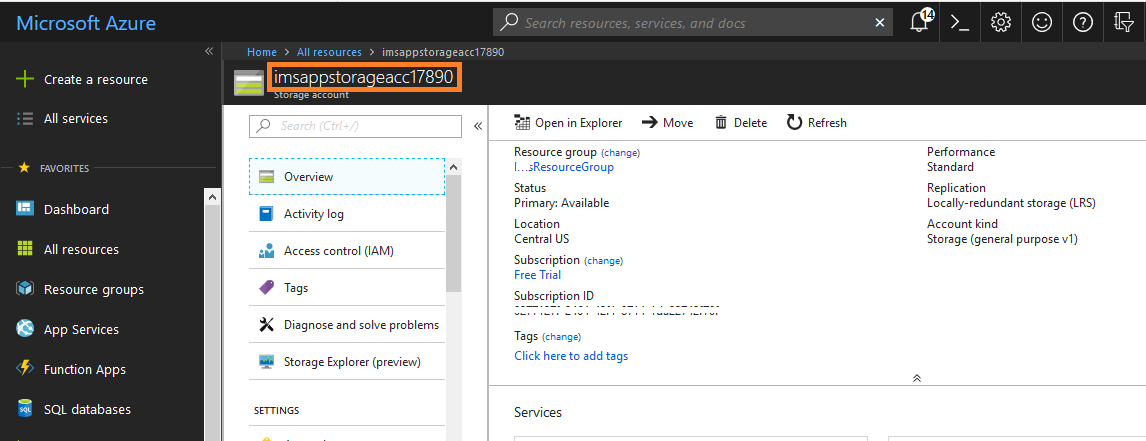


* Navigate to Settings -> Integrations Config Entity and Update the Key Value in **Azure Storage Account key**  Record “**config value”** field as shown below

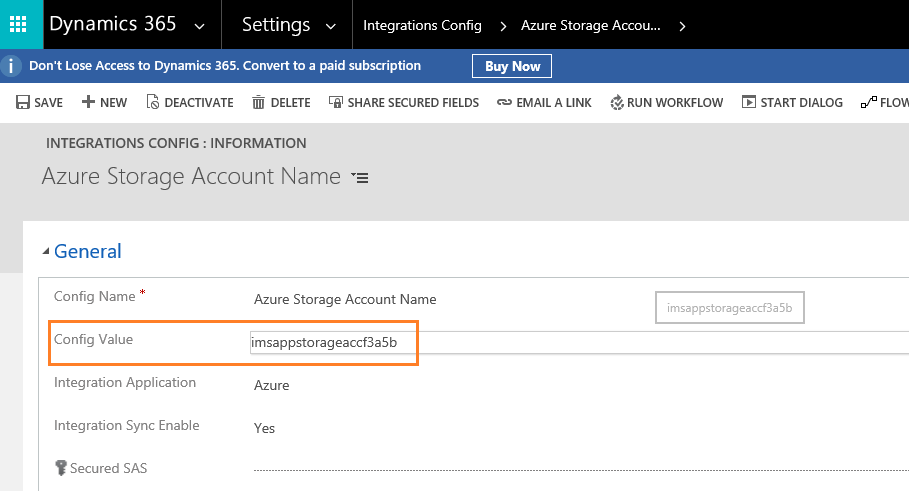


3) Azure Storage Account Name

* Navigate to Azure
* Open the Resource “imsappstorageacc<xxxxx>” as shown below
* Copy the storage account name as shown in below screenshot

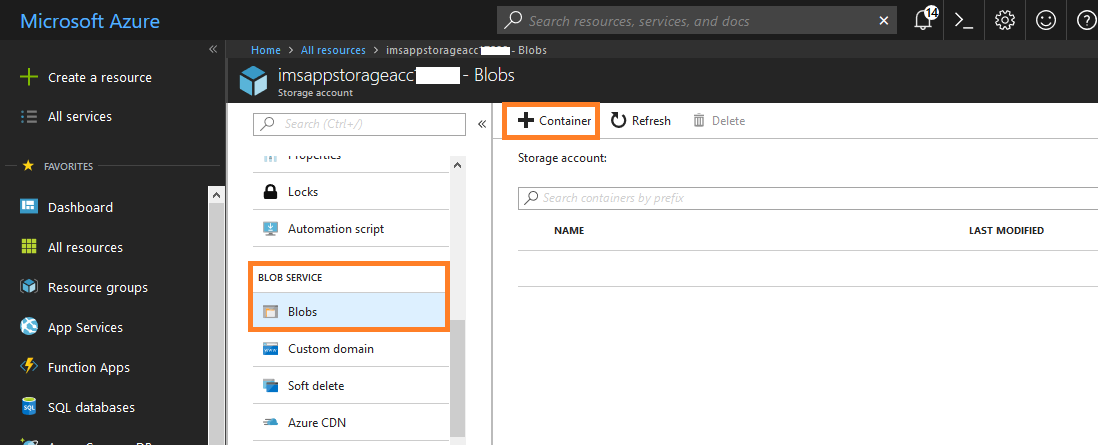


* Navigate to Settings -> Integrations Config Entity and Update the Account Name Value in **Azure Storage Account Name** Record “**config value”** field as shown below

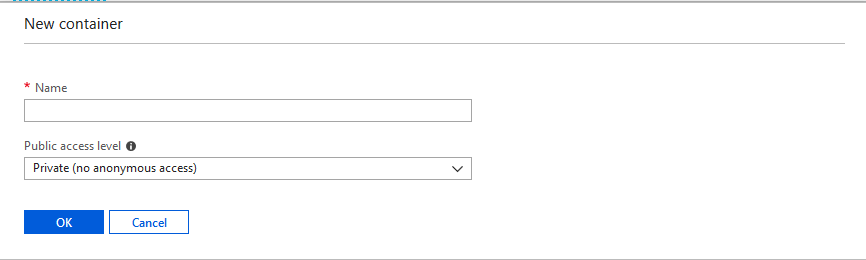


4) Azure Storage Container

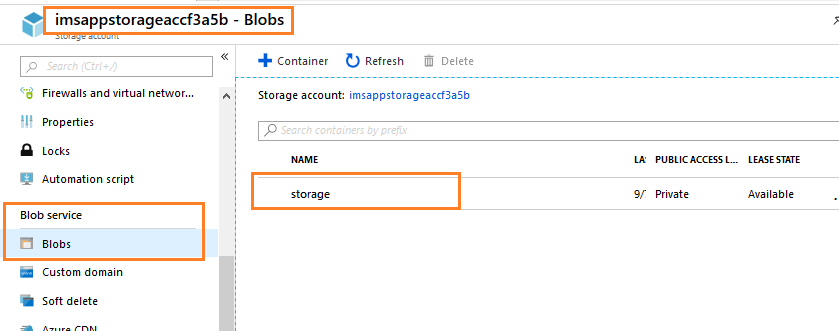
* Navigate to Azure
* Open the Resource “imsappstorageacc<xxxxx>” as shown below
* Go To Blobs in **Blob Service** and create the container as below



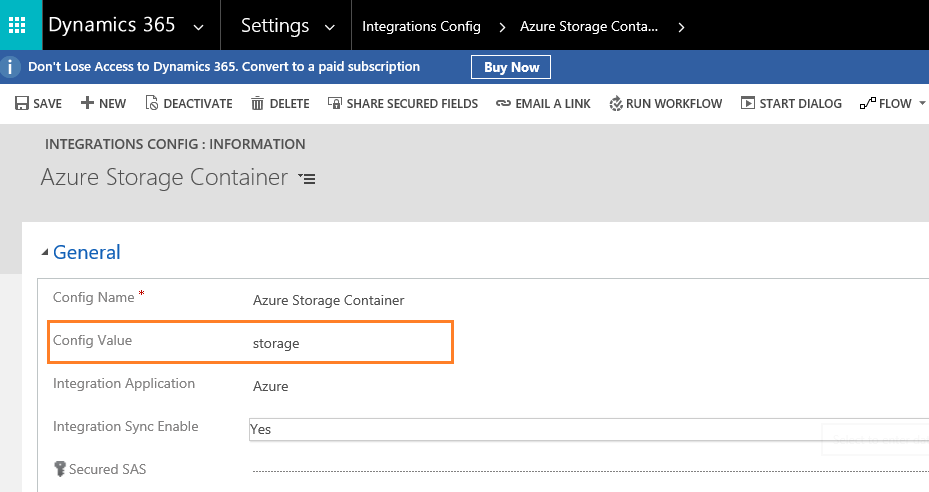
Provide name[user defined] and click on ok. name [user define] update on azure storage container “**config value”** on CRM



* Copy the Container Name

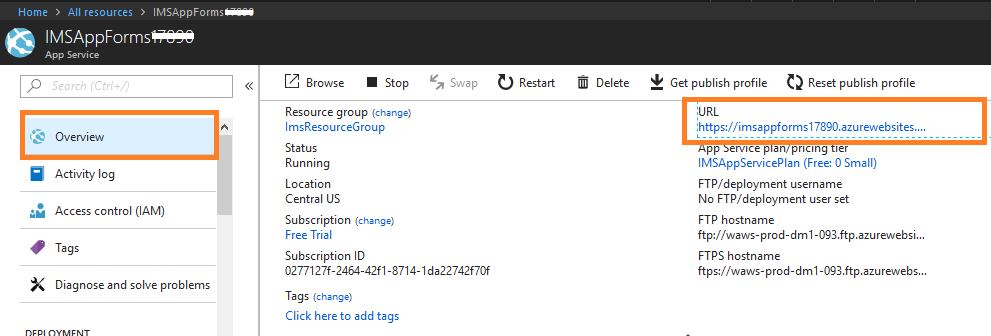


* Navigate to CRM -> settings -> Integrations Entity
* Update **Azure Storage Container** Record “**config value”** field as shown below



5) Forms Service API URL :

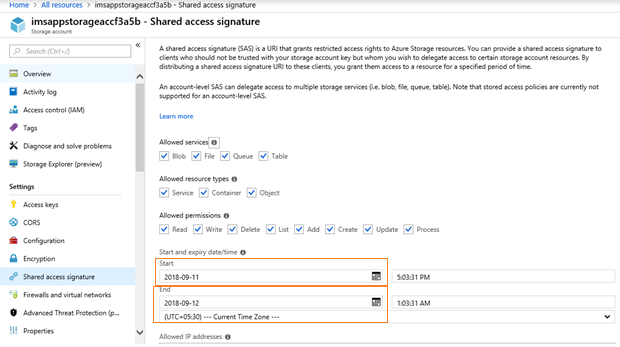
* Navigate to Azure
* Open the Resource “imsappforms<xxxxx>” as shown below
* On overview select the URL as shown in below screenshot, paste it on Forms Service API URL records “**config value”** field



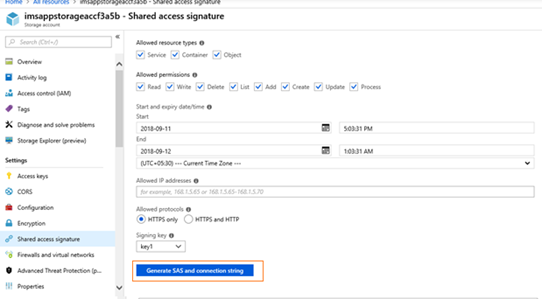
**6) Secured SAS Token**

Navigate to **imsappstorage<xxxxx>** resource and go to shared access signature under settings tab.

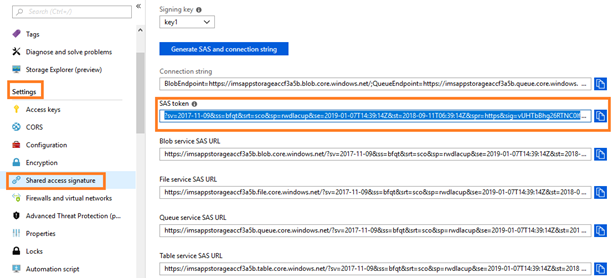
Please ensure that start date and end date fields are filled with valid dates and end date is greater than the current date.

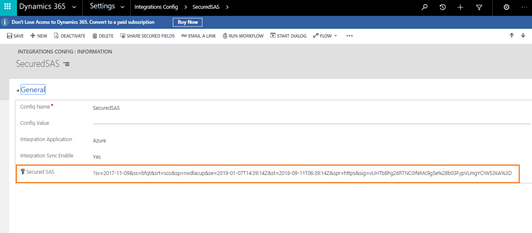


Click on Generate SAS and connection string.



Select the SAS token value as shown in below screenshots, paste it on “**Secured SAS**” field of Secured SAS record in Integrations config in CRM.





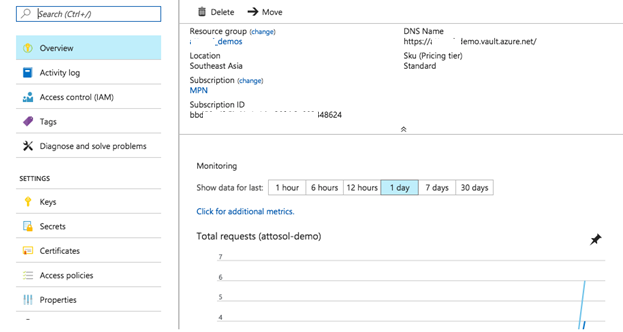
## 6.2 Azure Configurations

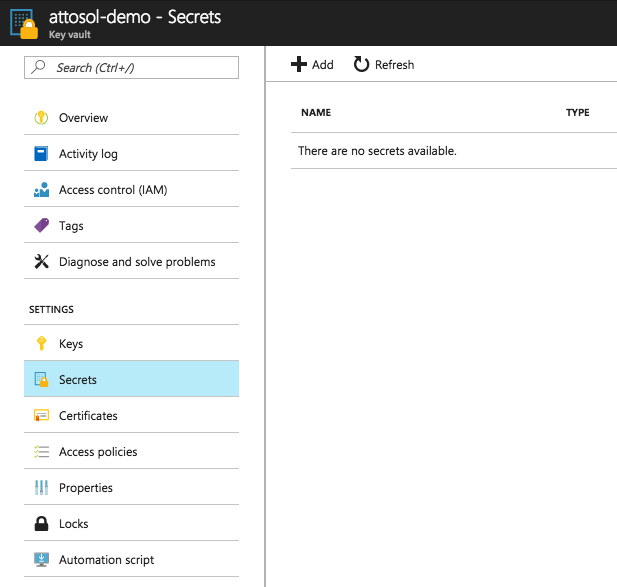
NOTE : Azure component names appear on document are same but on azure portal component names will end with your CRM Organization Unique Name (for ex:your CRM org name:org12345678 , on the Azure portal component names end with ComponentName12345 like : IMSAppPrintAll12345)

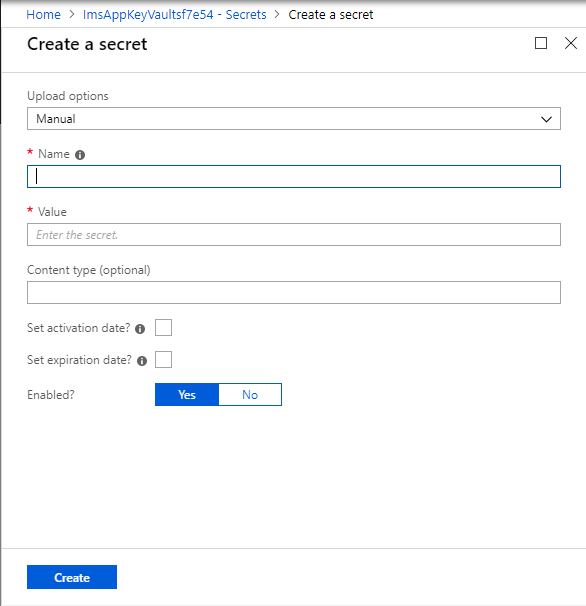
First step on azure configuration is creating key vaults. Post deployment, on azure portal you can find components named as **ImsAppKeyVaults** following with Unique org id on given azure portal under user provided resource group.

* **ImsAppKeyVaults:**

Open keyvault



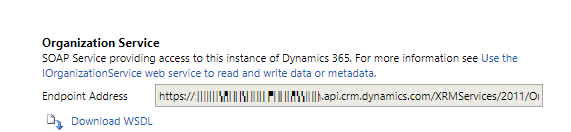
In the Azure Key Vault settings that you just created you will see a screen similar to the following. Click Secrets in the blade, followed by Add button on the top right.   


Type in your secret details:   


Secret details name should be as follows(yellow highlighted)

1. Name : BinderAdminUser Value :[provide binder application **Run as** username] ex:abc@ee.com
2. Name : BinderPassword Value :[provide binder application password]
3. Name : MIMSCRMOrgUrl Value :[Target CRM instance organization url]

On CRM : Settings 🡪 Customization 🡪 Developer Resource 🡪

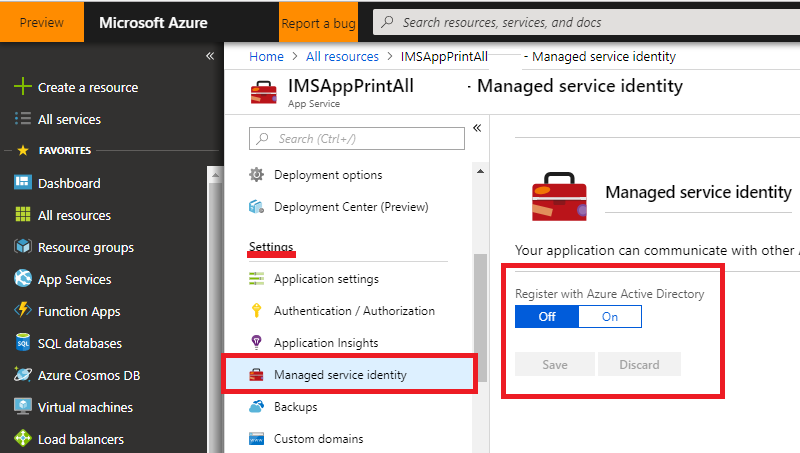


1. Name : PrintAllAdminUser Value :[provide printall application **Run as** username] ex:abc@ee.com
2. Name : PrintAllPassword Value :[provide printall application password]
3. Name : MIMSAppSourceAdmin Value :[provide web job applications **Run as** username]
4. Name : MIMSAppSourceCRMPwd Value :[provide web job applications password]

On part on Azure Configuration now we are going to configure for **IMSAppPrintAll** following with Unique org id **(**web app**)** and **IMSAppBinder** following with Unique org id **(**web app**) .** Below steps to create/modify.

* + **Managed Service Identity**  
    Goto Settings and select **Managed Service Identity.**

If is **ON** , don’t change anything. If it is **Off**, change it to **On**  and save. (After clicking on save it will take some time, wait till it save.)

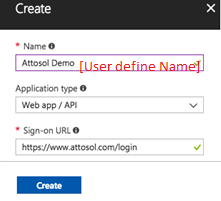


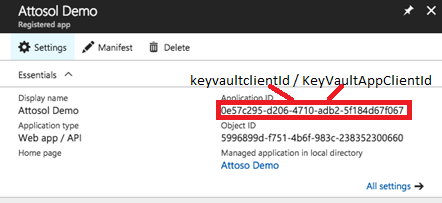
* + **Application Settings**

For application settings we need to create below parameters:

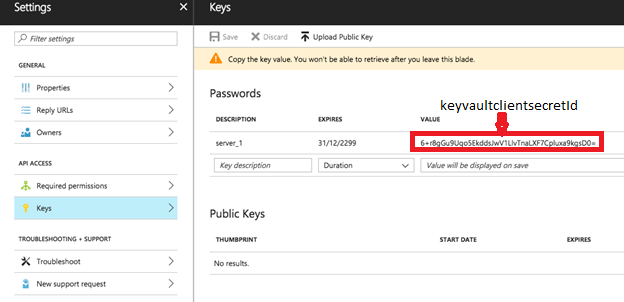
1. KeyVaultAppClientId :
2. keyvaultclientId
3. keyvaultclientsecretId

**Azure Portal > Azure Active Directory > App Registrations > New Application Registration**

ss

Note down your details. Remember, your client id is same as Application ID.   
 

Let's say you have a server where you intend to access the key from. You can use the server's hostname as the key description. If this server is compromised, you can revoke the access to AKV by simply deleting this key. That's neat!!!

While Generating the key for any app registration Provide some user defined name in the description and select “**Never** expires” as the duration and save the record. A key will be generated. Take this key as the keyvaultclientsecretId.  


1. keyvaulturl

Azure portal > ImsAppKeyVaults following with Unique org id > Overview > Take DNS Name



1. SecretUrlUserId

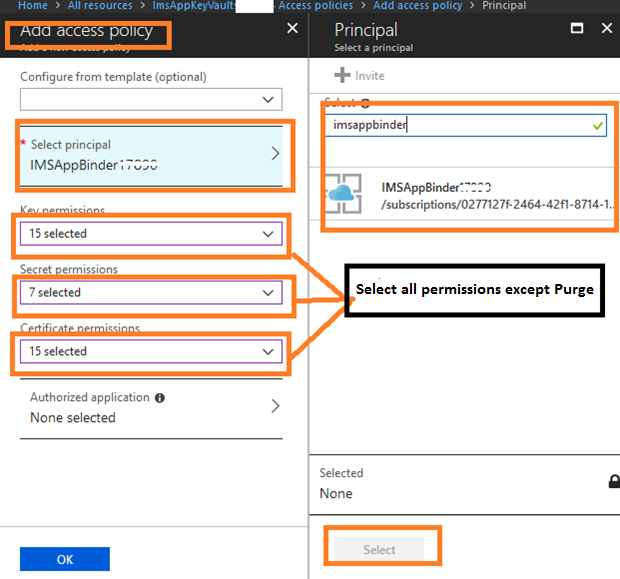
Azure portal > ImsAppKeyVaults following with Unique org id > Overview > Take DNS Name

Remove last charecter (/) from url

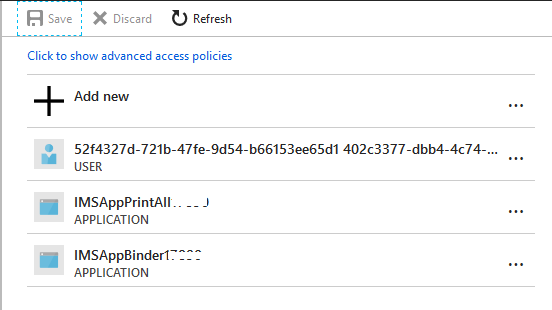
Now Go to Keyvault [ **ImsAppKeyVaults<xxxxx> ]**--> access policies --> click on add access policy and provide the application on select principal field

Select the application and provide the permissions except purge and click on ok.

Now save the access policies



Finally you will see the access policy like below



For Binder also we need to do same procedure

1) On and save managed service Identity

2) Azure active directory application registration binder

3) application settings – adding keyvalut url, application and secreat ids.

3) Adding application to access policy of keyvault..

CRM Configuration for Printall :

Settings 🡪 integration Configuration 🡪 search for “PrintAllApplicationURL“

For Config Value : Azure portal > IMSAppPrintAll following with Unique org id > Overview > URL

CRM Configuration for Binder :

Settings 🡪 integration Configuration 🡪 search for “PrintAllApplicationURL“

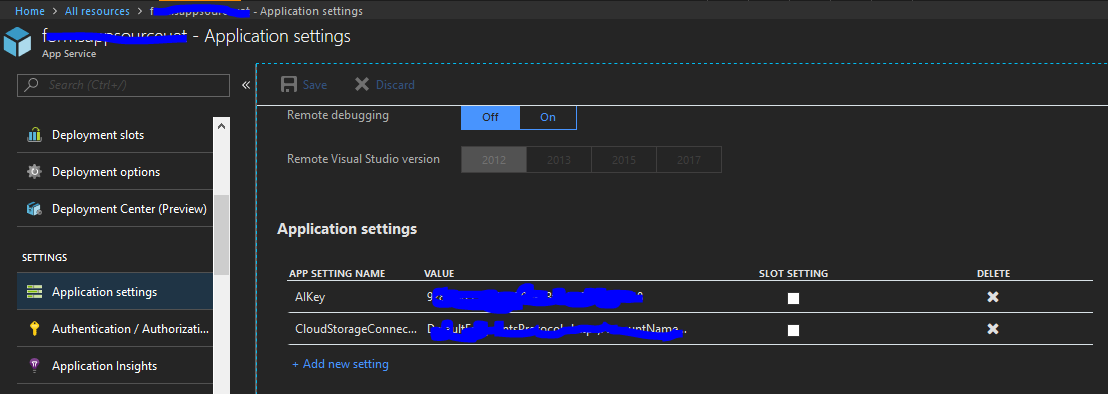
For Config Value : Azure portal > IMSAppBinder following with Unique org id > Overview > URL + /CreateBinderPrint.aspx?ID=

Ex: https://imsappbinder820f9.azurewebsites.net/CreateBinderPrint.aspx?ID=

**Form Configurations:**

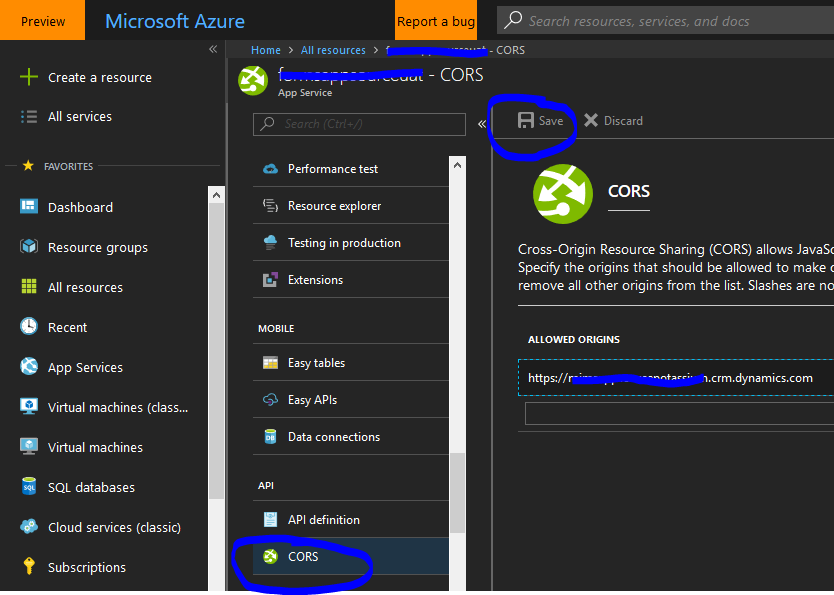
1. For Forms App (**IMSAppForms**) service need to add below items on app settings:
   1. AIKey : (navigation : application insights -- overview – Instrumentation key)
   2. CloudStorageConnectionString: (navigation : storageAccount – Access keys –Connectionstring)
   3. Save the Settings.

**Note:** Remove text “**;EndpointSuffix=core.windows.net**” form the CloudStorageConnectionString



Adding Application Settings on App service.

IMP Note: Target CRM instance URL must declare under Forms App (**IMSAppForms**) 🡪ss CORS



CRM side Form Management related configuration :

Settings 🡪 integration Configuration 🡪 search for “App Insights Instrumentation Key “

For the “App Insights Instrumentation Key” record we should provide “Config Value”

For Config Value :

Azure portal > IMSAppStorageAppInsight following with Unique org id > Overview > Instrumentation Key

Settings 🡪 integration Configuration 🡪 search for “Forms Service API URL“

For the “Forms Service API URL” record we should provide “Config Value”

For Config Value :

Azure portal > IMSAppForms following with Unique org id > overview > URL

After url add /api/v1/forms/ for Confiig Value

ex: <https://imsappforms82f09.azurewebsites.net/api/v1/forms/>

Update Master Forms Records :

Get all the Master Forms with “FormType” value Equals to “MasterForm” as shown below



To change the URL, follow below steps

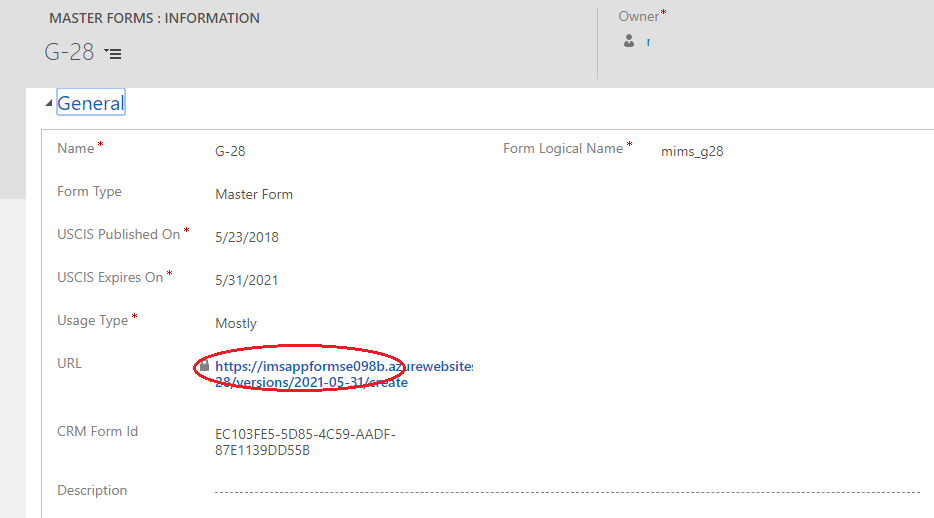
1) Take new Master Form and fill all field as it is except URL field

2) Save the form (Do Not Click “Save and Close” Button)

Follow the step 1 and 2 for all the Master Forms

Now able to see the URL with forms azure application name as below

Example : “G-28” Master Form

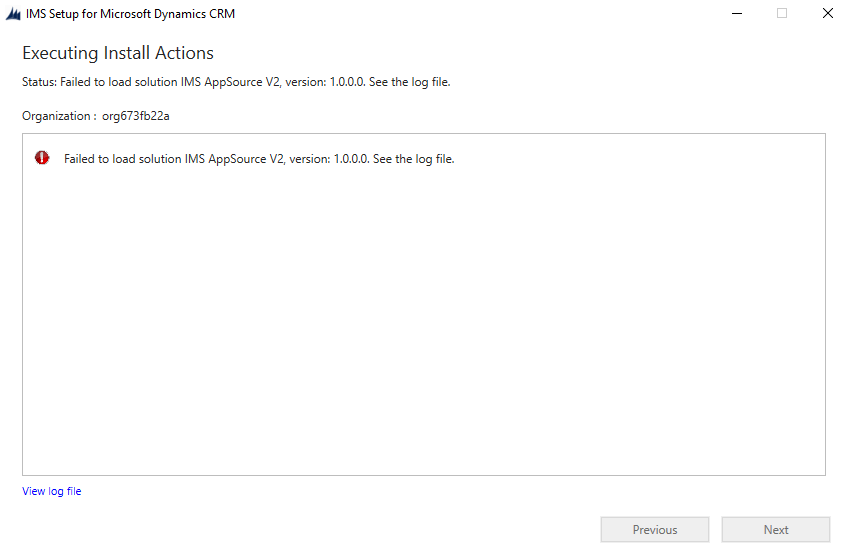


# 5.TroubleShooting Details

Follow below steps if any of the required components were not installed appropriately during deployment. Failures may occur if any of the pre-requisites not provided, network disconnections, Azure components take more time than anticipated to deploy on cloud or any other unexpected failures. Follow the below steps to resolve them manually if not completed automatically by Dynamics CRM OOB tool Package Deployer.

## 5.1 Solution Import Failure

Follow the below steps when the solution import gets failure through package deployer tool as shown below



* Navigate to CRM -> Settings -> Solutions
* Verify the “MIMSAppSourceV2” Solution is created or not (it may take some time to import solution)
* If you found “MIMSAppSourceV2” Solution try to rerun the package again.

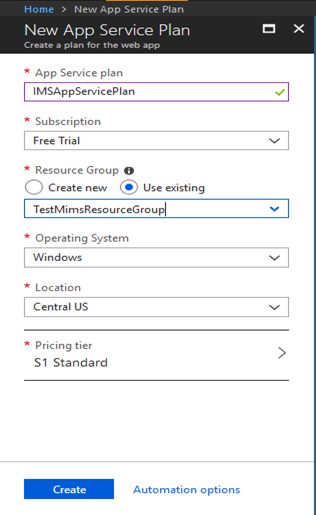
## 5.2 Azure Applications

## 5.2.1 Web Service Plan

If IMSAppServicePlan application is not created in azure as shown below



Create the AppService Plan with name of ‘IMSAppServicePlan’ as shown below

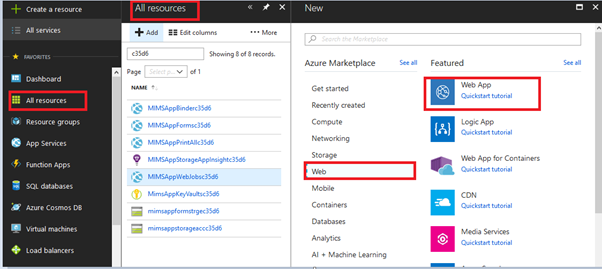


## 5.2.2 Web Jobs

If IMSAppWebJobs followed by CRM organization unique name application is not created in azure as shown below

Create new webapp as shown below

* Go To -> All Resources -> Web -> Web App

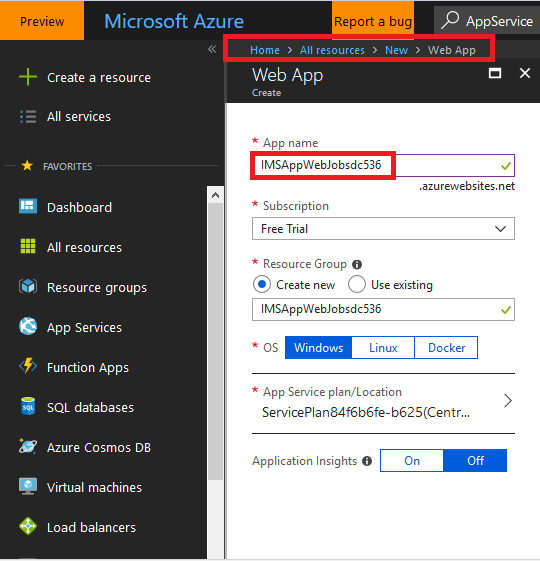


Create new application with name as below

IMSAppWebJobs followed by five characters of an organisation unique name as shown below

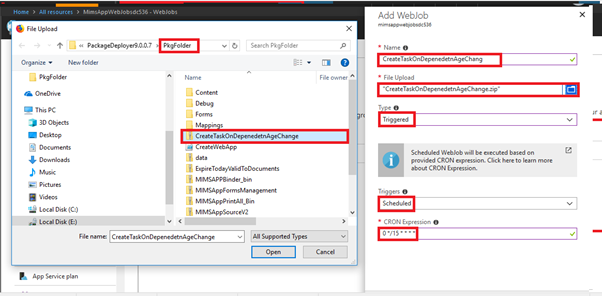
**Example**: Organisation unique name is “orgdc536fg”, and then the application name is like

“IMSAppWebJobsdc536” as shown below

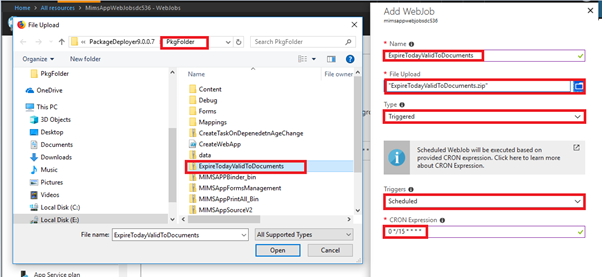


* Navigate to Web Jobs
* create the web jobs and upload the folder as shown below
* Cron Expression Must be always “0 \*/15 \* \* \* \*”
* All the web jobs are exists in PkgFolder

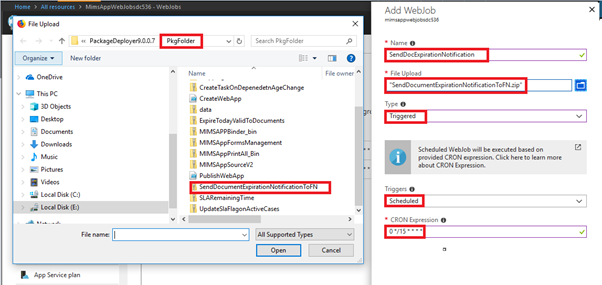
1. CreateTaskOnDepenedentAgeChang



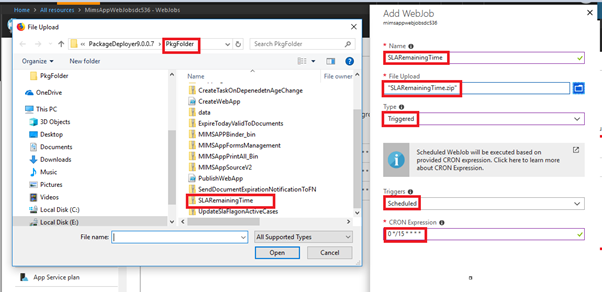
1. ExpireTodayValidToDocuments



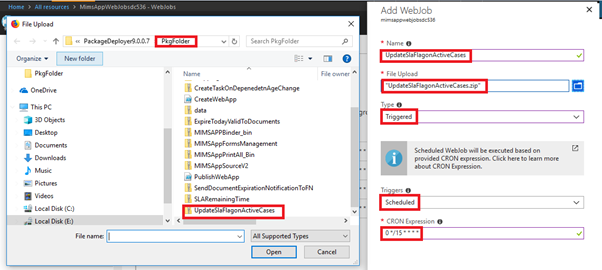
1. SendDocExpirationNotification



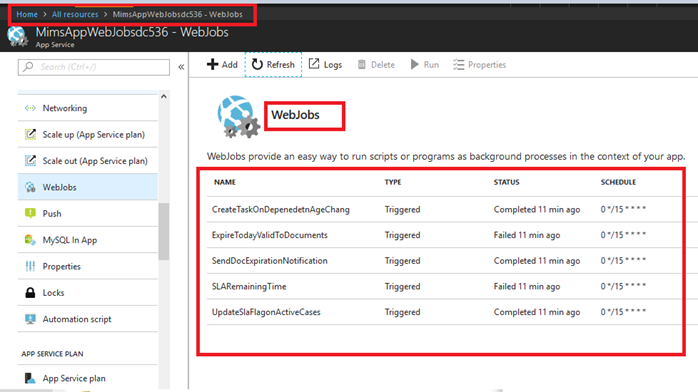
1. SLARemainingTime



1. UpdateSlaFlagonActiveCases



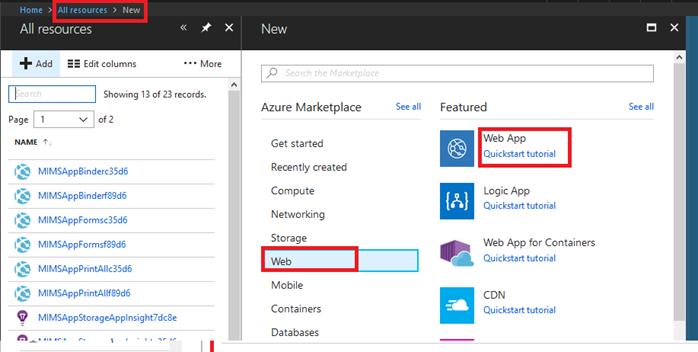
Web Jobs are created as below



## 5.2.3 Web Applications

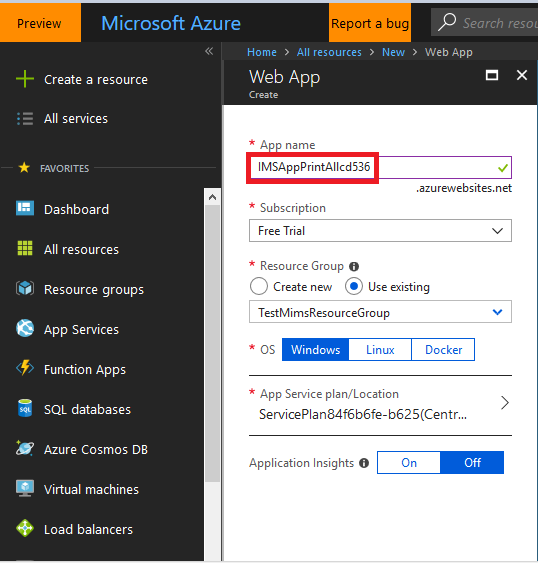
**IMSAppPrintAll Application**

* If IMSAppPrintAll followed by organization unique name application is not created in azure
* Create Webapp in azure as shown below



* Create “IMSAppPrintAll” followed by five characters of an organization unique name as shown below

**Example**: “IMSAppPrintAlldc536”



* Run the below PowerShell command as shown below

$subscriptionId = '' #Azure Subscription ID

$resourceGroupName = '' #Azure Resource Group Name

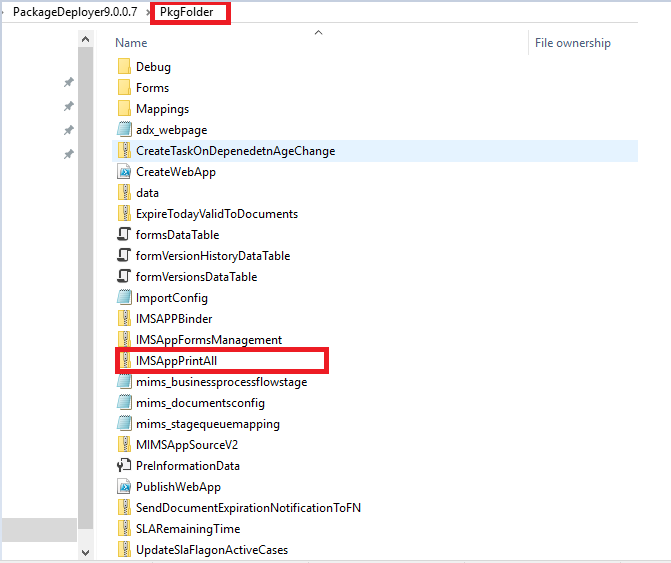
$IMSAppSourcePrintAll\_Path = '' # Print All Application Path from PkgFolder as shown below image

$IMSAppPrintAll ='' # Print All Application name, Example: 'IMSAppPrintAlldc536'

Add-AzureAccount

Select-AzureSubscription -SubscriptionId $subscriptionId

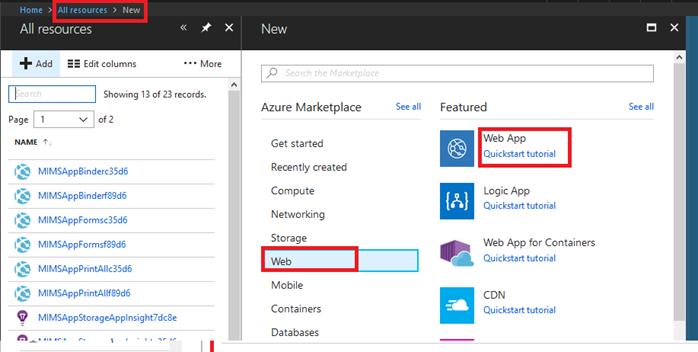
Publish-AzureWebsiteProject -name $IMSAppPrintAll –package $IMSAppSourcePrintAll\_Path



* Script has to be executed without getting error in PowerShell

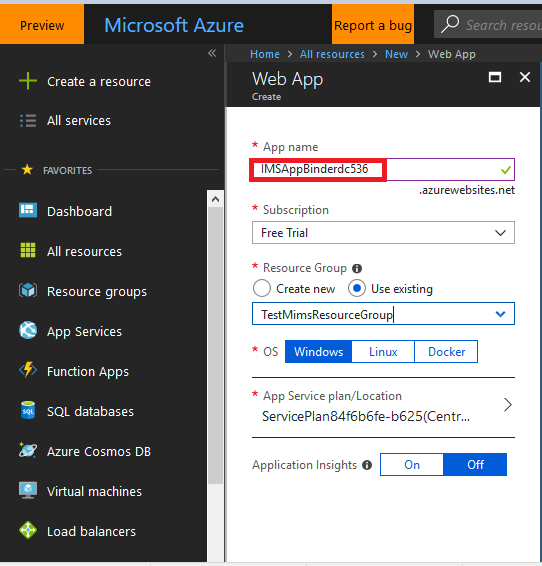
**IMSAppBinder Application**

* If IMSAppBinder followed by organization unique name application is not created in azure
* Create Webapp in azure as shown below



* Create “IMSAppBinder” followed by five characters of an organization unique name as shown below

**Example**: “IMSAppBinderdc536”



Run the below PowerShell command as shown below

$subscriptionId = '' #Azure Subscription ID

$resourceGroupName = '' #Azure Resource Group Name

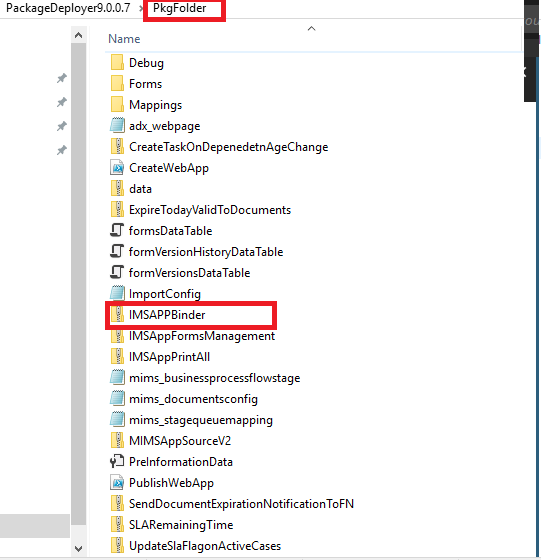
$IMSAppSourceBinder\_Path = '' # Binder Application Path from PkgFolder as shown below image

$IMSAppBinder ='' # Binder Application name, Example: 'IMSAppBinderdc536'

Add-AzureAccount

Select-AzureSubscription -SubscriptionId $subscriptionId

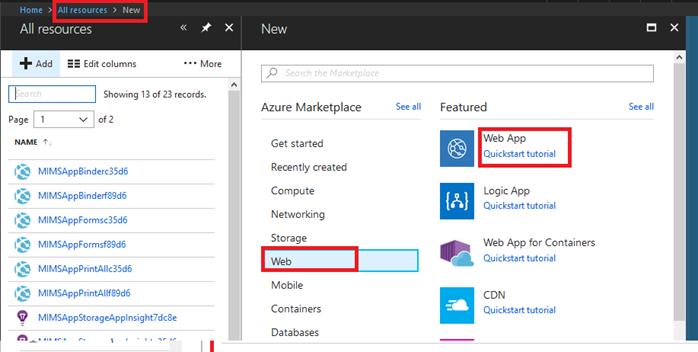
Publish-AzureWebsiteProject -name $IMSAppBinder –package $IMSAppSourceBinder\_Path



* Script has to be executed without getting error in PowerShell

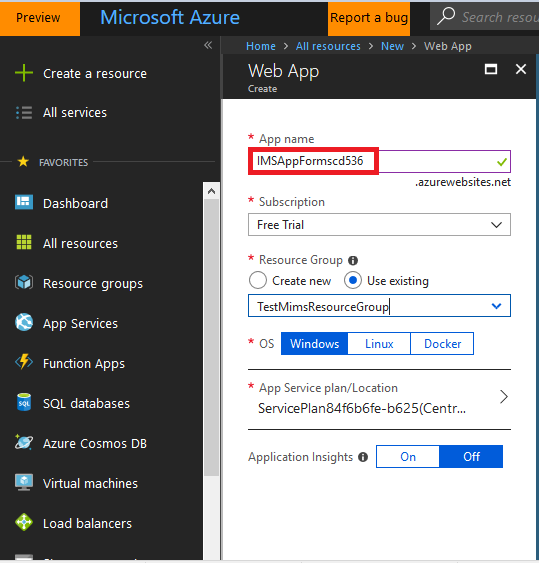
**IMSAppForms Application**

* If IMSAppForms followed by organization unique name application is not created in azure
* Create Webapp in azure as shown below



* Create “IMSAppForms” followed by five characters of an organization unique name as shown below

**Example**: “IMSAppFormsdc536”



Run the below PowerShell command as shown below

$subscriptionId = '' #Azure Subscription ID

$resourceGroupName = '' #Azure Resource Group Name

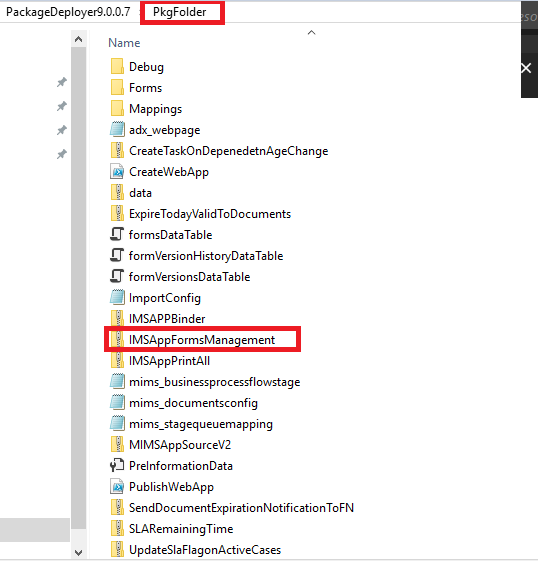
$IMSAppSourceForms\_Path = '' #Forms Application Path from PkgFolder as shown below image

$IMSAppForms ='' #Forms Application name, Example: 'IMSAppFormsdc536'

Add-AzureAccount

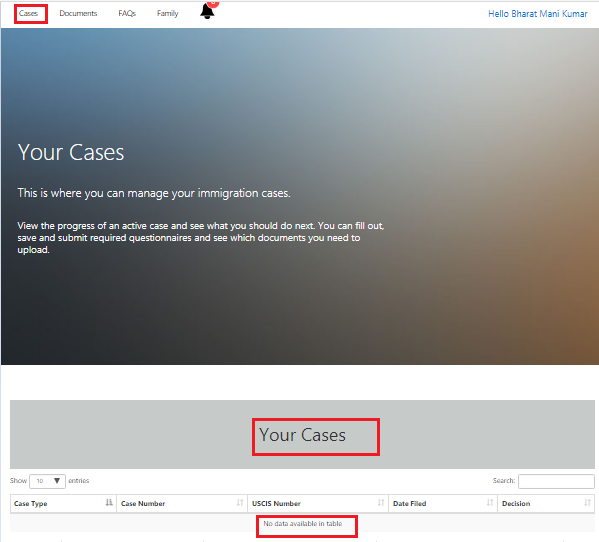
Select-AzureSubscription -SubscriptionId $subscriptionId

Publish-AzureWebsiteProject -name $IMSAppForms –package $IMSAppSourceForms\_Path



## 5.3 FN Portal

1. Not able to see the Foreign National Cases as shown below

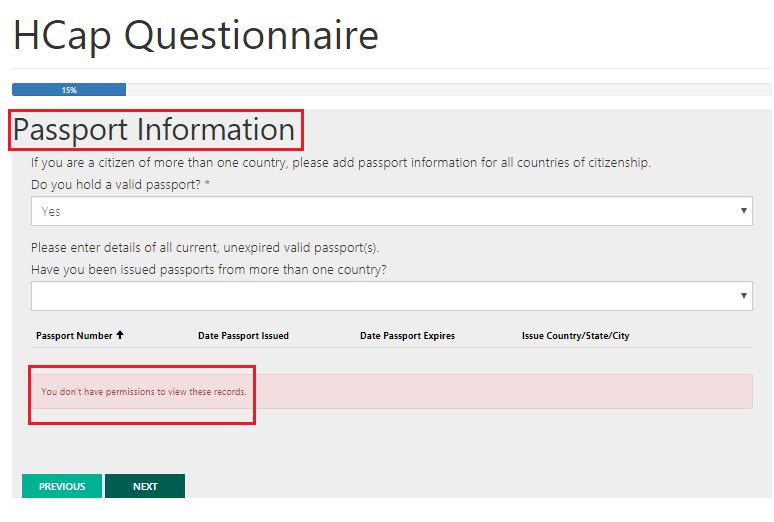


Resolution:

* Navigate to CRM -> Portals -> Web Roles -> Administrators -> Foreign Nationals
* Add the Foreign National to the administrator role
* Navigate to CRM -> Portals -> Web Roles -> Administrators -> Entity Permissions
* Add the "FN - Cases" Entity Permission to the administrator role
* Repeat all the above steps for the Documents, FAQ, Family and Tasks when you are not able to see the related entity records

## 5.4 Questionnaire Portal

1. Not able to create record through portal and getting the permissions issues as shown below



Resolution:

* Navigate to CRM -> Portals -> Entity Permissions
* Open “Birth & Passport Information” Entity Permission record
* Add the Administrator Web Role to the “Birth & Passport Information” Entity Permission record

# 7.Uninstalling Immigration AppSource System

When installing CRM Portal, multiple CRM solutions get imported.  Some of the solutions include Core portal features and others include specific functionality (i.e. Customer Portal or Employee Self Service).  When installing the solutions, the core solutions gets installed first and then the specific ones.  Once installed, you may then have done customizations to the portal solution.

To uninstall we need to follow the steps in reverse order:

1.      Undo customizations

2.      Uninstall specific solutions

3.      Uninstall core solutions

For example, for a Customer portal, uninstall below solutions first.

* **Undo Customizations**

1. Remove the portal related components from your solution
2. Remove the portal forms and views from the Custom App/ Sitemap that you created.
3. Remove all portal related customizations performed – remove portal related fields on your forms/ plugins/ JS/ Workflows/ actions etc.

* **Uninstall Specific Solutions in below order (if present):**

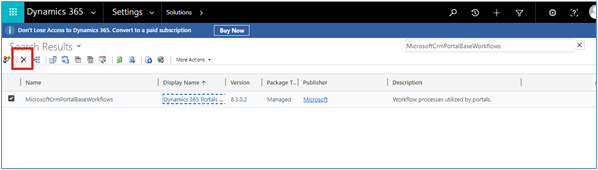
1. MicrosoftIdeasWorkflows
2. MicrosoftIdeas
3. MicrosoftForums
4. MicrosoftBlogs
5. Feedback
6. KnowledgeManagement
7. MicrosoftAzureStorage
8. MicrosoftBingMapsHelper
9. CustomerService
10. CustomerPortal

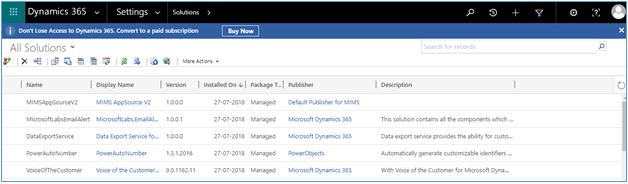
* **Uninstall the core solutions in this order:**

1. WebNotification
2. MicrosoftGetRecordIDWokrflowHelper
3. MicrosoftIdentity

[Special Note – This solution contains a Custom Workflow, which is used in Dialogs for Portal – for Change password and other workflows. It will show Dependencies when you delete, so manually removed the dialog steps to uninstall]

1. MicrosoftIdentityWorkflows
2. MicrosoftIdentitySystemWorkflows
3. MicrosoftCrmPortalBaseWorkflows
4. MicrosoftCrmPortalBaseSystemWorkflows
5. Portal TImeline
6. MicrosoftWebForms
7. MicrosoftCrmPortalBase
8. MicrosoftCrmPortalDependencies





End of Document