

Configure App Service Authentication for Your Azure Web App

Introduction

This lab is part of a series. This second lab will show you how to set up Azure App Service Authentication (aka. Easy Auth) for your Web App.

Terminology

The Portal uses a user interface concept that tends to expand horizontally towards the right. Every time that you choose something, rather than popping open a dialog box, it creates a new panel of in the user interface. These panels are called **blades**. I'll be referring to UI blades through this lab.

Variables

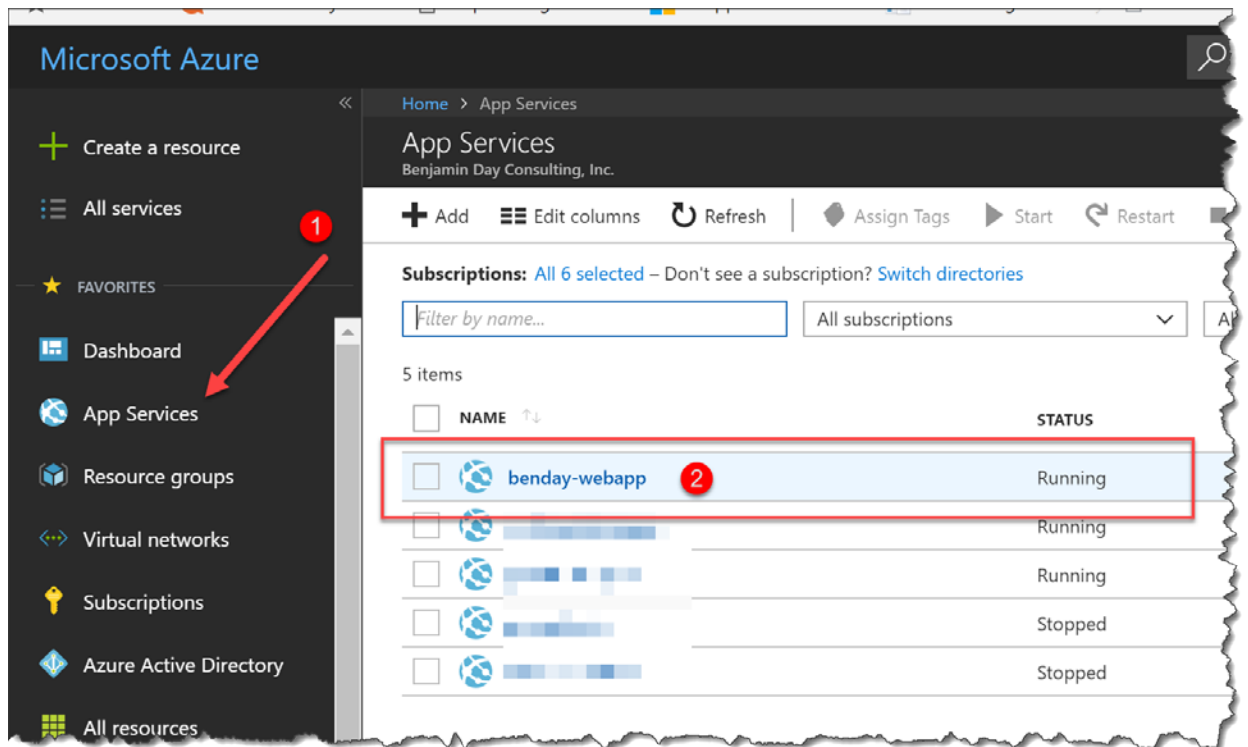
A lot of the resources that you create in this lab are going to need unique names. When I say unique, I mean that they're going to need to be unique for Azure and not just fun and creative. Since I can't possibly know which values that you're going to need to choose, I'm going to give you the list of these values now and let you choose them. I'll refer to these as "variables" throughout the lab and when I refer to them, I'll put them in squiggle brackets like this – {{Variable Name}}.

Variable Name	Description	Your Value
{{App Name}}	This is the name of your application in Azure. This will eventually turn into the URL for your application. For example, if my App Name is 'thingy123' application URL that azure generates will be https://thingy123.azurewebsites.net .	
{{Resource Group}}	This is the name of the Azure resource group.	
{{App Service URL}}	This is the URL for your web app. This value is generated for you by Azure.	https://{{App Name}}.azurewebsites.net
{{Client Id}}	This is the MSA application Id that's generated for you	
{{Client Secret}}	This is the MSA application secret that's generated for you	

Edit Your App Service

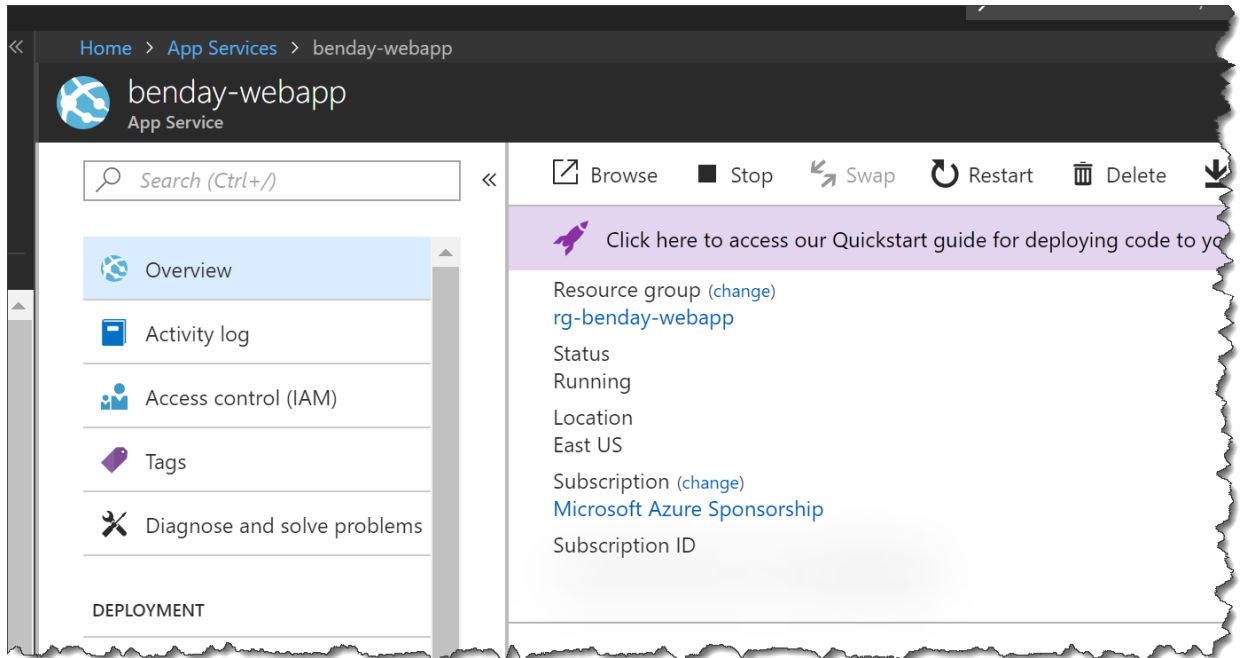
The first step is going to be to go to the Azure Portal (<https://portal.azure.com>). The Azure Portal is how you administer all your stuff in Azure.

1. Open a browser and navigate to <https://portal.azure.com>
2. In the left column of the of your Azure Portal window, click on **App Services**. (NOTE: if App Services isn't already in your Favorites, click on **All services** and search for App Services.)

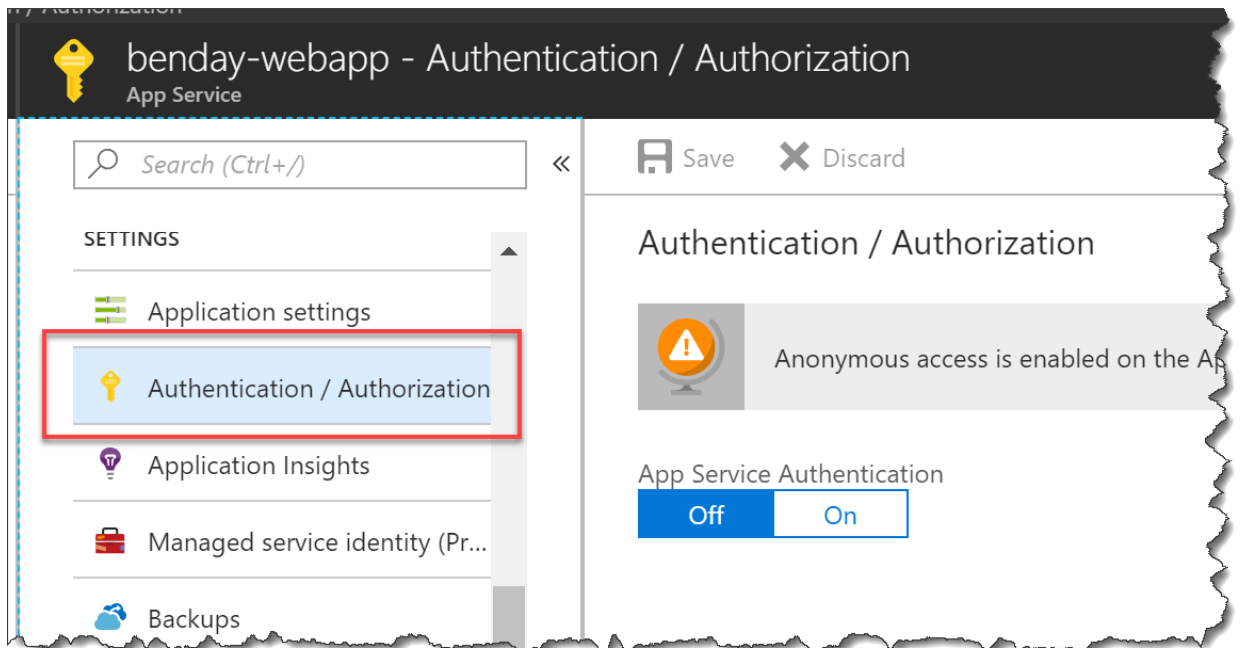


3. A new UI panel should be visible that has the title of App Services and you should see all the app services that are in your account. Locate the row for **{{App Name}}** and click that row to start editing it.

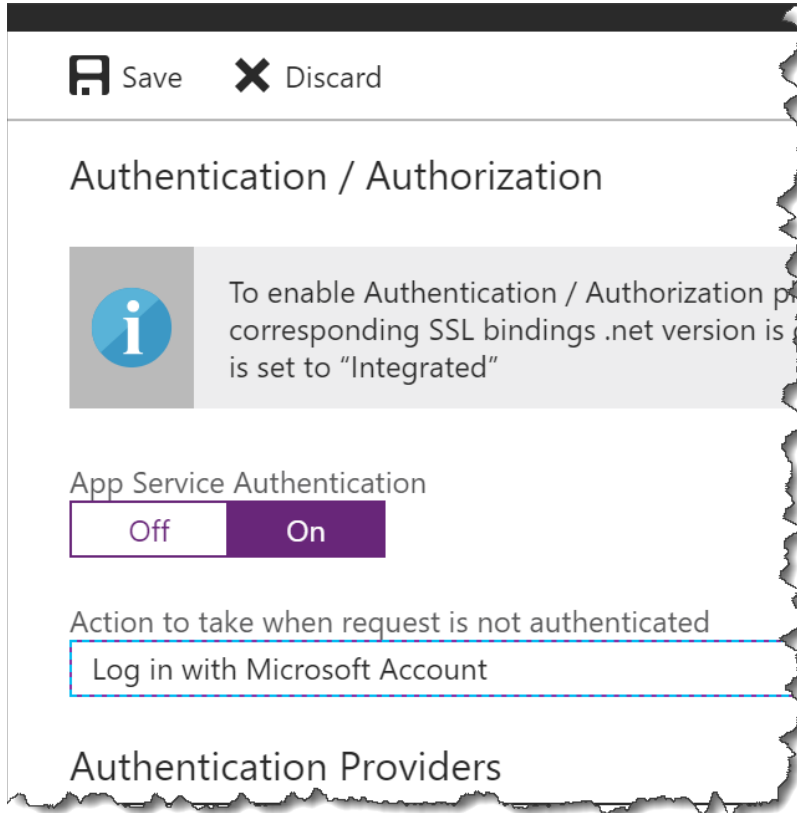
- You should now see the Overview page for your Azure App Service Web App and it should look something like the following image.



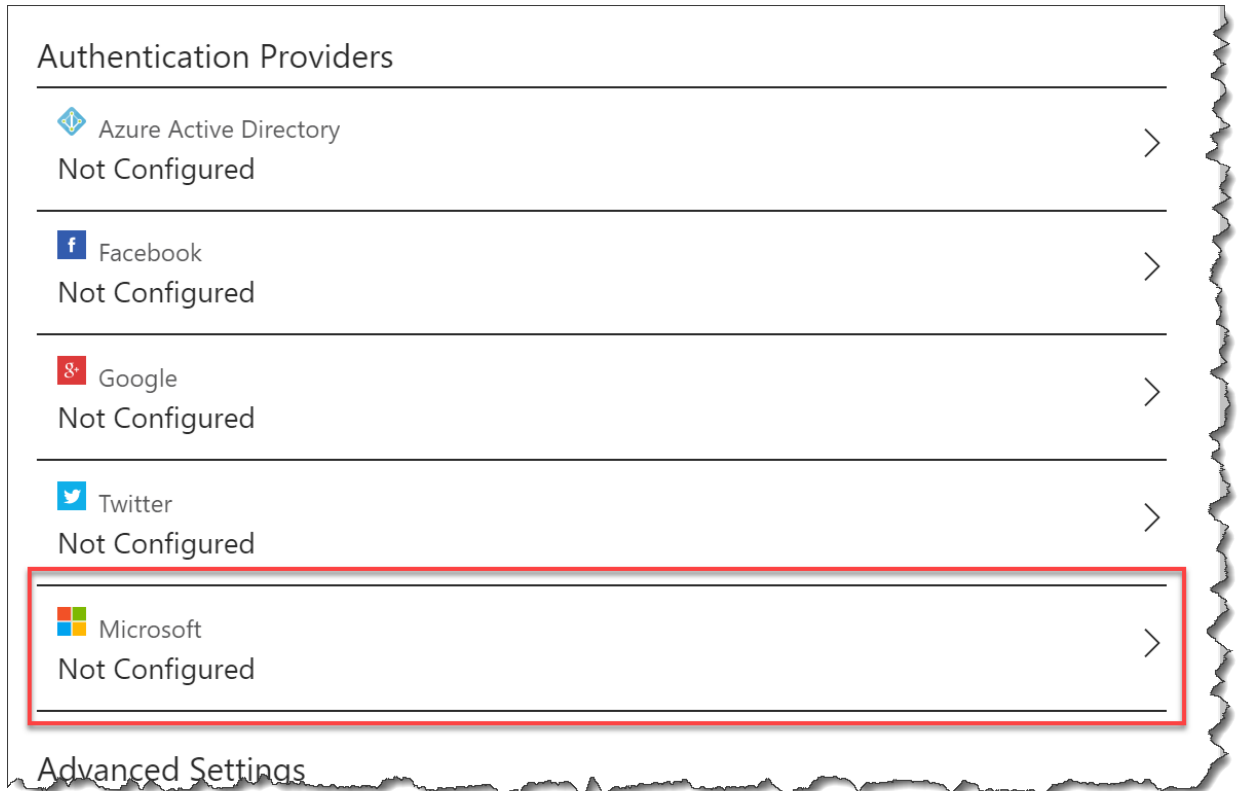
- In the left column of the App Service blade, you should see a menu bar with a bunch of options. At the moment, you're probably on Overview. Scroll down until you locate **Authentication / Authorization**. Click **Authentication / Authorization**. You will see that your web app is set to run in anonymous mode and App Service Authentication is turned off.



Under **App Service Authentication**, click **On**. From the **Action to take when request is not authenticated** drop down, choose **Log in with Microsoft Account**. (NOTE: if you want to use a different authentication provider, you can...but then you'll be on your own for a lot of the remaining steps in this lab.)

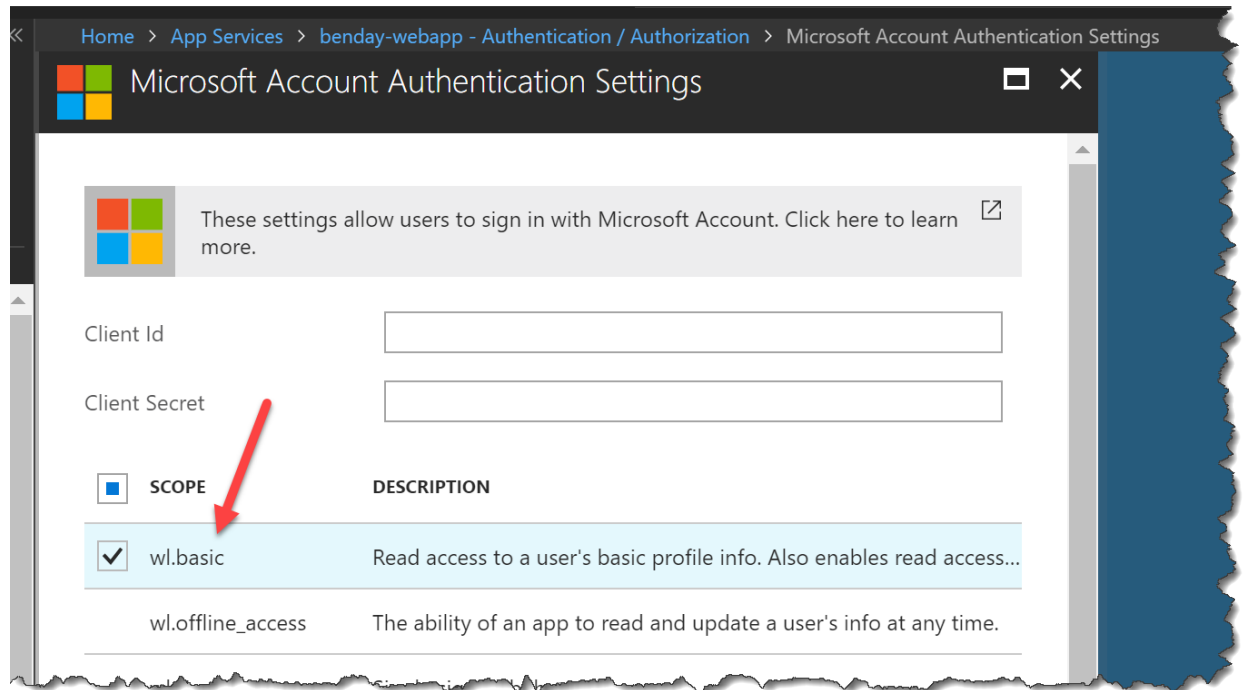


- Under **Authentication Providers**, locate the **Microsoft** account provider. At the moment, it should say **Not Configured**. Click the **Microsoft** account provider.



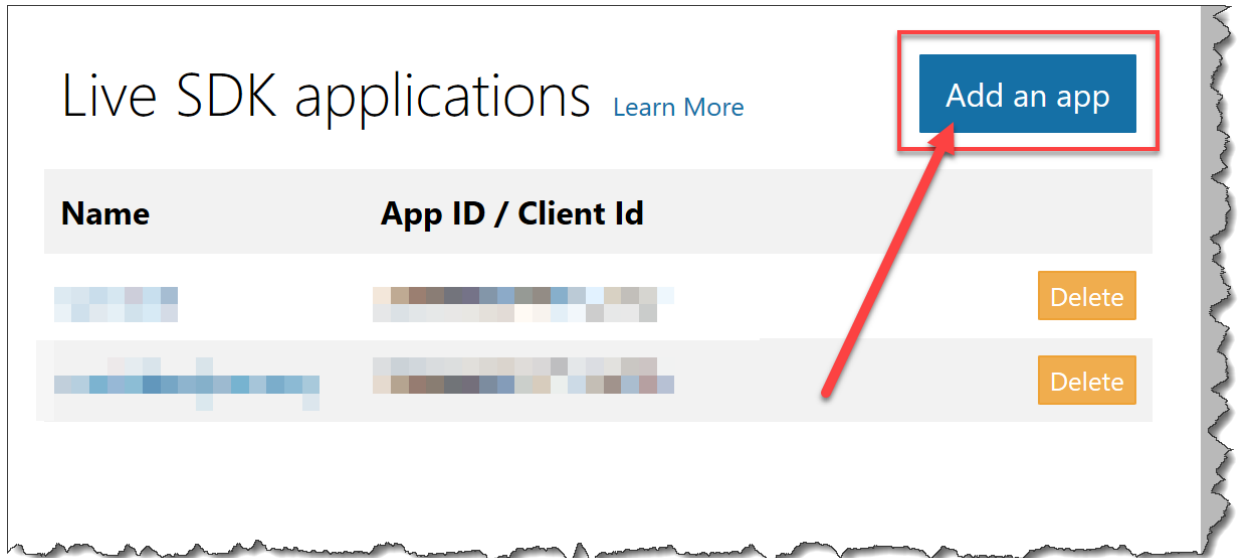
7. You should now be on the **Microsoft Account Authentication Settings** blade. This is how you make your application known to the Microsoft Account (MSA) framework. There's not a lot we can do on this page right now because we have to go to another website to get the values for **Client Id** and **Client Secret**.

BUT! While we're here, locate **wl.basic** and check its checkbox.

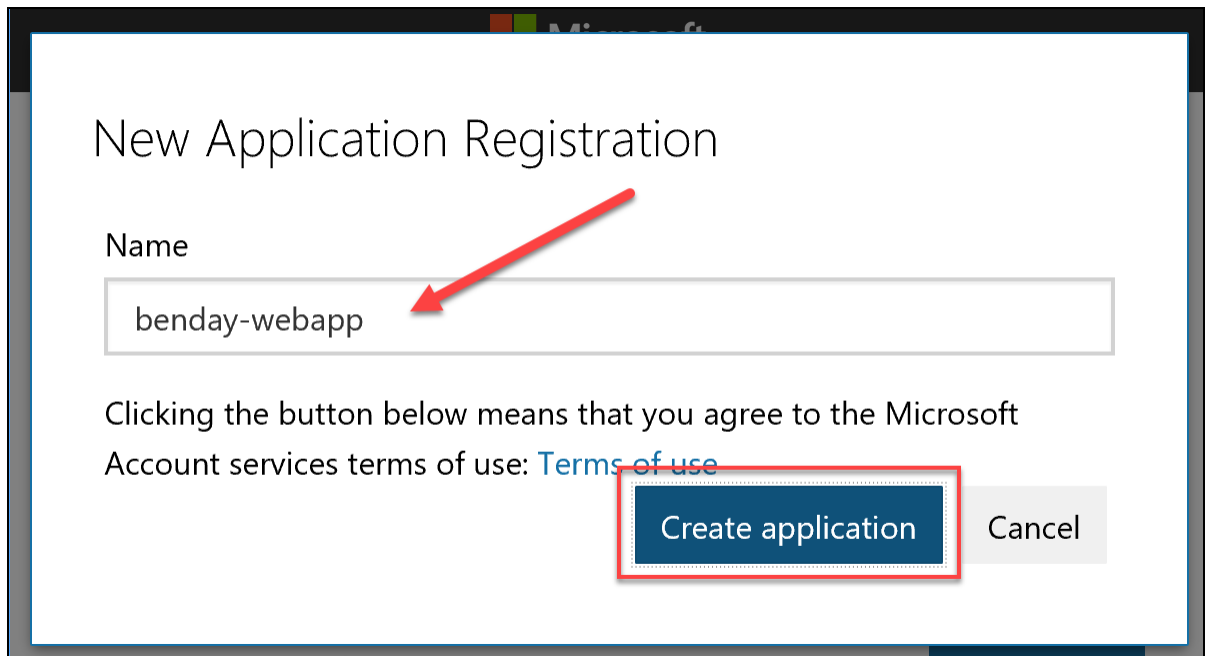


8. Open a new tab in your browser and navigate to <https://apps.dev.microsoft.com/#/appList>

- Under **Live SDK applications** click the **Add an app** button.

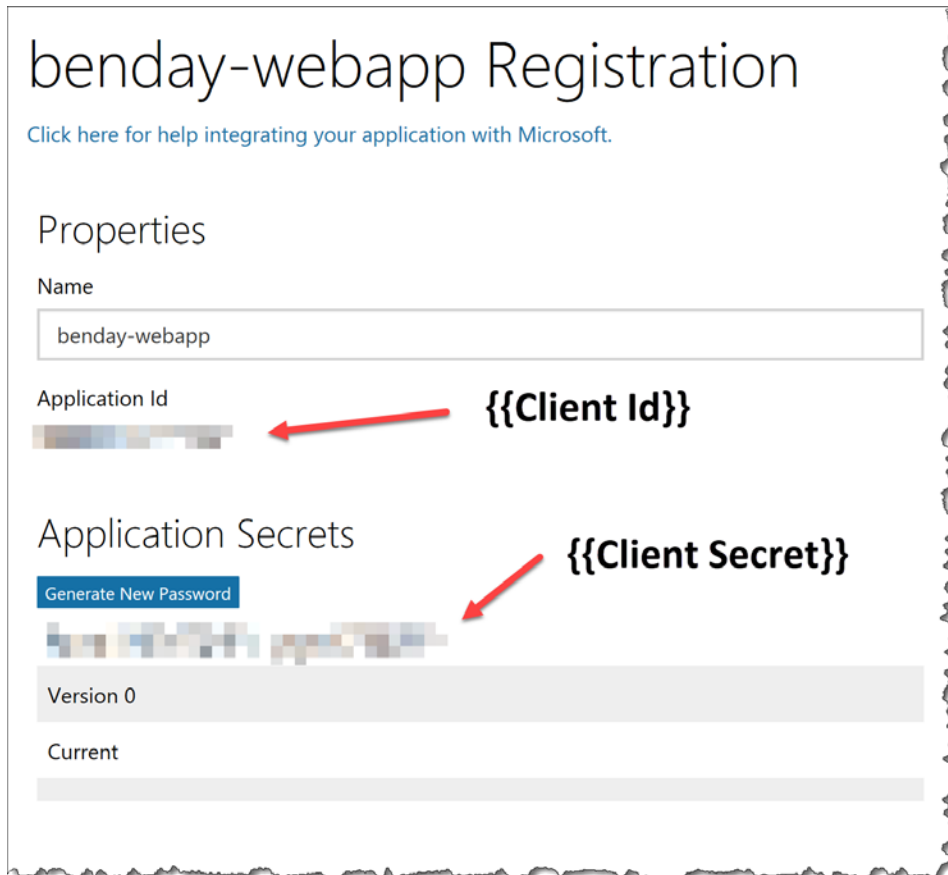


- You should see a **New Application Registration** dialog. In the **Name** box, enter your **{{App Name}}** value. Click the **Create application** button.

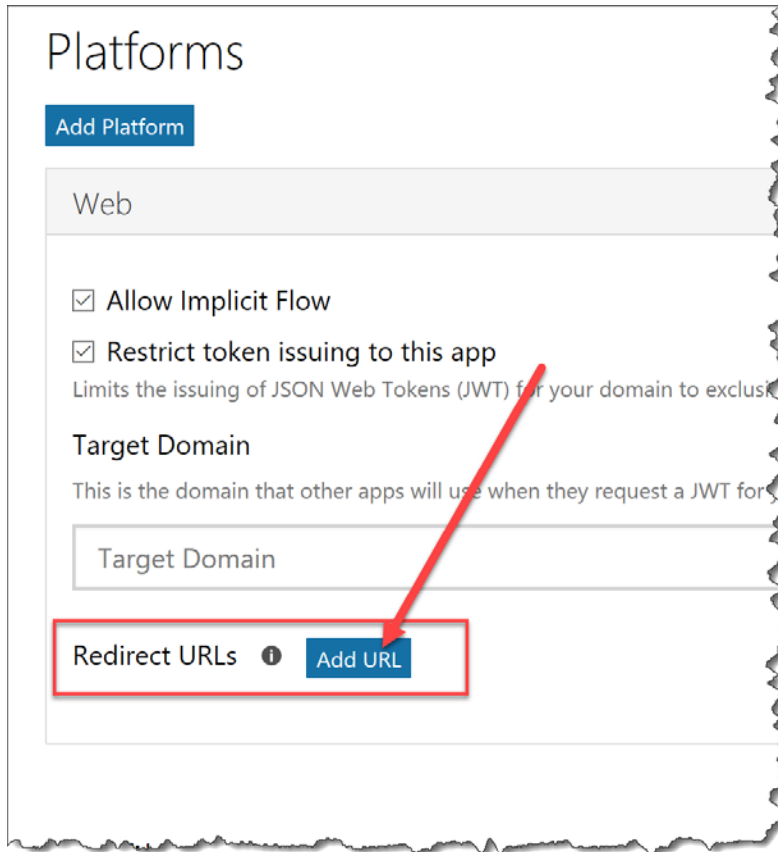


- 11. You should now see the registration information page for your MSA application. This page should be showing you two very important pieces of information. PLEASE TAKE NOTE OF THESE VALUES!!! Put them in notepad or something.

I know that this is a little bit confusing but I'm going to have you refer to *Application Id* as **{{Client Id}}** and the *application secret* as **{{Client Secret}}**.



12. Scroll down until you see the **Platforms** section of the application registration. Locate the **Redirect URLs** section and click the **Add URL** button.



Platforms

Add Platform

Web

Allow Implicit Flow

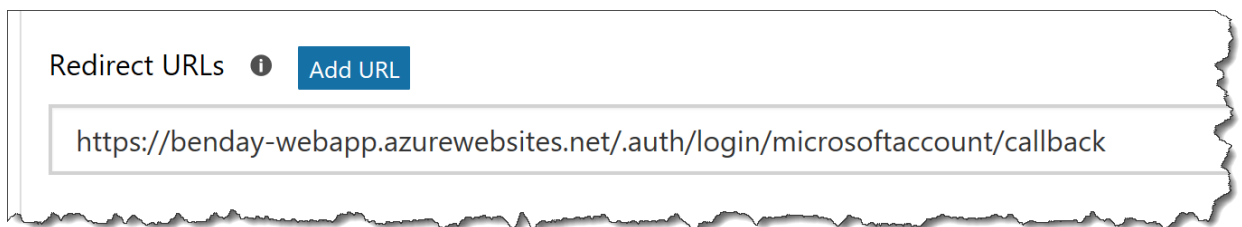
Restrict token issuing to this app
Limits the issuing of JSON Web Tokens (JWT) for your domain to exclus

Target Domain
This is the domain that other apps will use when they request a JWT for

Target Domain

Redirect URLs ⓘ Add URL

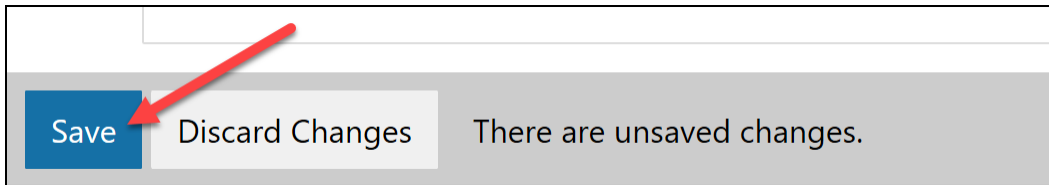
13. You should now have a box under Redirect URLs. Enter your **{{App Service URL}}** value in that box, followed by **/.auth/login/microsoftaccount/callback**.



Redirect URLs ⓘ Add URL

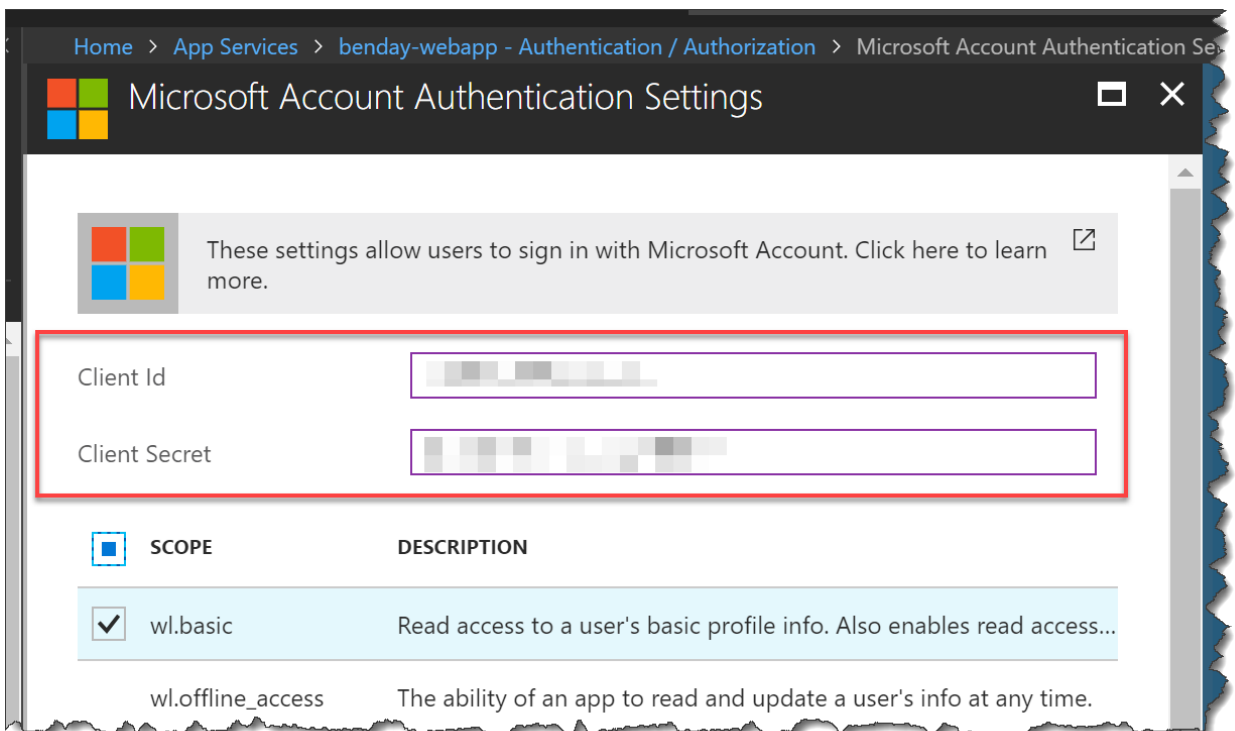
https://benday-webapp.azurewebsites.net/.auth/login/microsoftaccount/callback

14. At the bottom of the page, click the **Save** button.



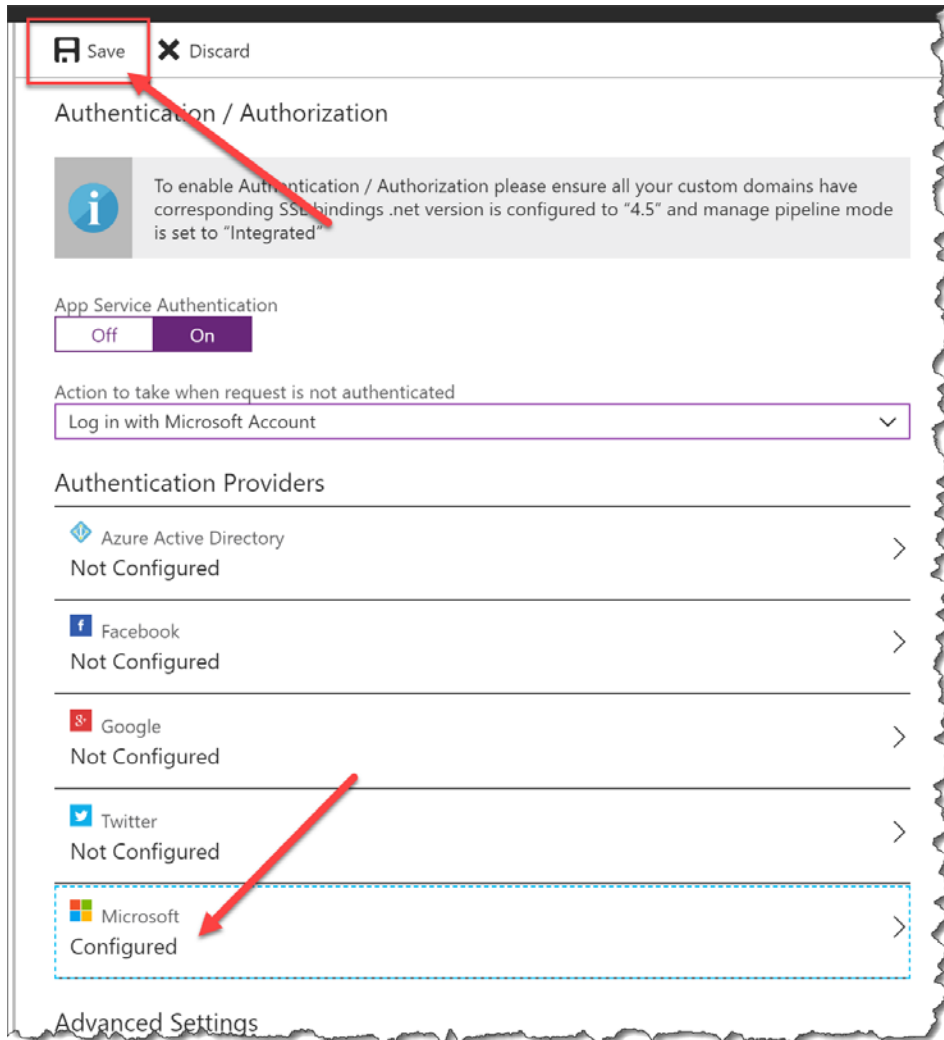
15. Go back to your Azure Portal browser tab. It should still be on the **Microsoft Account Authentication Settings** blade.

16. In the **Client Id** box, enter your **{{Client Id}}**. In the **Client Secret** box, enter your **{{Client Secret}}**.

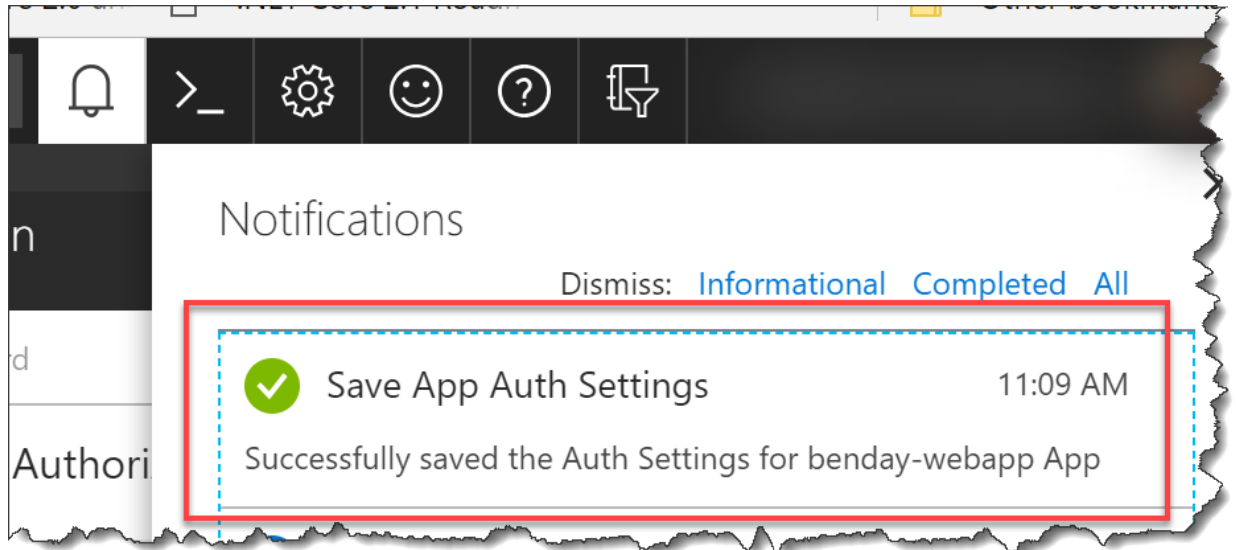


17. At the bottom of the **Microsoft Account Authentication Settings** blade, click the **OK** button.

18. You should now be back on the **Authentication / Authorization** blade. The Microsoft provider should now say **Configured**. Click the **Save** button to save your settings.



19. In your **Notifications**, you should see a notification that says **Save App Auth Settings** saved successfully.



You're done. You've just set up an Azure App Service Authentication identity provider for your Azure Web App. You're ready to use "Easy Auth".