Azure Prerequisites

v 1.0

[*02 May 2018*]

Confidential, Nubeva, Inc.

Author: Erik Freeland
Title: Dir of Customer Engineering
Email: erik@nubeva.com

Setup Azure Prerequisites

1. Before the nubevapoc environment is installed into the Azure environment, it is necessary to link the Controller with the AD accounts used in Azure as well as accept the EULA for the controller. This is a critical pre-requisite to the integration with the Azure authentication environment.

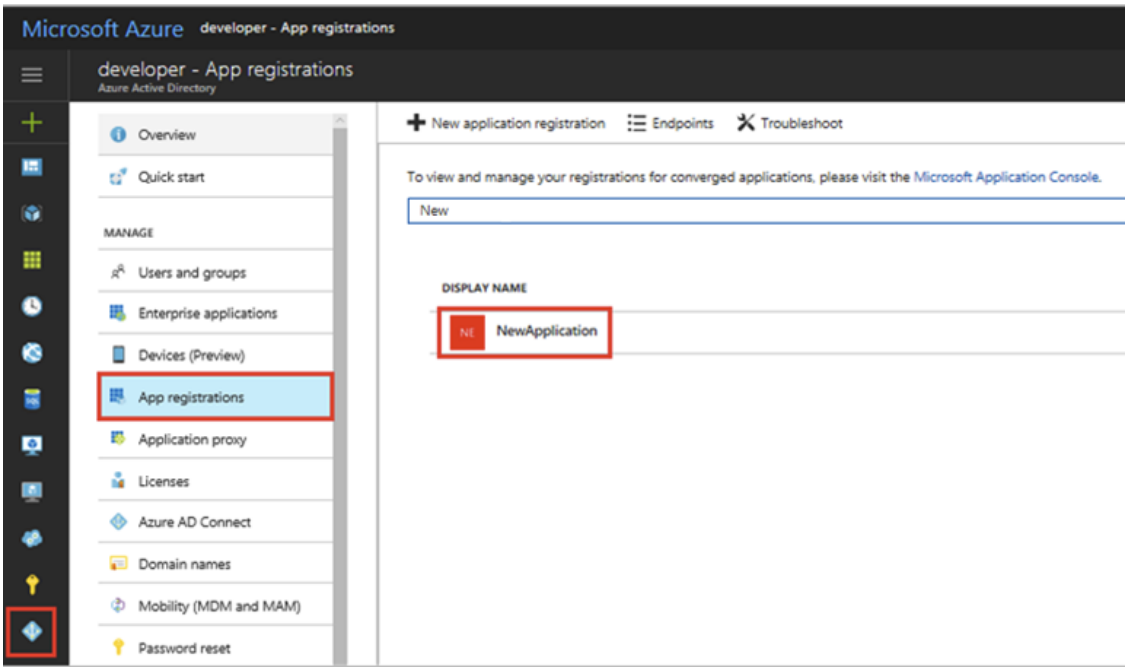
2. **Accept Nubeva Controller License.** Via the Azure GUI portal, connect to the following URL: https://portal.azure.com/#create/nubeva-inc.controller-templatebyol. Install this VM anywhere into the Azure subscription where you will be installing the POC environment. The setup/config details for the controller do NOT matter, choose the default if possible or just set an available value. Most importantly, accept the terms & conditions EULA. Once this VM deploys, simply delete it, the real controller will be deployed via the scripts. NOTE: This is the same URL you can use to deploy the controller into any environment once you are familiar with the Nubeva solution.

3. **Create a service principal**. This can be completed in several ways. One method using the Azure Portal GUI can be found below in Appendix A.

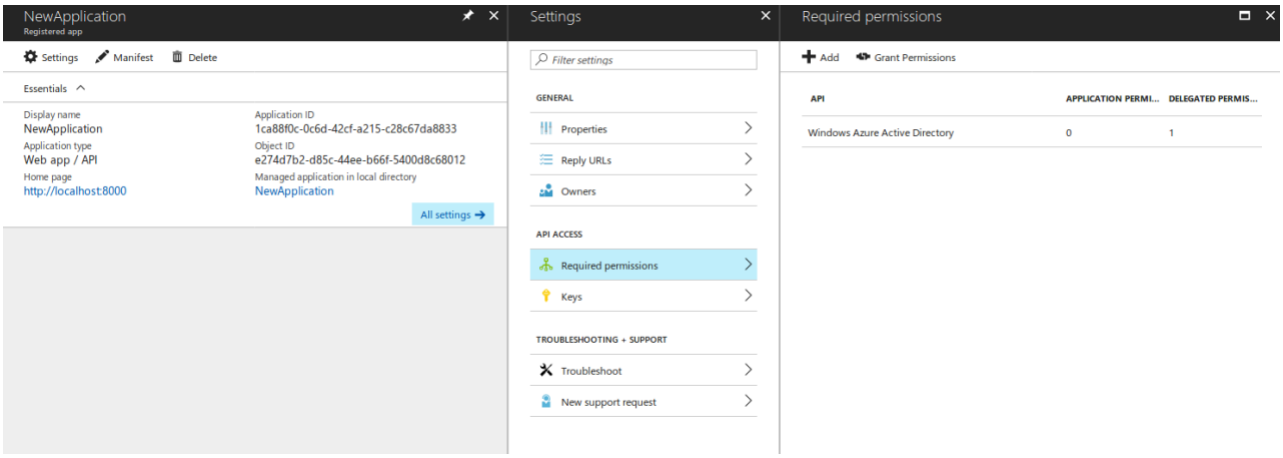
    a. A service principal can also be created with the following CLI command:
    `az ad sp create-for-rbac`

4. **Adding Permissions**. Nubeva's product uses Microsoft's OAuth authentication system in order to authenticate and authorize users. In order to access the authentication system, you must grant your application certain permissions. This process should be done through Azure Portal.

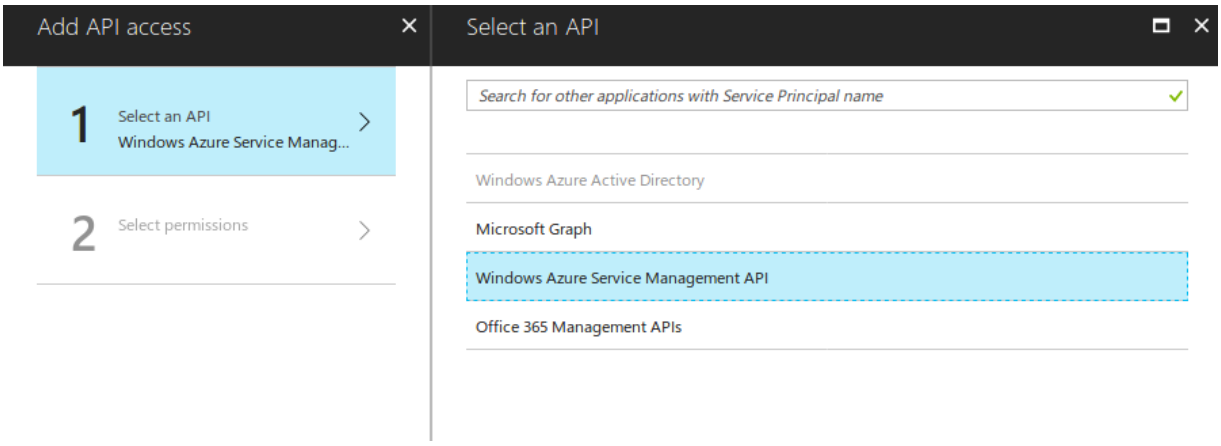
    a. In the left-hand navigation pane, click **Azure Active Directory** > **App registrations** > then click on the application that you want to configure.



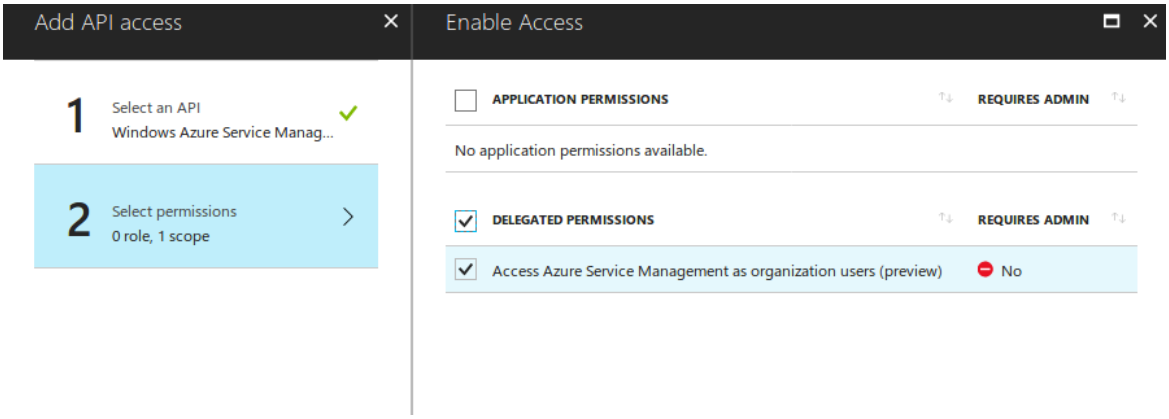
b. In the **Settings** blade, click on **Required permissions** under "API ACCESS"



c. Click **Add** and select **Windows Azure Service Management API**. Then press **Select** at the bottom of the blade.



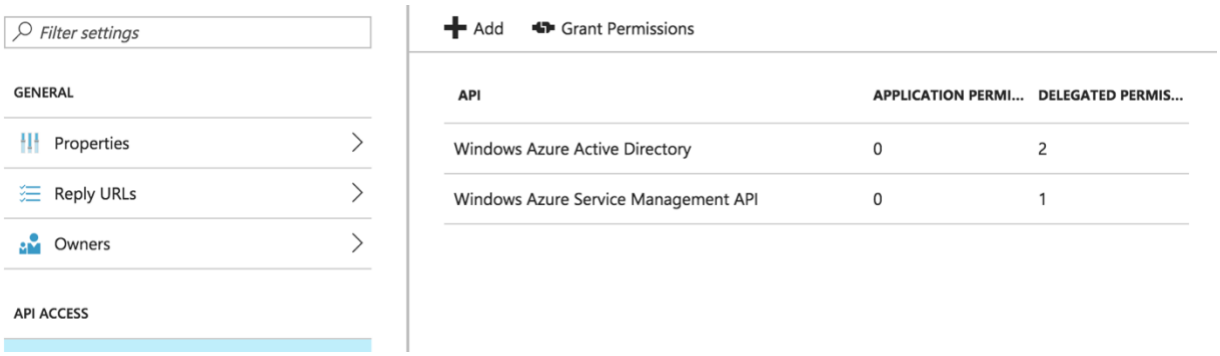
d. Select **Access Azure Service Management as organization users (preview)** under "DELEGATED PERMISSIONS". Then press **Select** and **Done** once finished.



e. You will be brought back to the application's **Settings** and **Required permissions** blades. Under **Required Permissions**, Add **Windows Azure Active Directory** and select the following under "DELEGATED PERMISSIONS":
   - **Read all users' basic profiles**
   - **Sign in and read user profile**

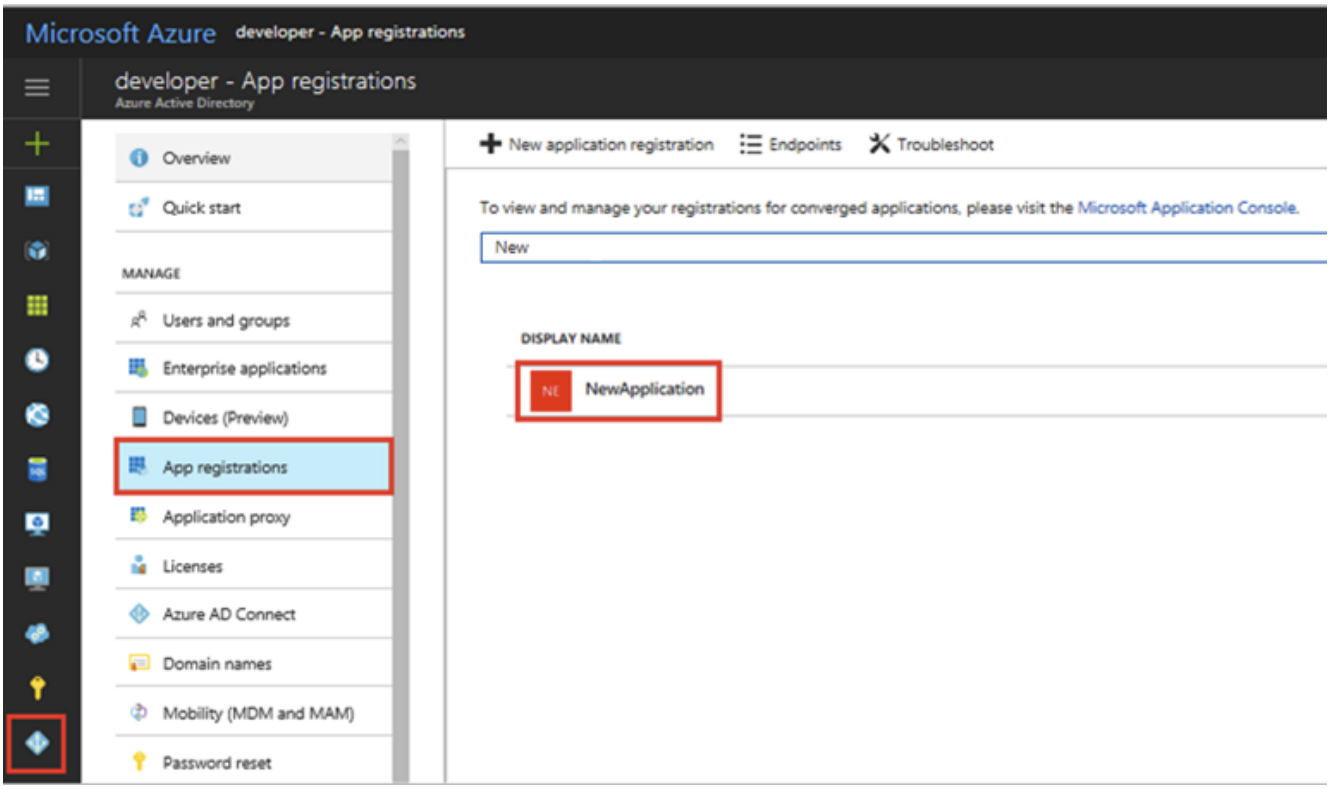
	DELEGATED PERMISSIONS	REQUIRES ADMIN
	Access the directory as the signed-in user	No
	Read directory data	Yes
	Read and write directory data	Yes
	Read and write all groups	Yes
	Read all groups	Yes
	Read all users' full profiles	Yes
✓	Read all users' basic profiles	No
✓	Sign in and read user profile	No
	Read hidden memberships	Yes

f. Click **Save** at the top left of the blade. In the end, the **Required permissions** blade should look like the following:

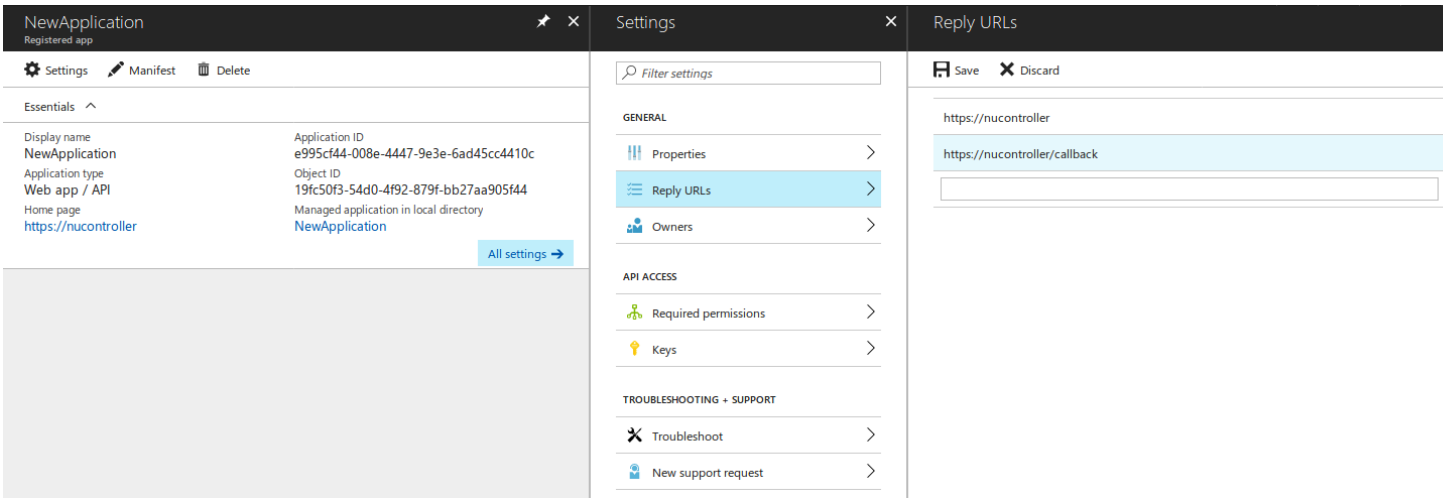


5. **Updating Reply URL**. The Azure Active Directory Reply URL must be updated so that an authentication token can be received after login.

    a. In the left-hand navigation pane, click **Azure Active Directory** > **App registrations** > then click on the application that you want to configure.



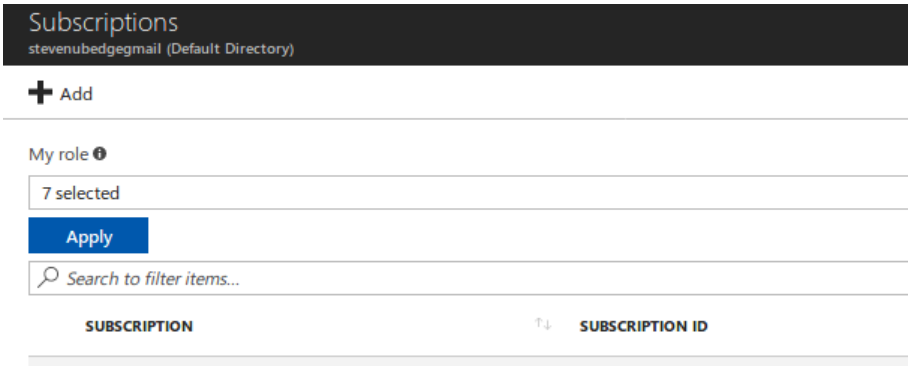
b. In the **Settings** blade, click on **Reply URLs** under "GENERAL", and add the following reply url: https://nucontroller/callback.



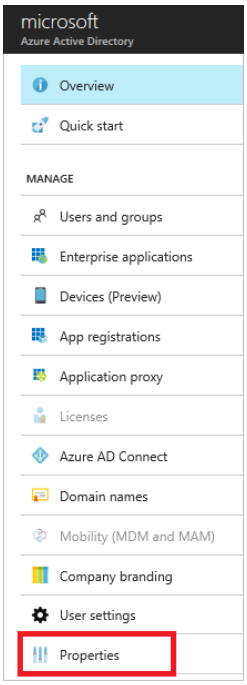
c. Click **Save**.

6. **Retrieving Credentials**. Nubeva's Controller requires the Subscription ID, Tenant ID, Password, and Application ID. All of this information can be found in the output of the CLI command above in Step 4a plus the command: `az account show`. The instructions for obtaining this information via the Azure Portal is detailed below.

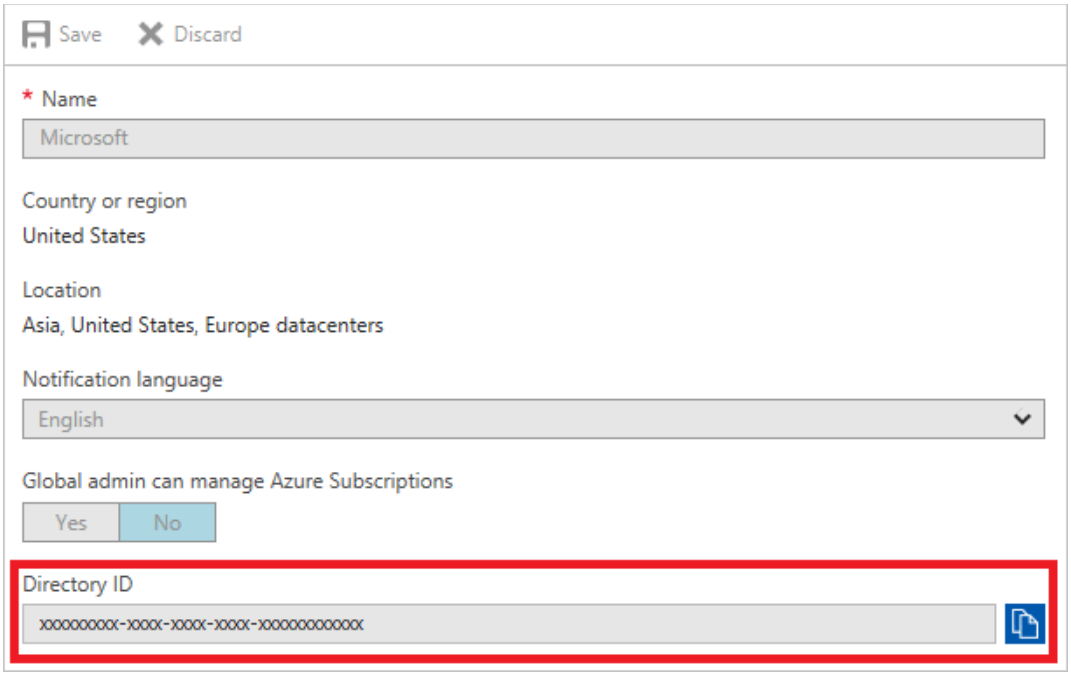
    a. To retrieve the Subscription ID, click on **Subscriptions** on the left hand navigation pane of the portal. Then copy the **Subscription ID** that you have linked your application to.



7. The Tenant ID is under Directory ID. Go to **Azure Active Directory** on the left hand navigation pane of the portal. Then click **Properties**.

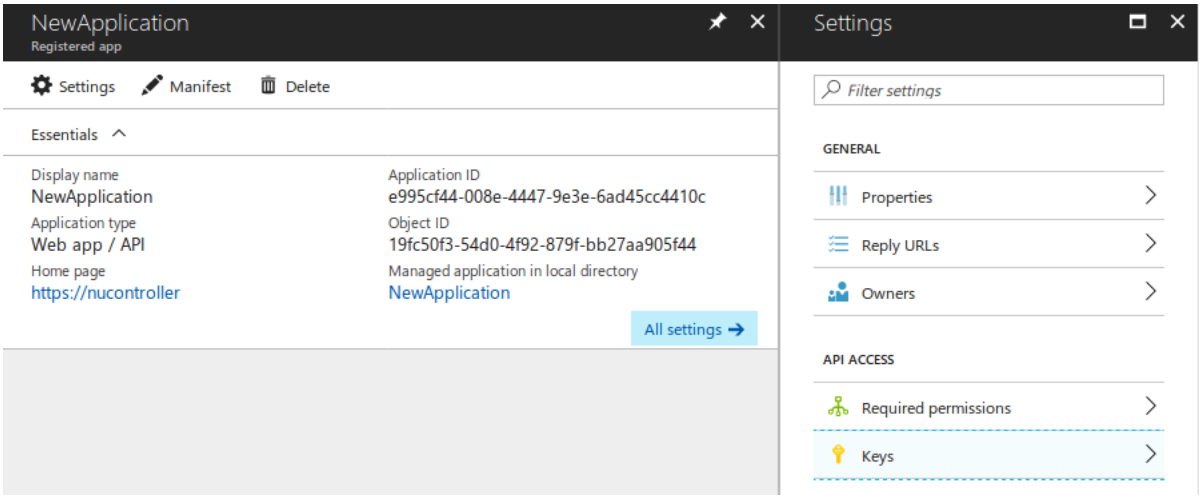


8. Copy the **Directory ID** since this is your Tenant ID.

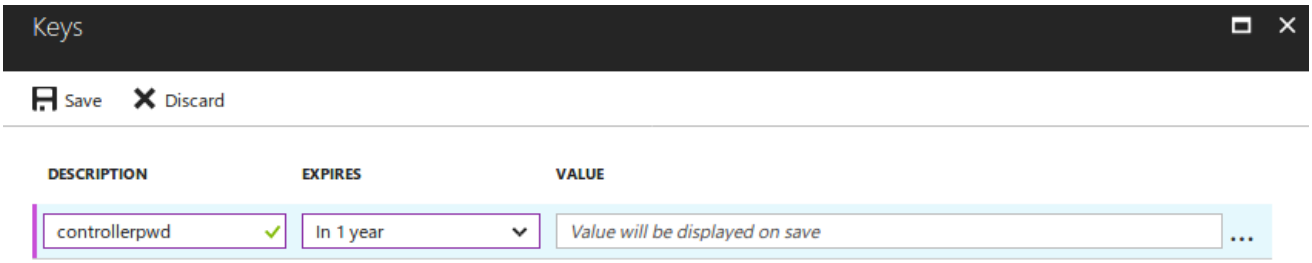


9. The password is a **Keys** Value under the Azure Active Directory application's keys. Go to **Azure Active Directory** on the left hand navigation pane, click on **App registrations**, and click on your desired application.

10. The password can be created under **Keys** in the "API ACCESS" section:

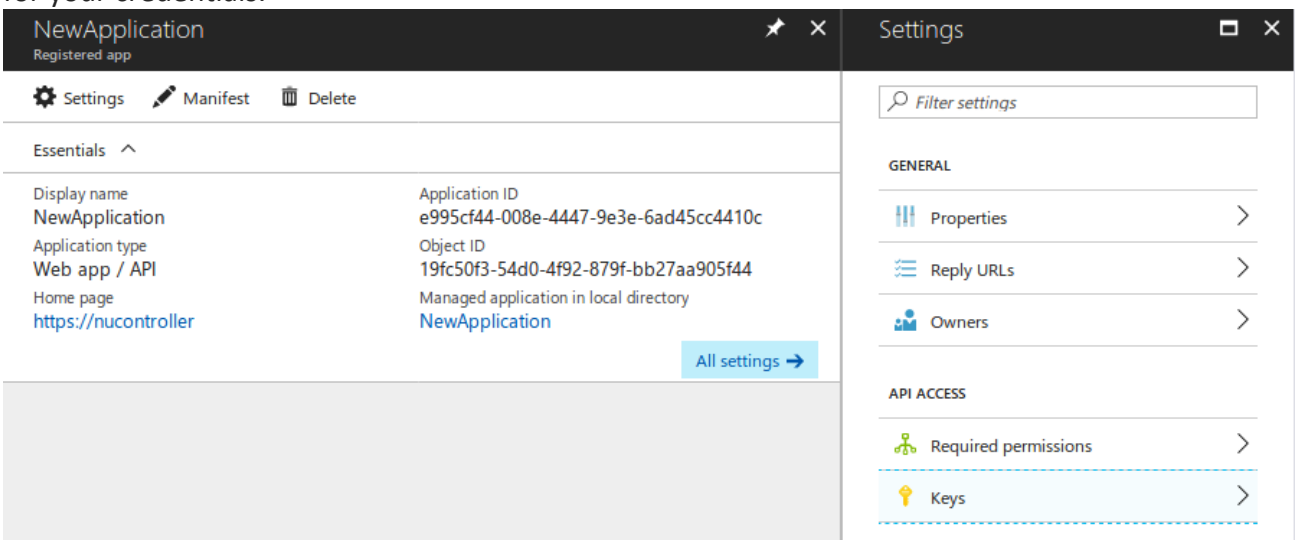


11. Fill in the fields under the **Keys** blade and click **Save**. Make sure to keep record of the **Value** that is displayed right after you click **Save**.
12. **IMPORTANT:** You will need that key value when sending your credentials to the Controller API endpoint.



13. For the **Application ID** . Go to **Azure Active Directory** on the left hand navigation pane, click on **App registrations**, and click on your desired application.

14. Under the application's pane, you will see **Application ID** field. Copy this field and save it for your credentials.



One-Time Controller Configuration

1. The final step in the process is to link the controller to the Azure service principal. All Controller configuration will use Postman to make the appropriate RestAPI calls. Any program or script that has the ability to make RestAPI calls can be used.

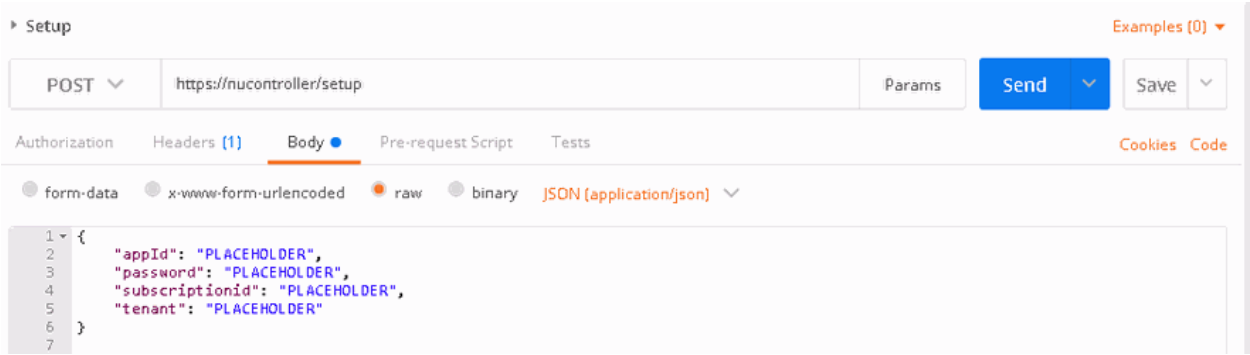
   Note: The controller can only be accessed from inside the Vnet/VPC by default. This can be changed using various cloud connectivity options. The simple solution is to launch a windows VM and load postman on the VM. This is part of the Nubeva POC environment.

   Note: If you need additional assistance in setting up postman with the various Nubeva collections and environmental parameters, please refer to the Postman Setup for Nubeva.

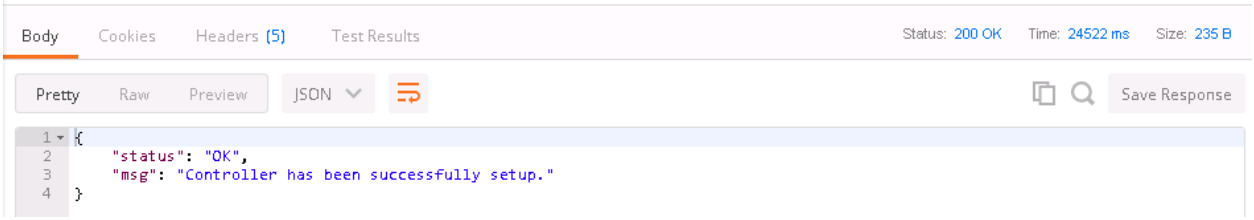
2. In postman, locate the POST command with the following URL: https://{{apiUrl}}/setup

   a. The {{apiUrl}} is an environmental variable. It is usually "nucontroller" in most environments. The value can be any IP address or FQDN.
   b. This is the command which will associate this controller with the service principal created earlier.

3. Then click on the **Body** tab, and include your Azure service principal credentials created above. There is also example data in the Nubeva Postman collection. All of this data can be collected from the command "az account show" and from the detailed properties of the app service you created above.



4. Then click **Send**. This call should take about 30 seconds. At the end, you should see a similar result as below.

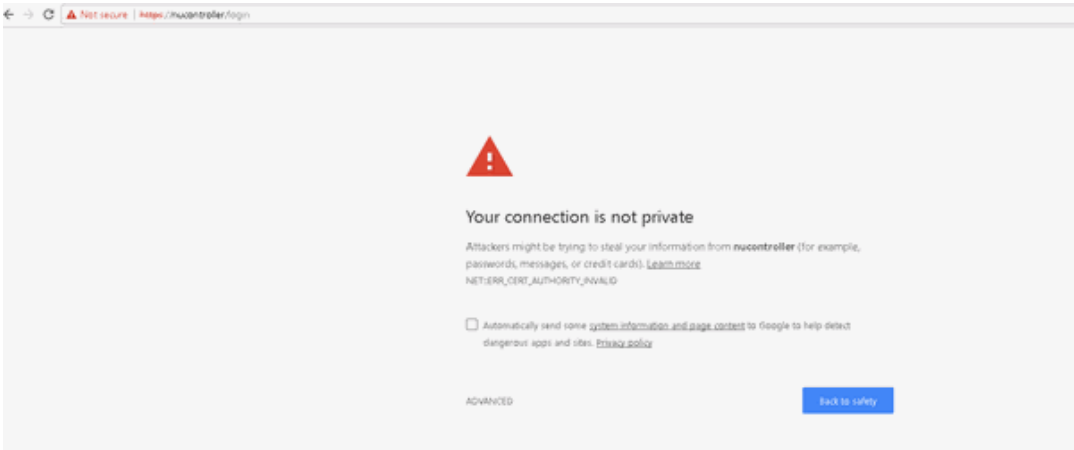


NOTE: If you do not have permissions to access Microsoft's Graph API endpoints, then your reply url cannot be confirmed. So, you will receive a message to check that the reply url has been set through the Azure Portal. Even though you may receive this message, your controller has been successfully setup. Please make sure that the reply URL is included in your Azure Active Directory Application's list of reply URLs.

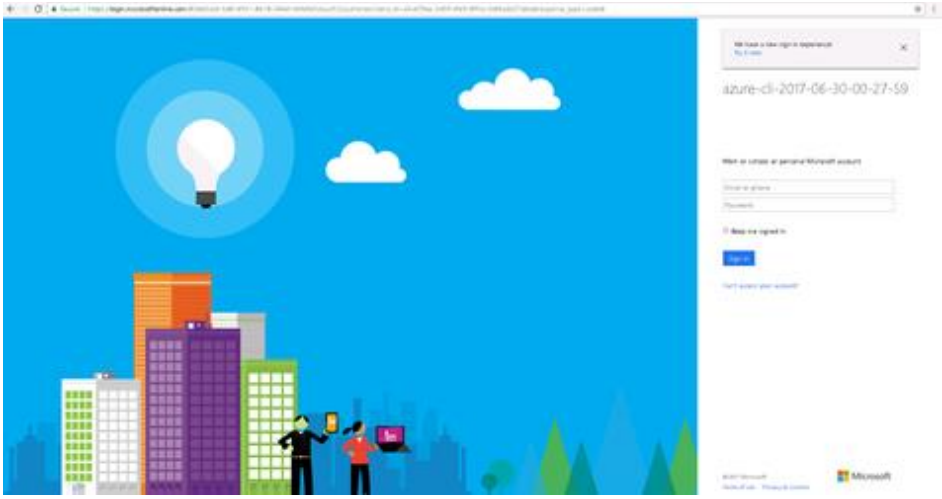
```
{
  "status": "OK",
  "msg": "Controller has been successfully setup.",
  "unconfirmedValidations":
          ["ReplyURLUnconfirmed: https://nucontroller/callback. Possibly
due to insufficient privileges or requirement was not met. Verify
through Azure Active Directory application's reply-urls list."]
}
```

5. The last step in the setup is validating an authentication token. This process will be repeated every time a new token is required. As with most RestAPI environments, these tokens last for 60 minutes.

6. To begin the authentication process, access https://nucontroller/login, via a web browser.



7. Accept the SSL certification errors and proceed to the destination URL. By proceeding to the webpage, you will be redirected to Microsoft's Account login page.



8. Please enter in your credentials. If valid, you will be redirected to the /callback endpoint, which will display your “auth_code.” This “auth_code” maps directly to the “token” variable inside the “Nubeva Controller” environment in Postman. Copy this auth_code into the token field.

9. With the auth_code/token set, the Nubeva Controller is fully installed and configure. In the next section, we will document how to install a StratusEdge node for tapping.