Security Removable Media Manager
(secRMM)

Intune Access Control Setup Guide

Version 9.9.28.0

(November 2020)

Protect your valuable data

If you have any questions regarding your potential use of this material, contact:

Squadra Technologies, LLC
7575 West Washington Ave
Suite 127-252
Las Vegas, NV 89128 USA
www.squadratechnologies.com
email: info@squadratechnologies.com

Refer to our Web site for regional and international office information.

TRADEMARKS

Disclaimer

Squadra Technologies Administrator Guide
Created - March 2011

Contents

Introduction

Squadra Technologies *security Removable Media Manager* (**secRMM**) software is Windows security software that runs on your company's workstations and servers. secRMM manages and monitors removable media. In this context, Removable media is defined as external hard disks, USB (flash) drives, smart phones, tablets, SD-Cards, CD-ROM and DVD. Such devices typically use the computers Universal Serial Bus (USB) ports to connect to the computer. Removable media devices are popular because they are very convenient when you want to copy files around or backup data. secRMM allows you to track all write activity to the removable media devices in your computer environment as well as giving you the ability to control (or authorize) who can write to the removable media devices.

This document is focused on removable media that is contained within mobile devices (either the devices flash storage or an SD-card). Mobile devices are so popular now that there are many software products which help organizations manage how mobile devices are used within the work place. These software products are called/categorized as "Mobile Device Management" (MDM) products. Microsoft has a MDM product named Intune that runs in the Microsoft cloud. Microsofts cloud is named Azure. All of the MDM products focus on security.

Unfortunately, when it comes to connecting the mobile device over a USB cable, the MDM products either allow or disallow a USB connection (i.e. either on or off). This is a sub-optimal solution for two reasons:

1. The device must be enrolled in the MDM to enforce this rule (i.e. either USB allowed or disallowed). This is roughly analogous to saying that police officiers can catch all criminals but only if the criminals first go to the police station to get finger-printed. Otherwise it is not possible to catch them.
2. A rule that only enforces allowed or disallowed is on the one hand (allowed) too relaxed and on the other hand (disallowed) to restrictive. The right solution is to have policy such as secRMM to control read, write and who can have this access, from where they can copy data from, etc.

secRMM can be configured to use the mobile device definitions in Microsoft Intune to decide if a mobile device can be used over the USB connection. A mobile device can be used over a USB connection to transfer files to and from the mobile device to the Windows Desktop computer it is connected to (over the USB cable). You can configure secRMM to check if the mobile device is simply enrolled in Intune or that the device's state (within the MDM) must be "compliant" before it can be used over the USB connection. Whether or not a mobile device is compliant is defined by the organization configuring the MDM and the devices within the MDM. An example would be that a device is compliant if the organizations apps were installed on the device.

If the functionality in the paragraph above is a desirable feature for your environment, this document will help you setup this secRMM feature.

Prerequistes

You will need to have a licensed Intune instance in Azure. By default, this also means you will have an "Azure Active Directory" (AAD) instance. Both Intune and AAD are defined within your Azure tenant. A tenant is a Microsoft term that can be thought of as a container that holds services, programs, device

defininitions, data and virtual computers in the cloud that your company can access. Each tenant within Azure has a unique id (Microsoft calls this the "tenant id" and "directory id").

Setup overview

Here are the high-level steps we will take to setup the secRMM connection to your Azure tenant (i.e. AAD and Intune).

1. Setup an Azure application
2. Configure secRMM to get mobile device state from Intune via the Azure application

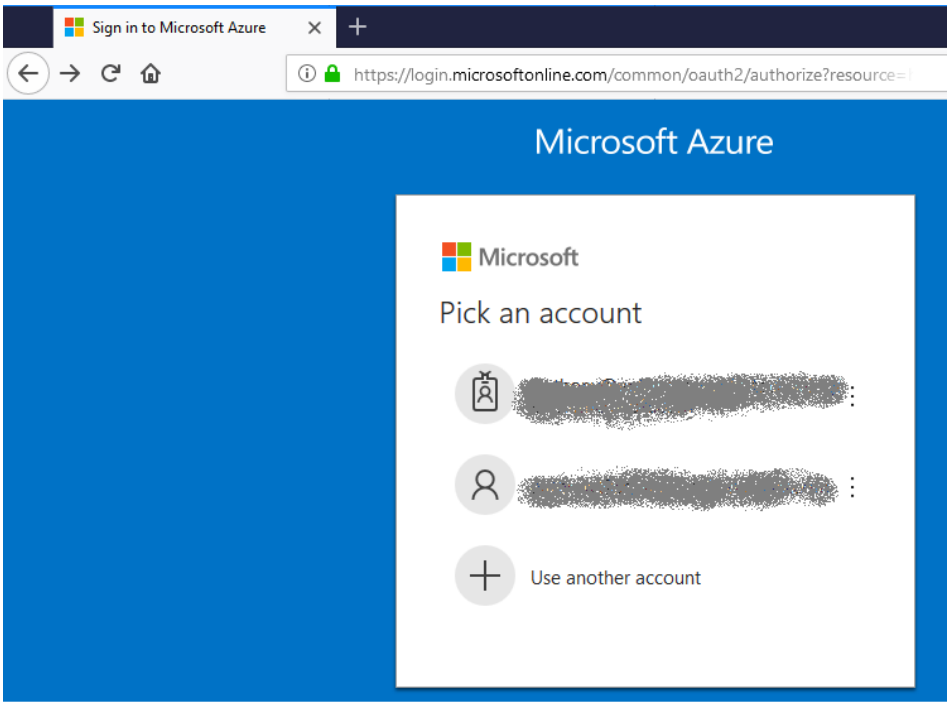
Setup details

Setup an Azure "application"

Of course, Microsoft does not allow external programs access to your Azure tenant by default. If you want to allow a program access to the services (i.e. Intune) within your Azure tenant, you must define an "application" within your Azure tenant (via your AAD). The external program must go through these Azure applications to access the services and data within Azure. These Azure applications are really nothing more than a collection of security settings that tell Azure what parts of Azure the external program can access. This document will walk you through the process of setting up the application so you really do not need to have a deep understanding of the whys and hows. Once you follow the steps, the end result will be that secRMM can access the mobile device data in your AAD and Intune so that secRMM can make the decision about mounting your mobile devices over USB connections based on your Azure tenant data.

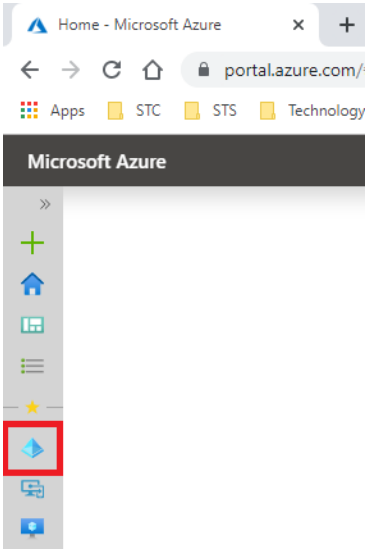
Login to your Azure tenant

Using a web browser, go to URL https://portal.azure.com. You will need to supply your Azure userid and password. The userid you use must be defined as the Azure Global Administrator account. Azure Global Administrators are the only userids that can define Azure applications.

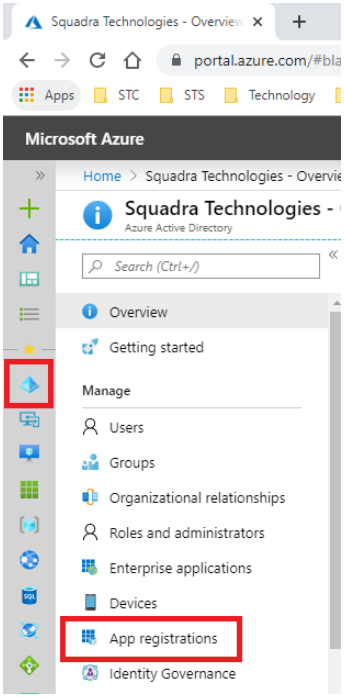


Create Azure Application

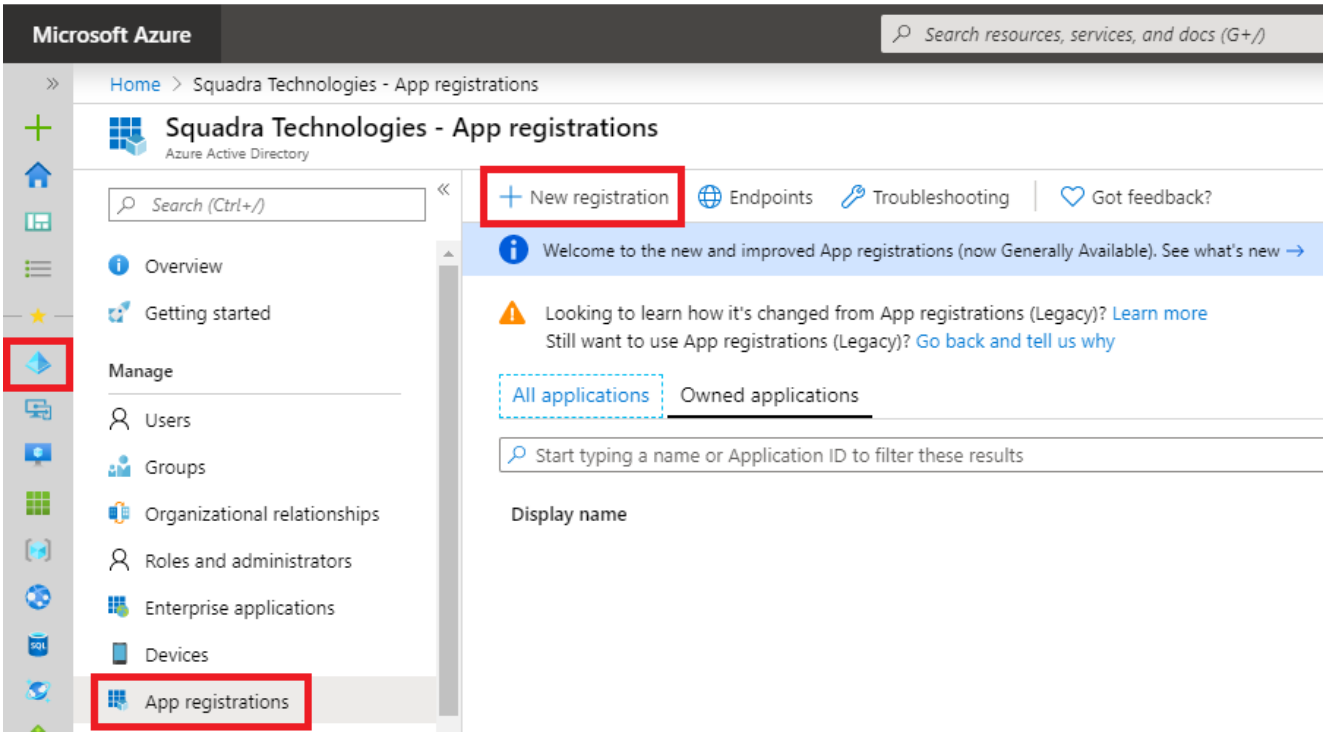
Once you are logged in, you will be at your Azure tenant Dashboard.
On the left hand side of the web page, find and select "**Azure Active Directory**".



A new column will appear just to the right of the Dashboard column.
In the new column, select "**App registrations**".

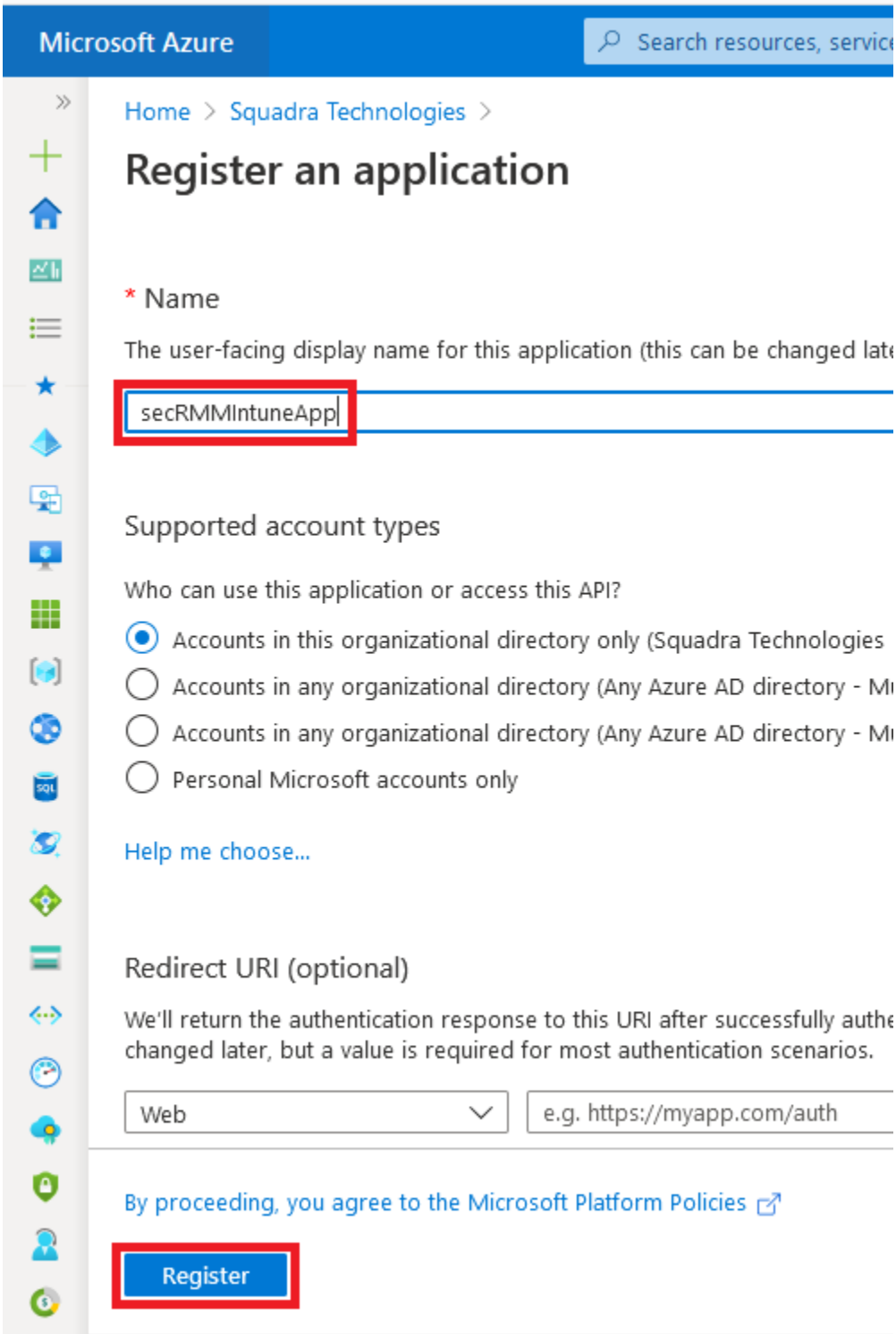


A new page will appear just to the right of the "App registrations" column. In the new page, at the top, select "**New application registration**".

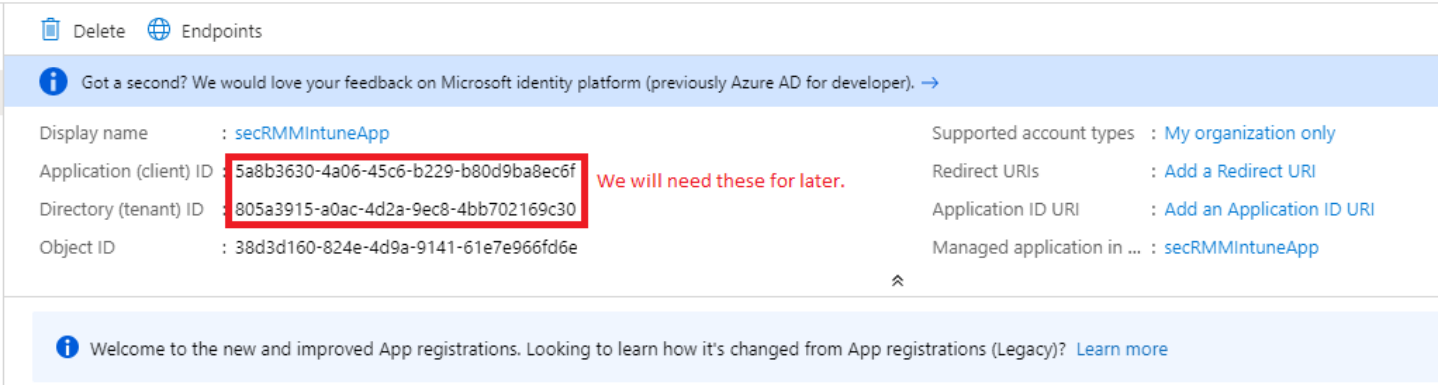


A form will appear that wants you to specify the "Application name", "Supported account types" and a "Platform configuration" as shown in the screenshot below. For the "Application name", type **secRMMIntuneApp** (although this is a free form text field and can have any value you want, we recommend you specify secRMMIntuneApp so the documentation below will match your environment).

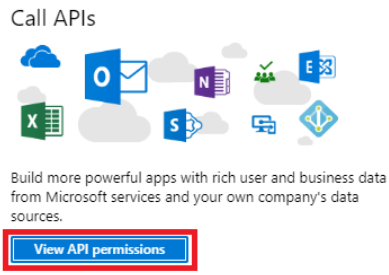
You do not need to change anything else so now click the **Register** button at the bottom as shown in the screenshot below.



Azure AAD will create the application when you click the Register button and will show you the new application as shown in the screenshot below. We will need to use the "Application ID" and "Directory ID" on this screen when we configure secRMM later in this document.

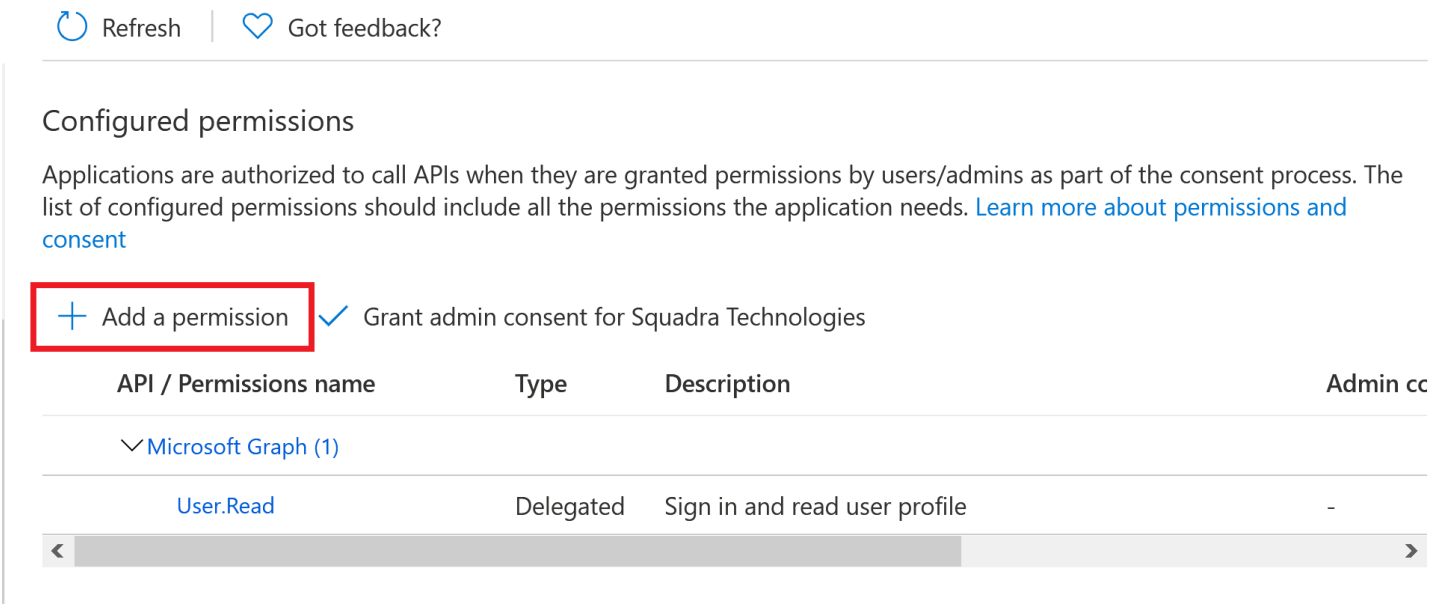


Next, click the "**View API permissions**" button (in the middle part of the page) as shown in the screenshot below.

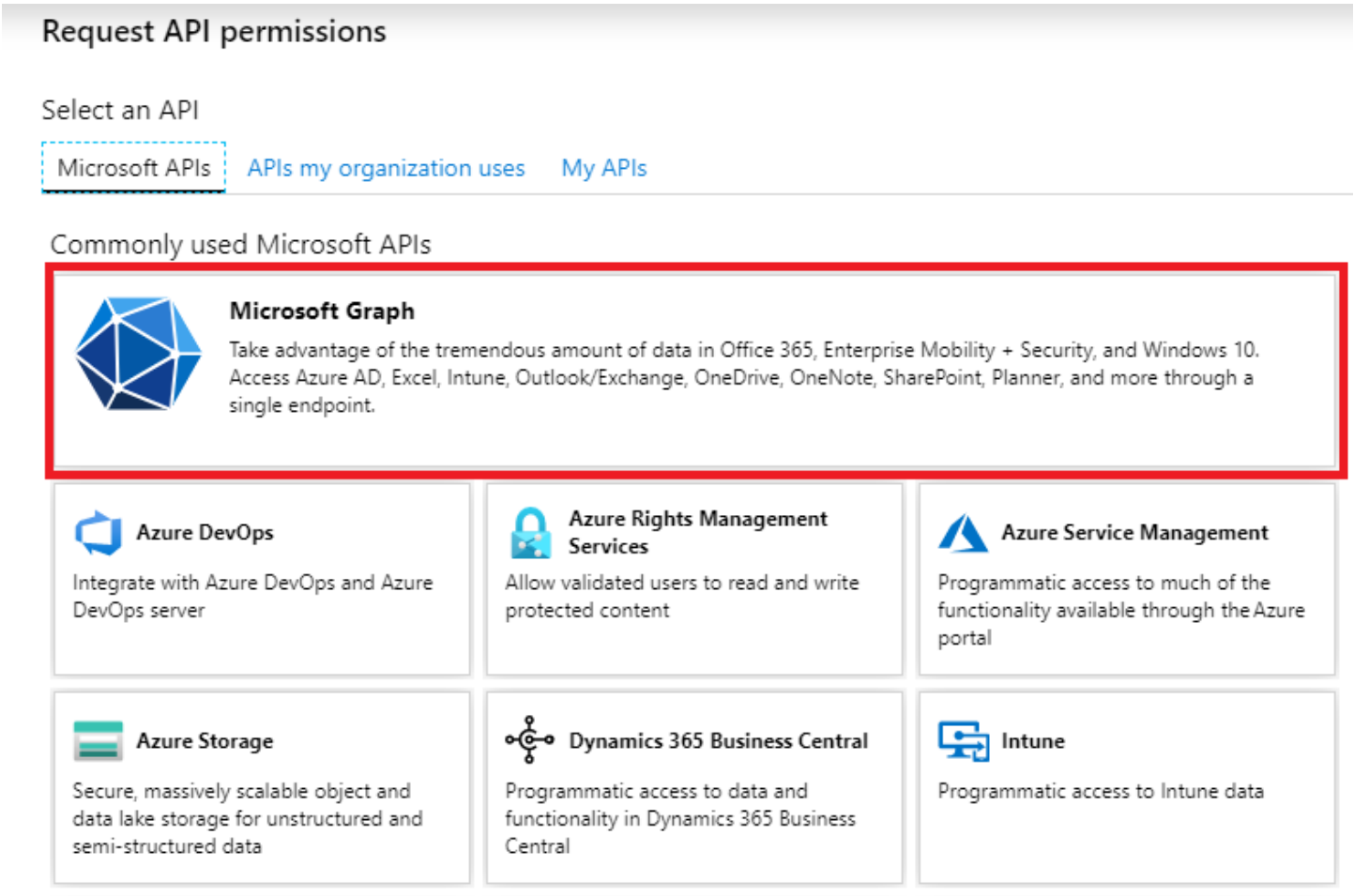


When you click the "**View API permissions**" button, you will see a new column to the right of the Application information named "API permissions" as shown in the screenshot below.

To add the new API/Permission, click the "**Add a permission**" button.



A new window will appear on the right hand side of the screen as shown in the screenshot below. Click the Microsoft Graph button.



At this point, you will need to decide which permission (Delegated or Application) you want to use for your environment.

The "Delegated permission" will let you specify an Azure userid and password to have secRMM authenticate with Azure Intune. **Note: As of 12/01/2020, using a "delegated permission" (userid and password) seems to no longer be supported by Azure. If you are currently using "delegated permission", please contact Squadra Technologies support for help migrating.**

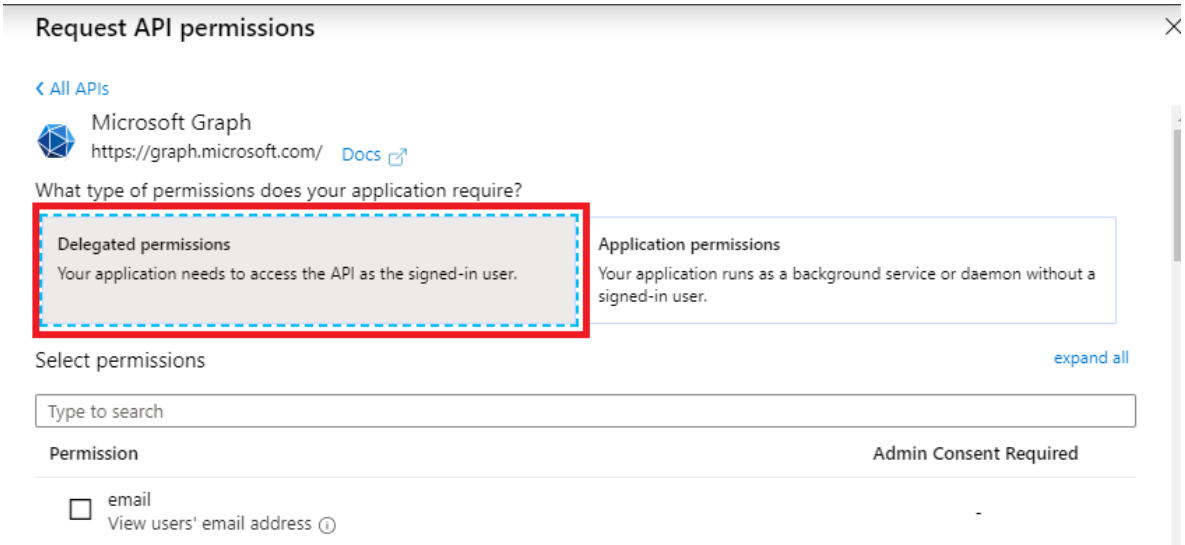
The "Application permission" will let you use an "application secret" (i.e. a password associated with the application) to have secRMM authenticate with Azure Intune.

One of the main reasons for not using "Delegated permission" (i.e. an Azure userid and password) is if your environment requires all userids to use multifactor authentication. If this is the case, then you need to use "Application permission".

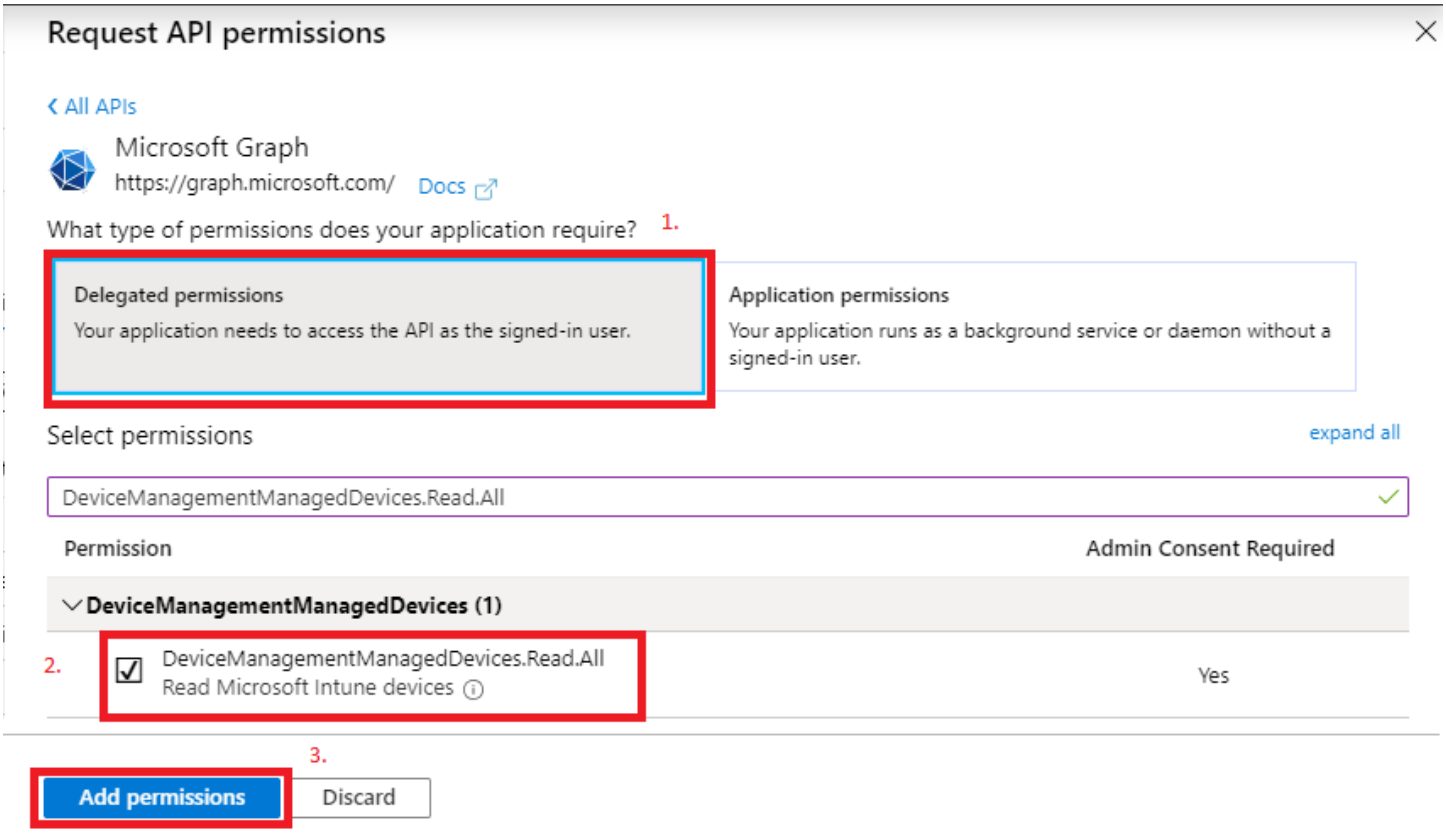
This document will show you how to configure both security types in the two subsections below.
If you are unsure about which one to use, please contact Squadra Technologies support (support@squadratechnologies.com) for assistance.

Delegated permission (userid/password)

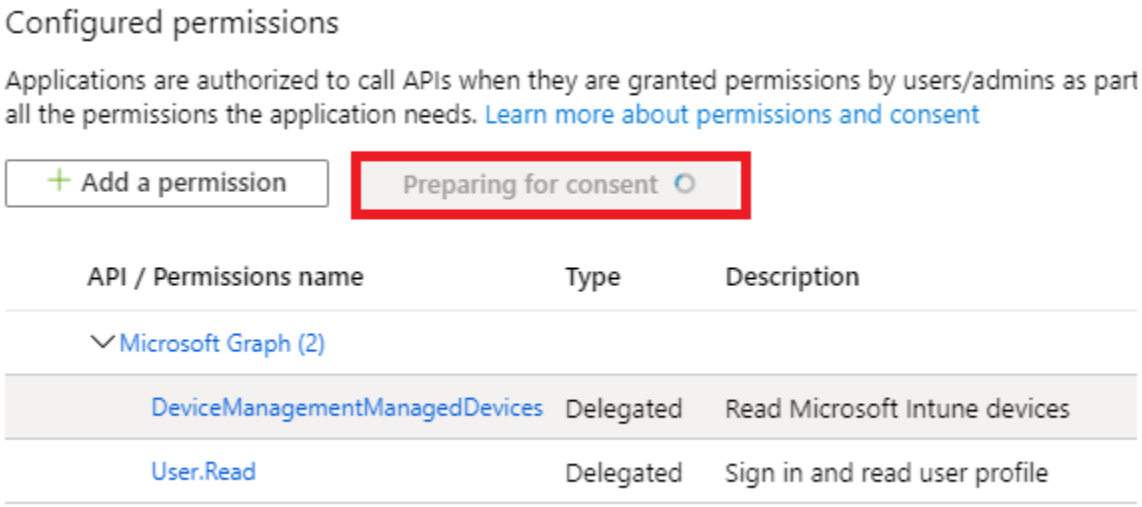
Click the Delegated button as shown in the screenshot below.

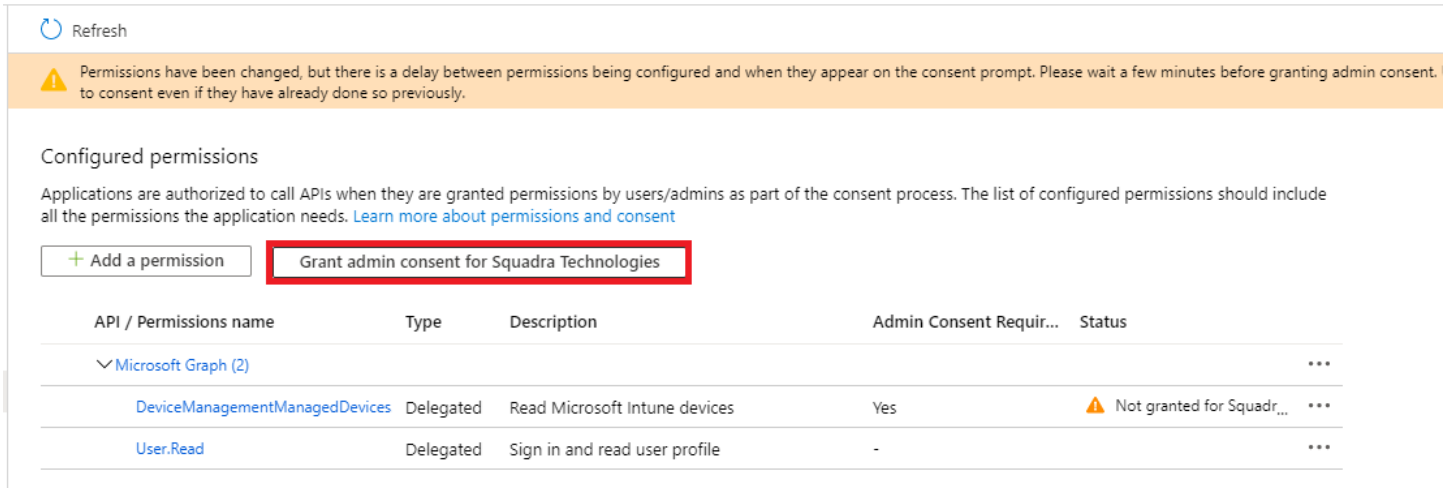


Select the "DeviceManagementManagedDevices.Read.All" permission and then click the "Add permissions" button as shown in the screenshot below.

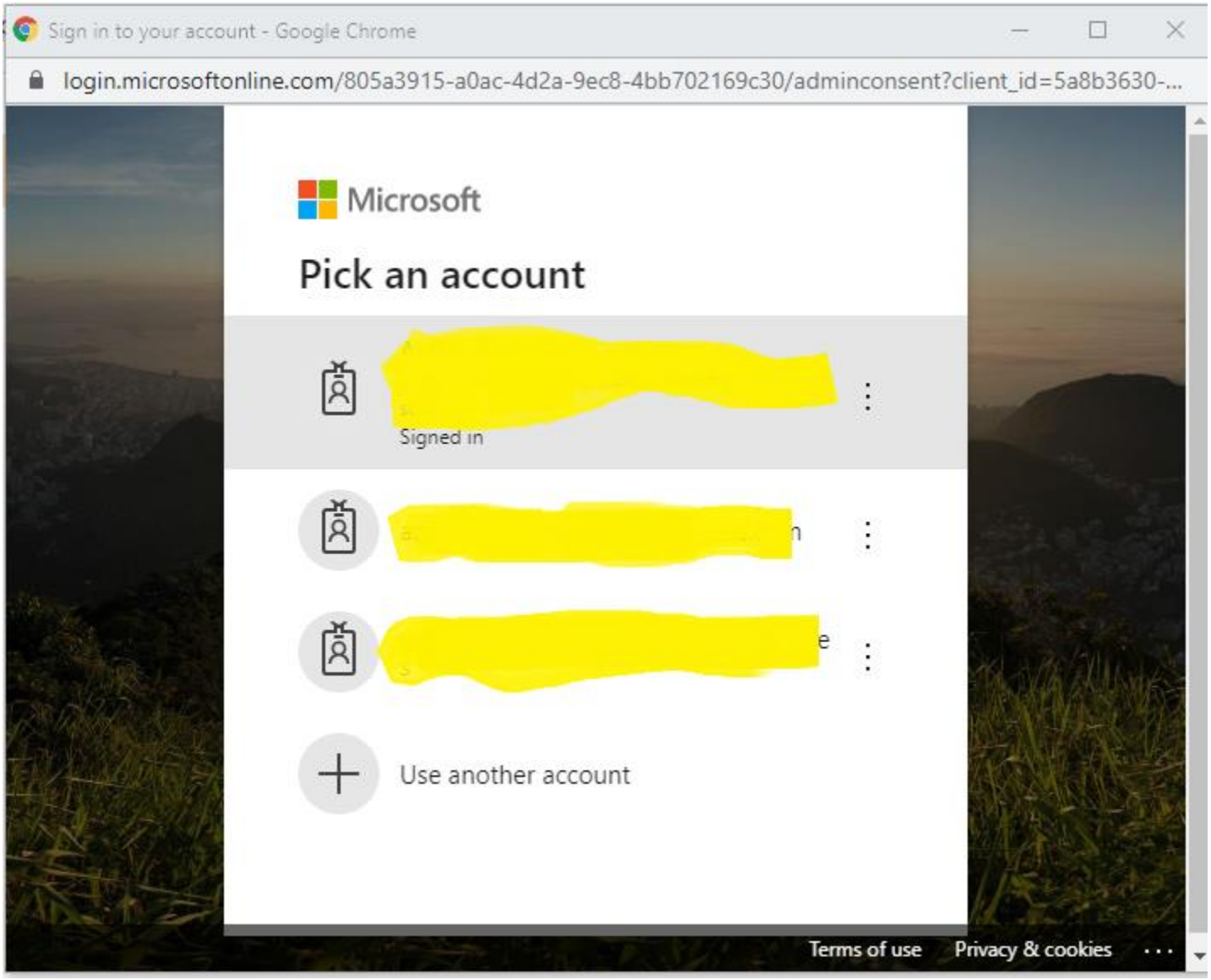


You will now be back on the "Configured permissions" page. Wait for the "Grant admin consent" button to become active and then click the "Grant admin consent" button as shown in the screenshot below.

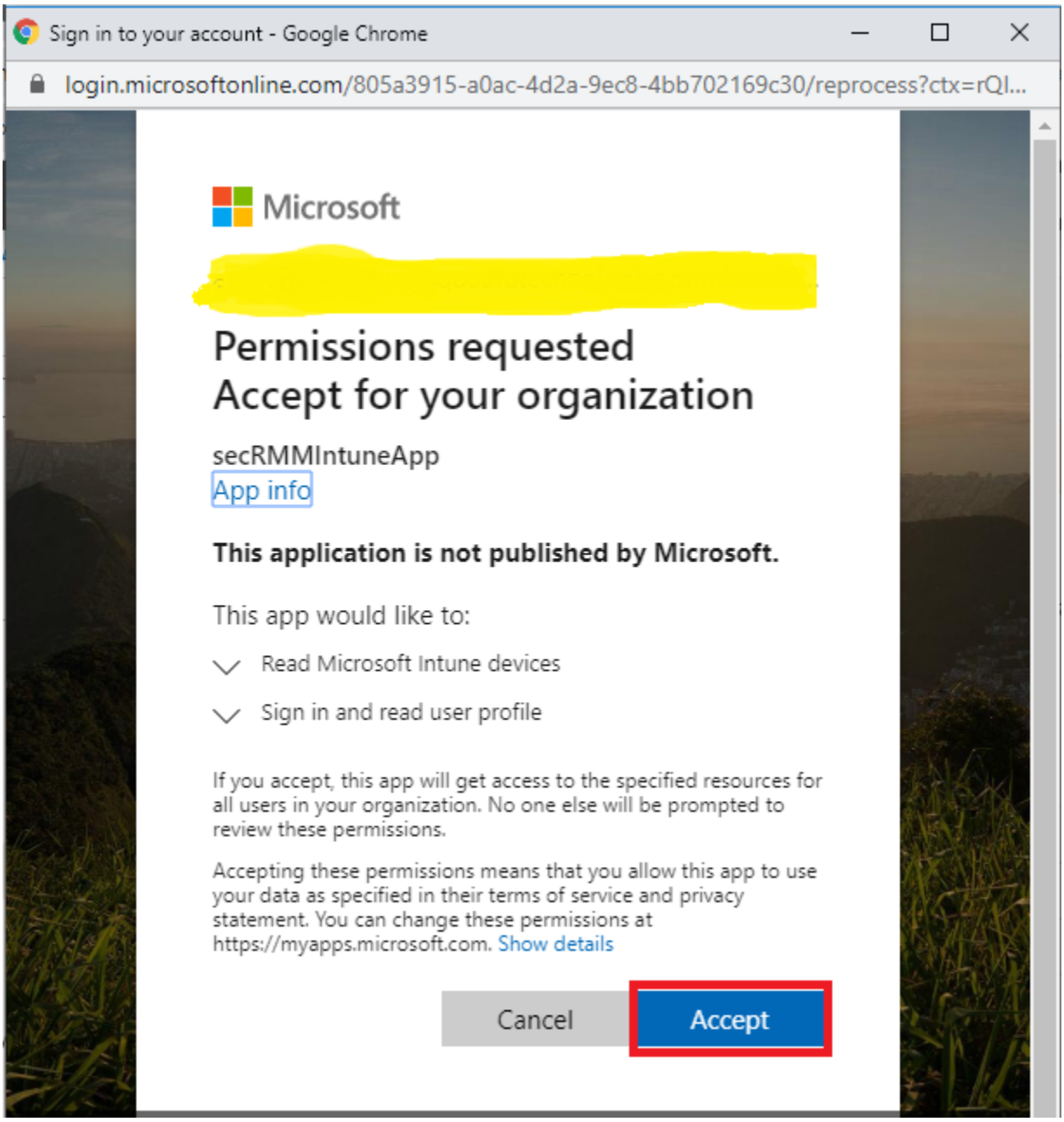




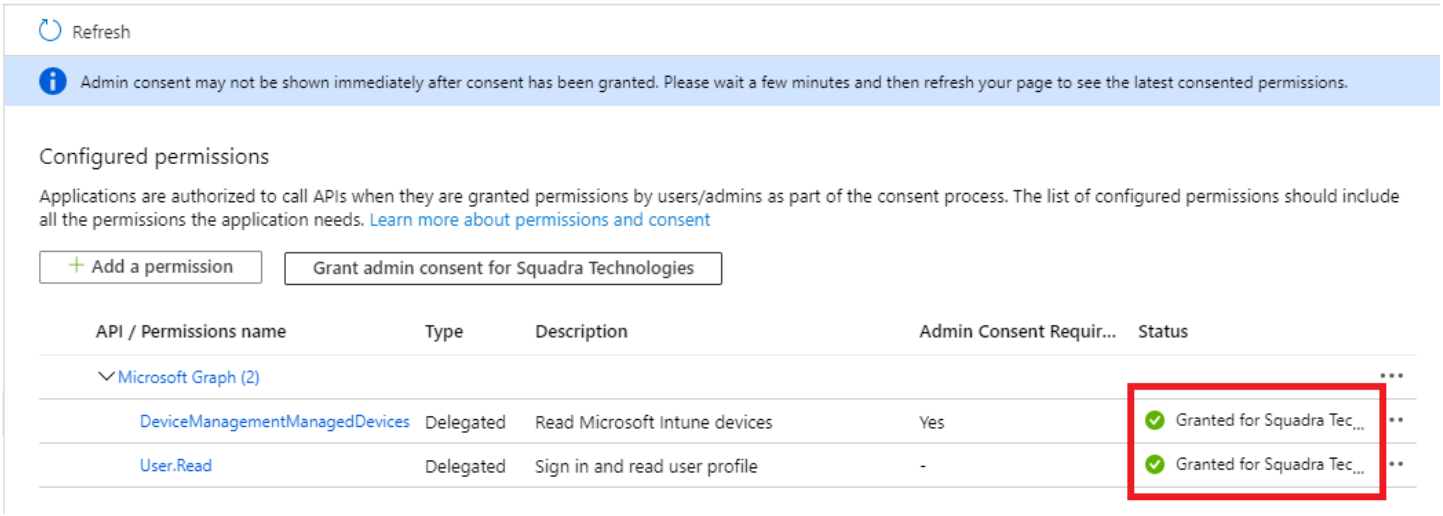
A new window will appear asking you to select an Azure user account as shown in the screenshot below. Make sure you select the Azure global administrator account.



Click the "Accept" button as shown in the screenshot below.

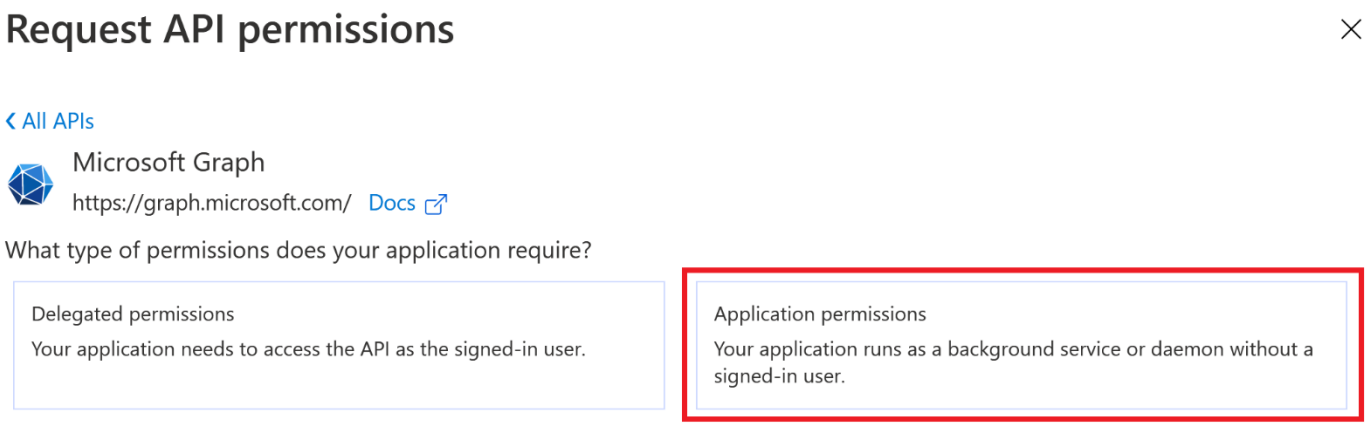


You will now see that the permissions have been granted as shown in the screenshot below.



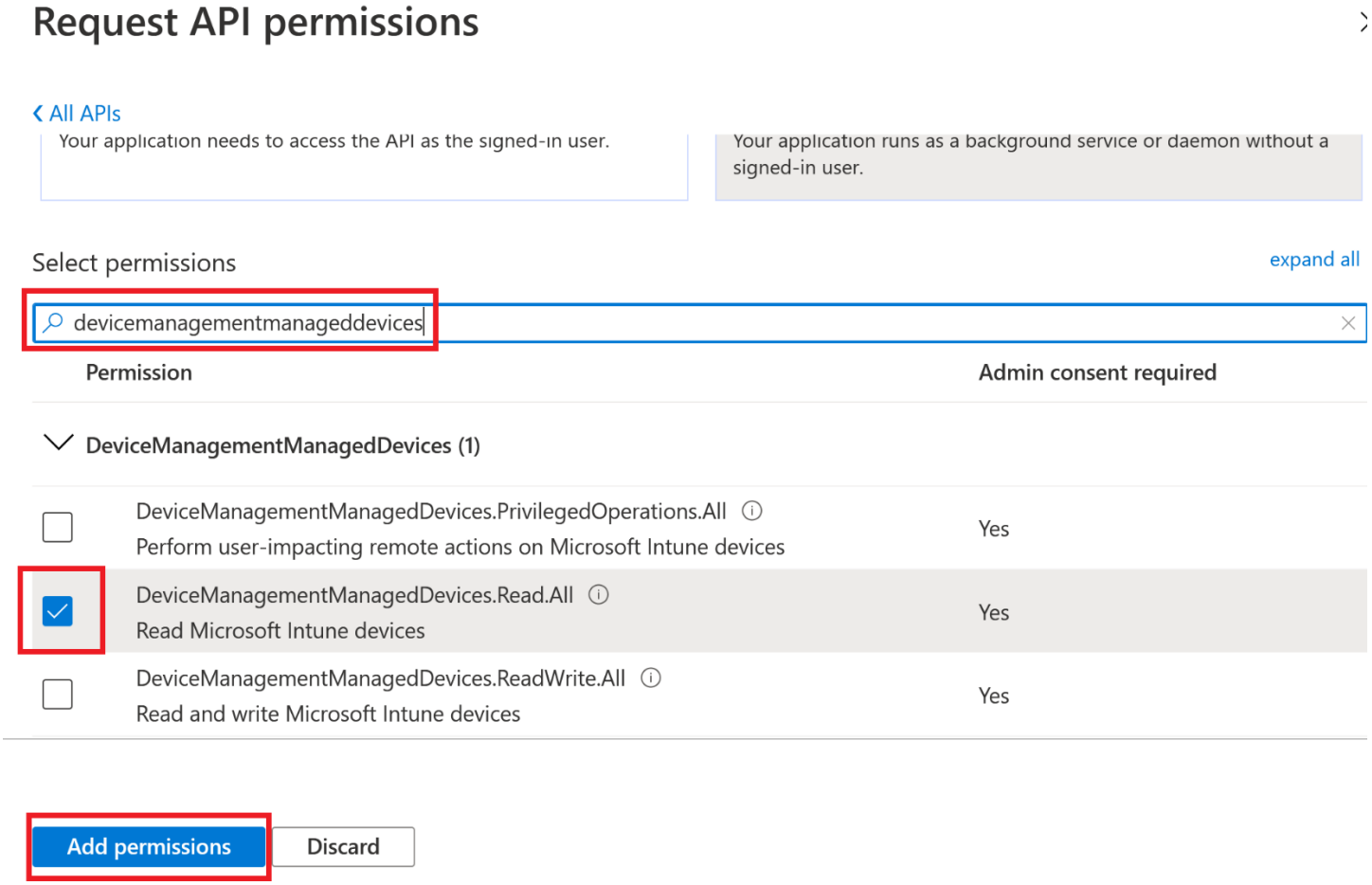
Application permission (client secret)

Click the "Application permissions" button as shown in the screenshot below.



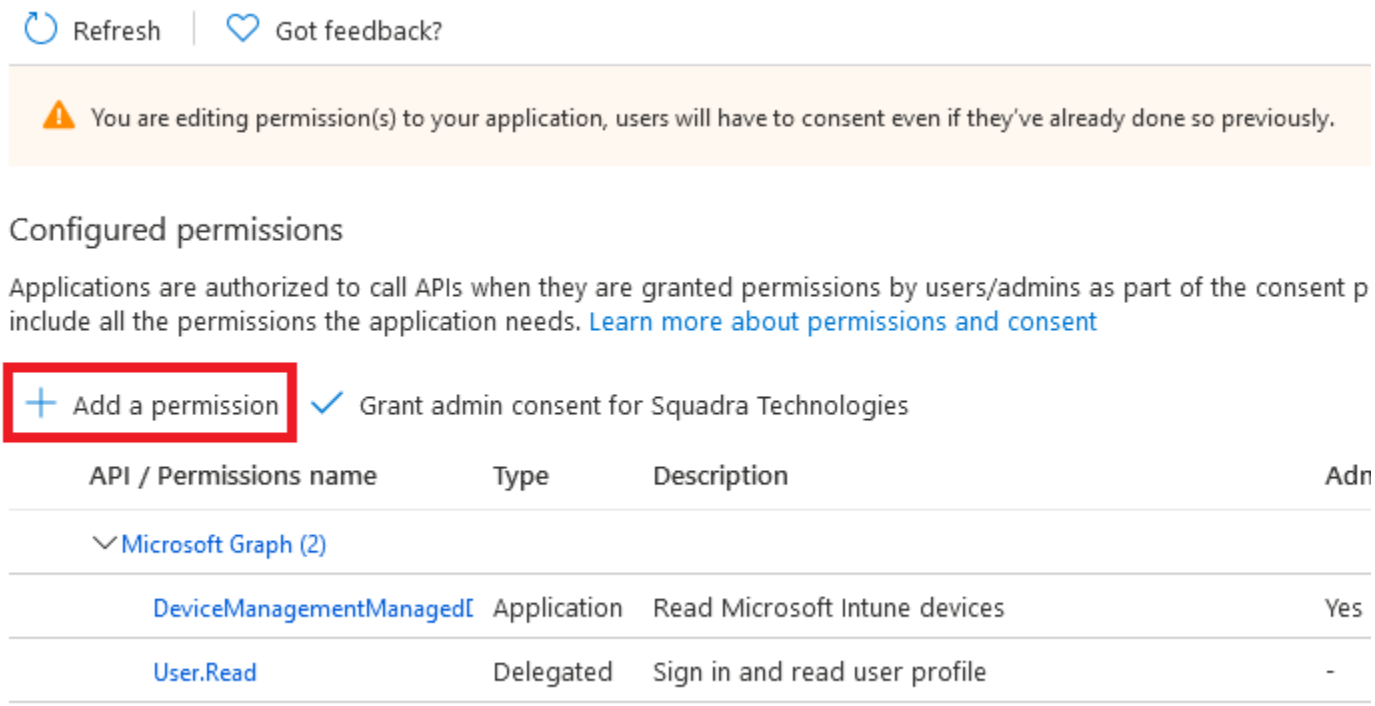
DeviceManagementManagedDevices.Read.All permission

Select the "DeviceManagementManagedDevices.Read.All" permission and then click the "Add permissions" button as shown in the screenshot below.

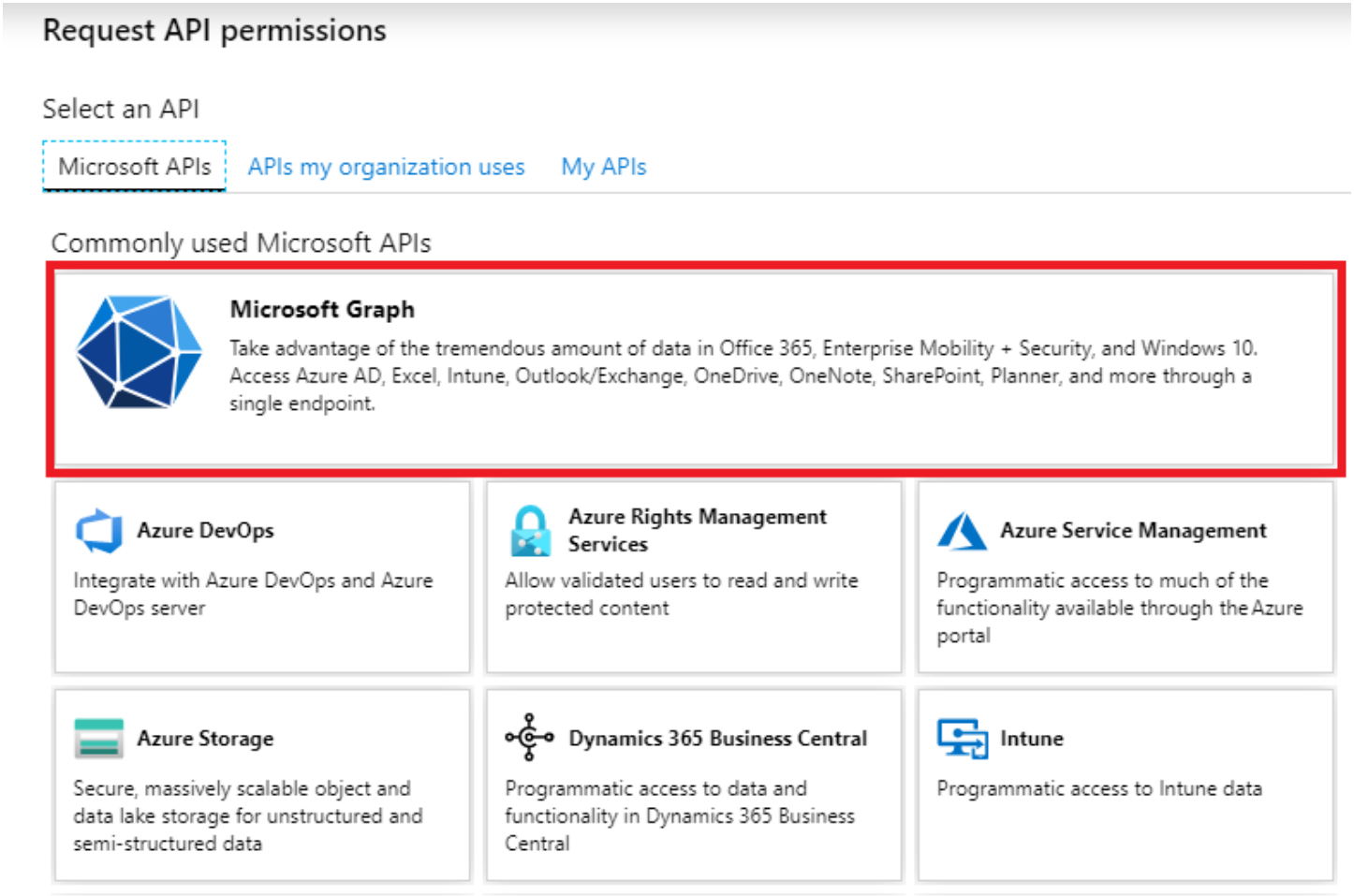


You will now be back on the "Configured permissions" page. Click the "Add permission" button again as shown in the screenshot below.

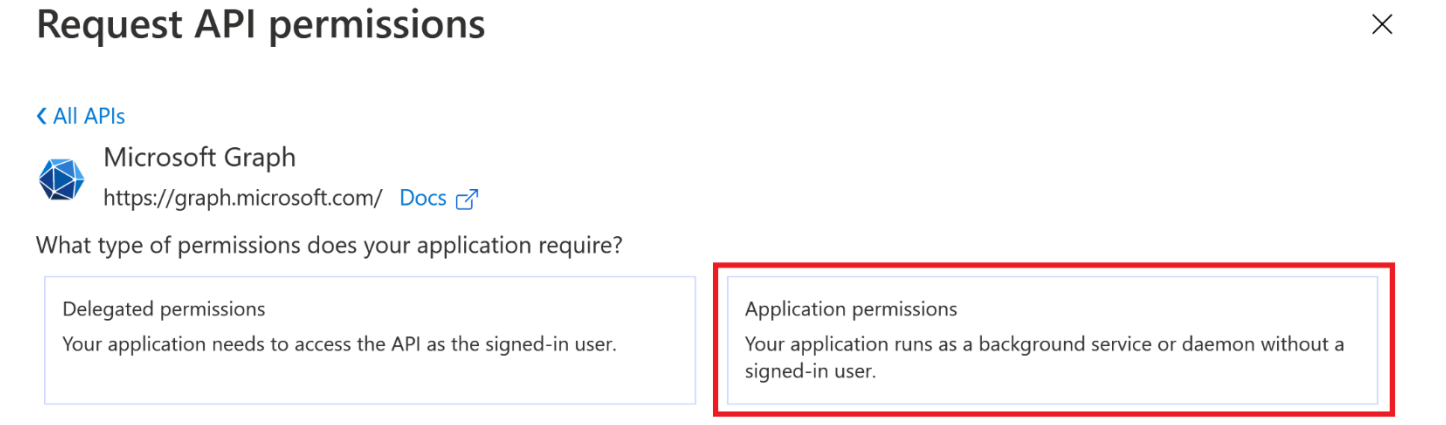
User.Read.All permission



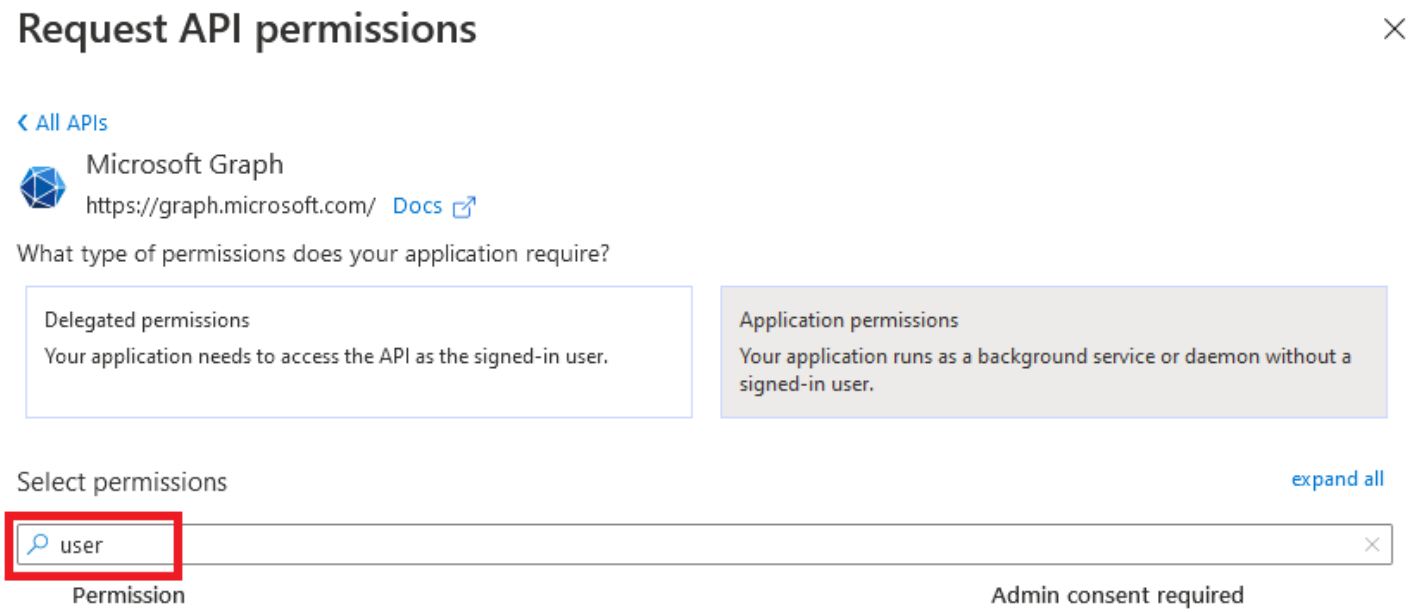
A new window will appear on the right hand side of the screen as shown in the screenshot below. Click the Microsoft Graph button.



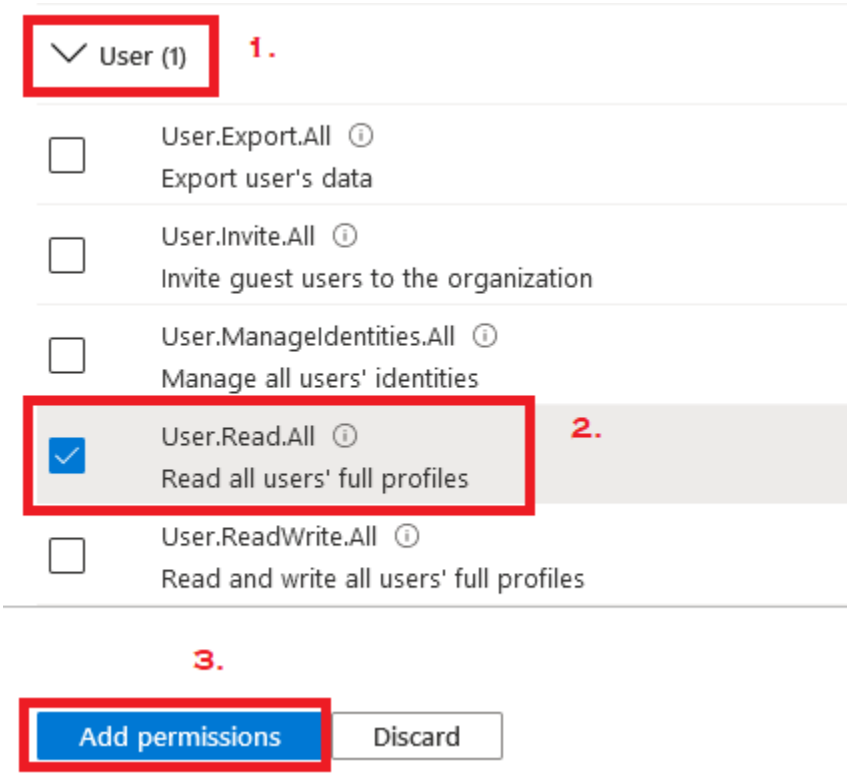
Click the "Application permissions" button as shown in the screenshot below.



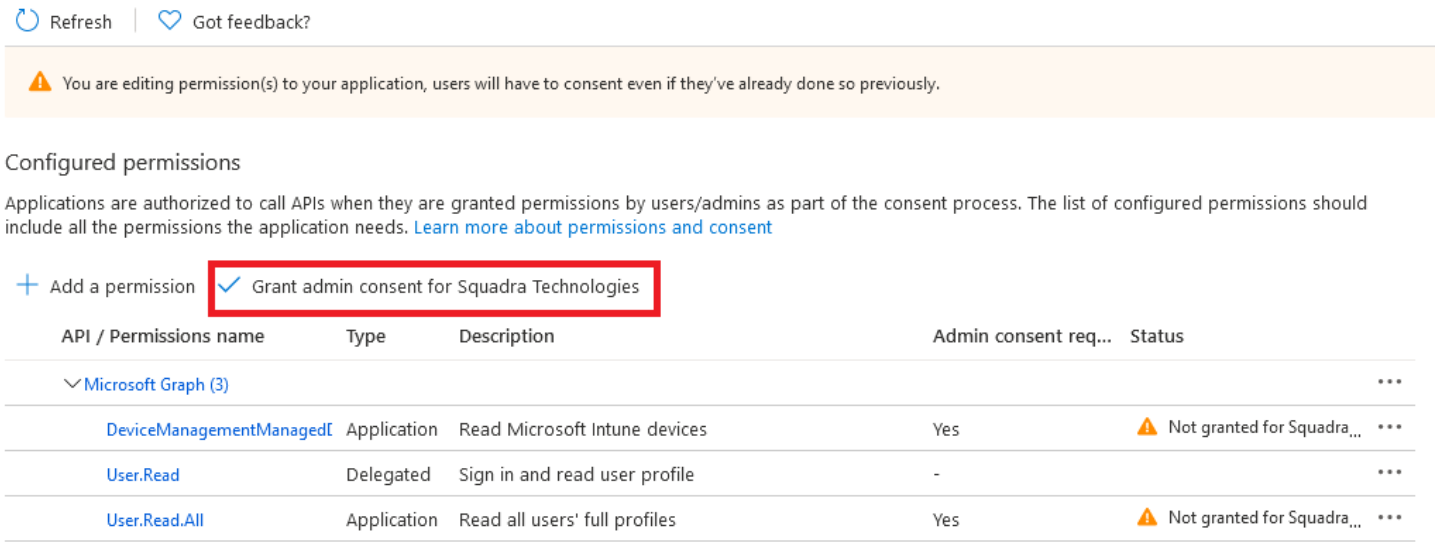
Type "user" in the "Select permissions" textbox as shown in the screenshot below.



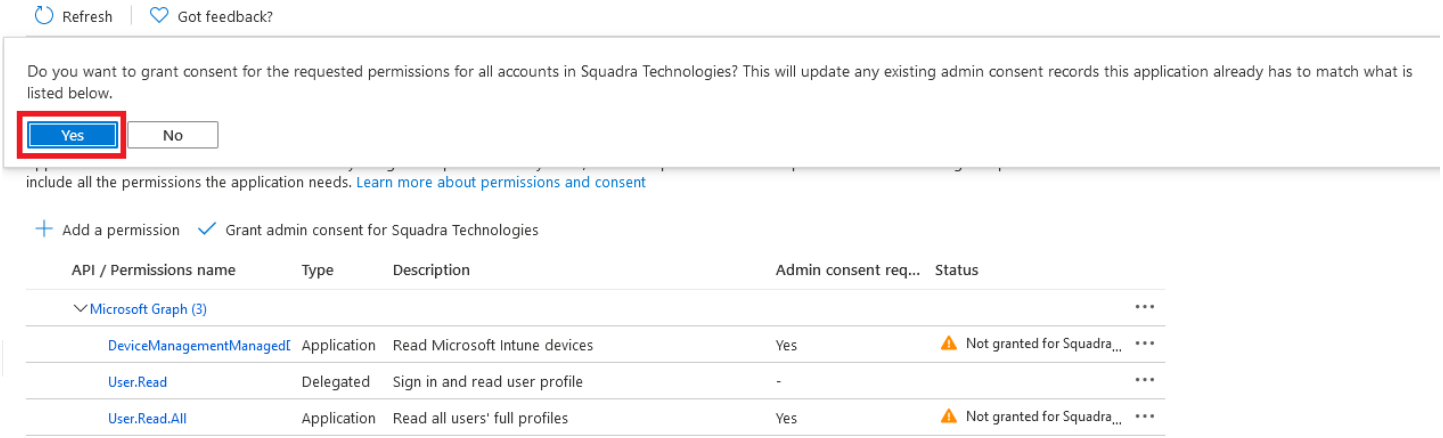
Scroll down the list, expand the user item, select "User.Read.All" and then click the "Add permissions" button as shown in the screenshot below.



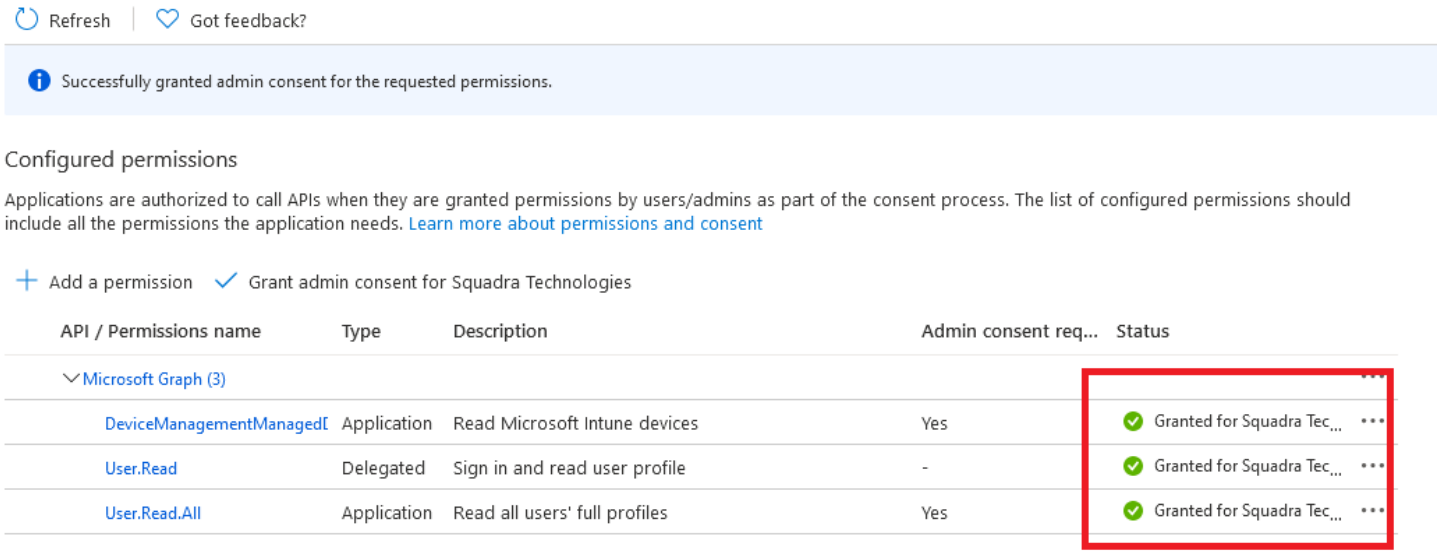
You will now be back on the "Configured permissions" page. Wait for the "Grant admin consent" button to become active and then click the "Grant admin consent" button as shown in the screenshot below.



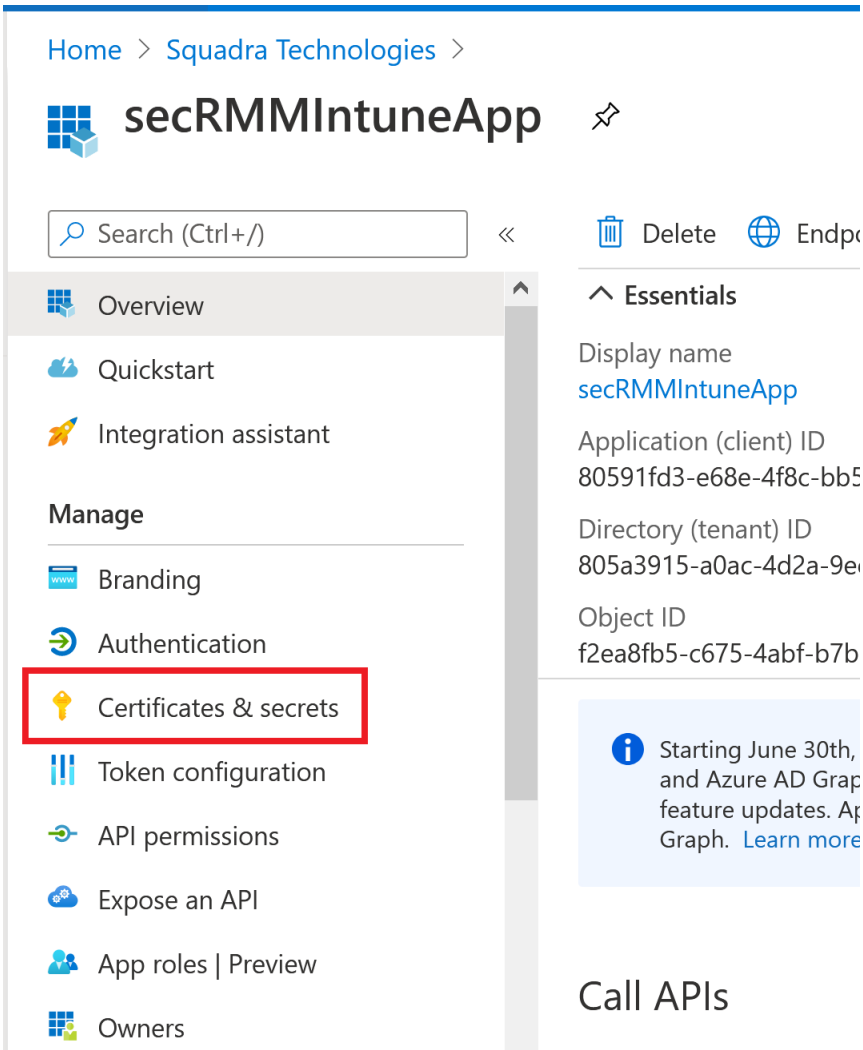
Click the "Yes" button as shown in the screenshot below.



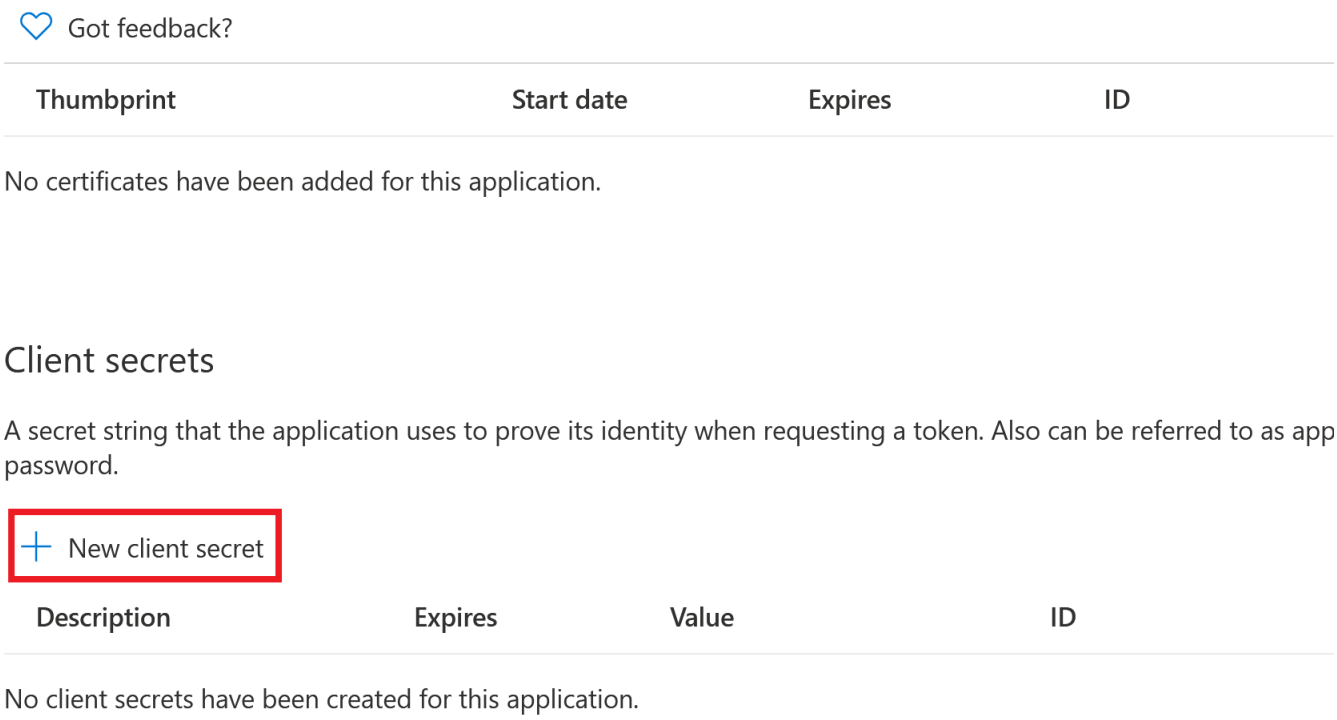
The permissions will not turn to green icons showing success as shown in the screenshot below.



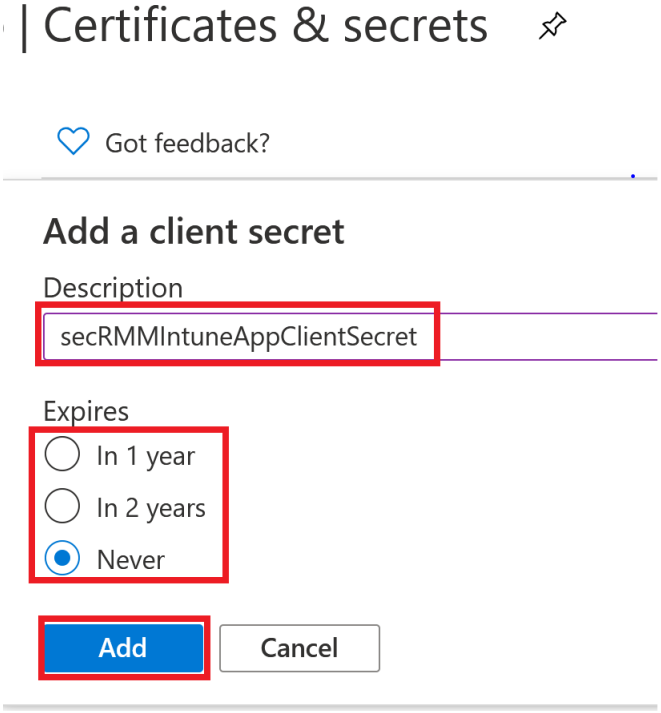
Now click the "Certificates & Secrets" as shown in the screenshot below.



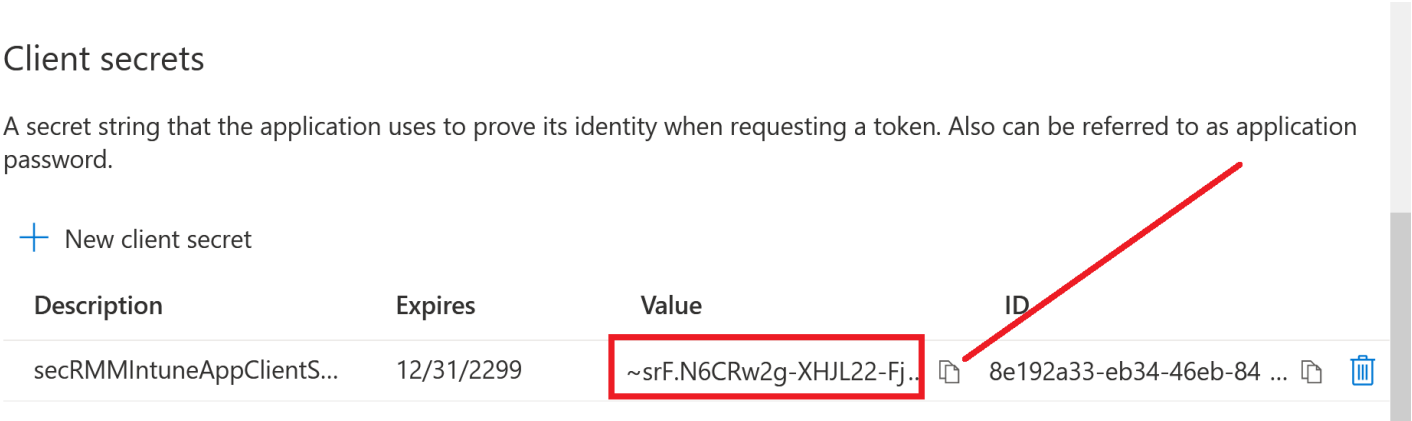
Scroll down a bit on the page and select "New client secret" as shown in the screenshot below.



Fill in the "client secret" information as shown in the screenshot below. Note the the description is free form and you are free to choose when the client secret will expire (based on your environment).



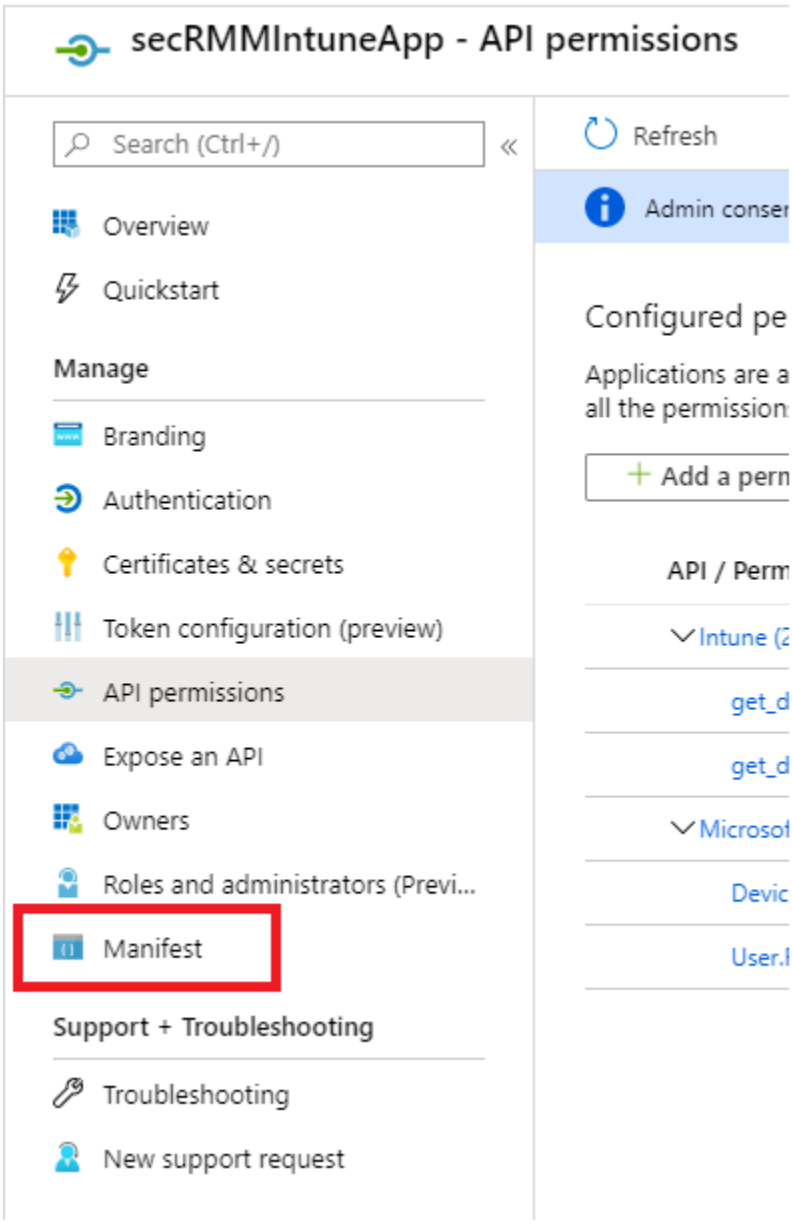
The "client secret" will be generated. You will need to take the Value as shown in the screenshot below. The value will need to be copied into the secRMM property. This is shown later in this document in the section named "Configure secRMM to get mobile device state". You must copy this value now because once you go off the page, the value will not be displayed again. If you forget to copy the value, just delete the old secret and create a new one.



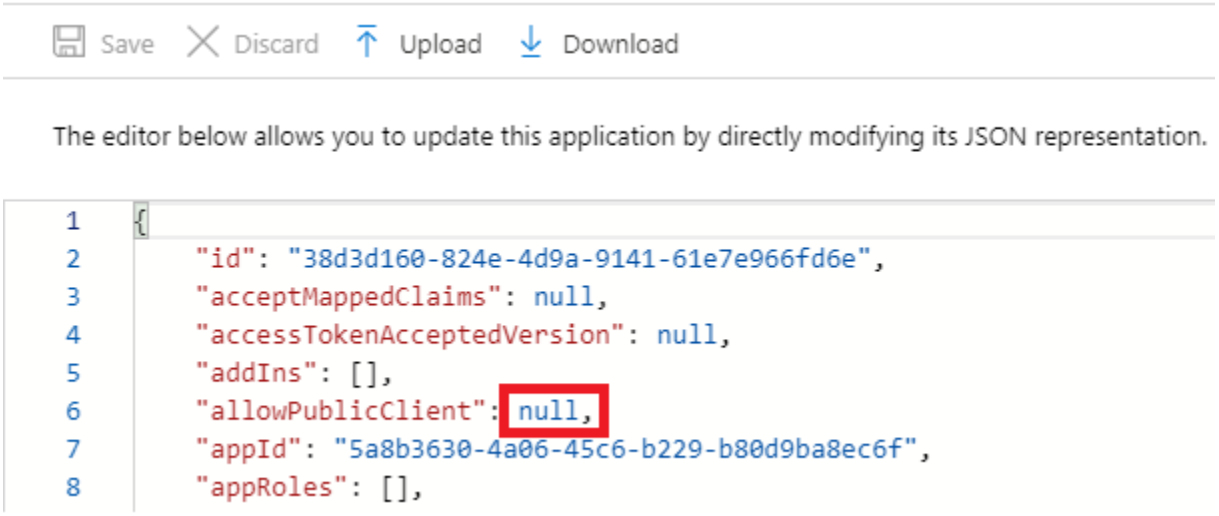
Updating the Manifest

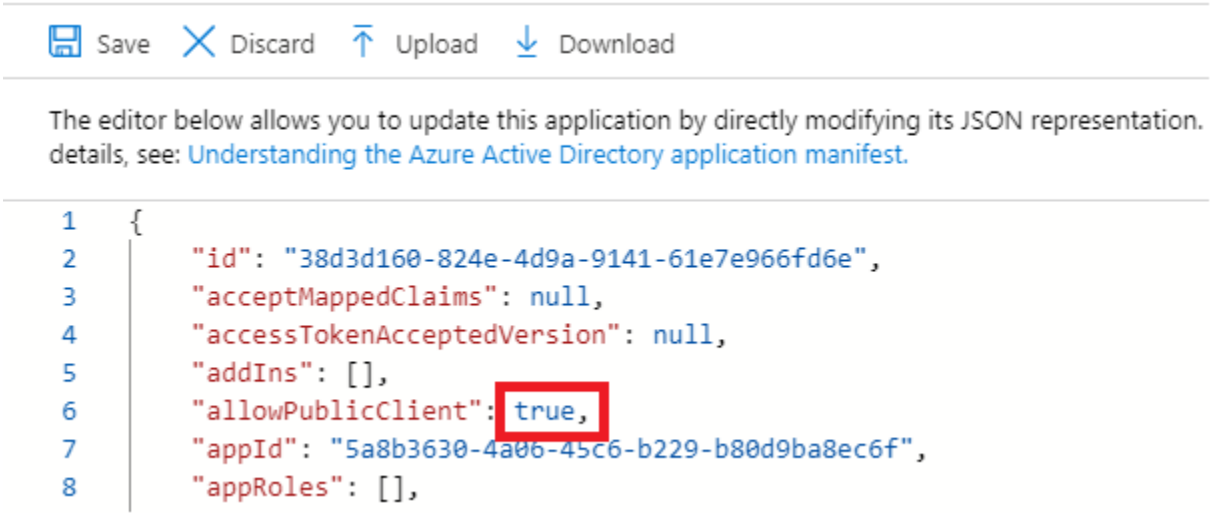
Whether you chose to use the "Delegated permission" (Azure userid and password) or "Application permission" ("application secret"), you need to update the Azure Application "Manifest" as described below.

Click the "Manifest" button on the left side of the page as shown in the screenshot below.

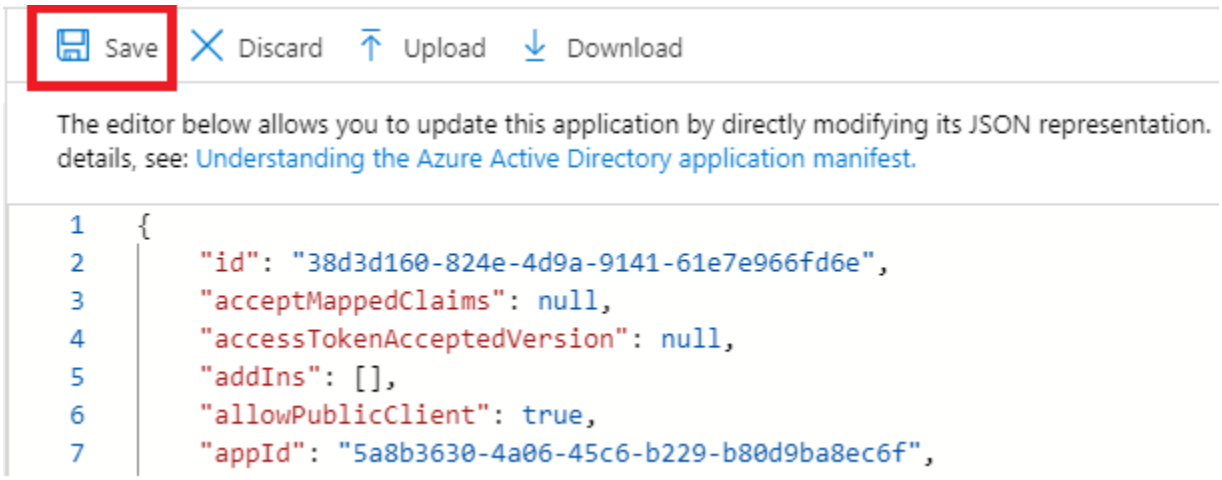


Change the word "null" on the line with "allowPublicClient" to "true" as shown in the screenshots below.





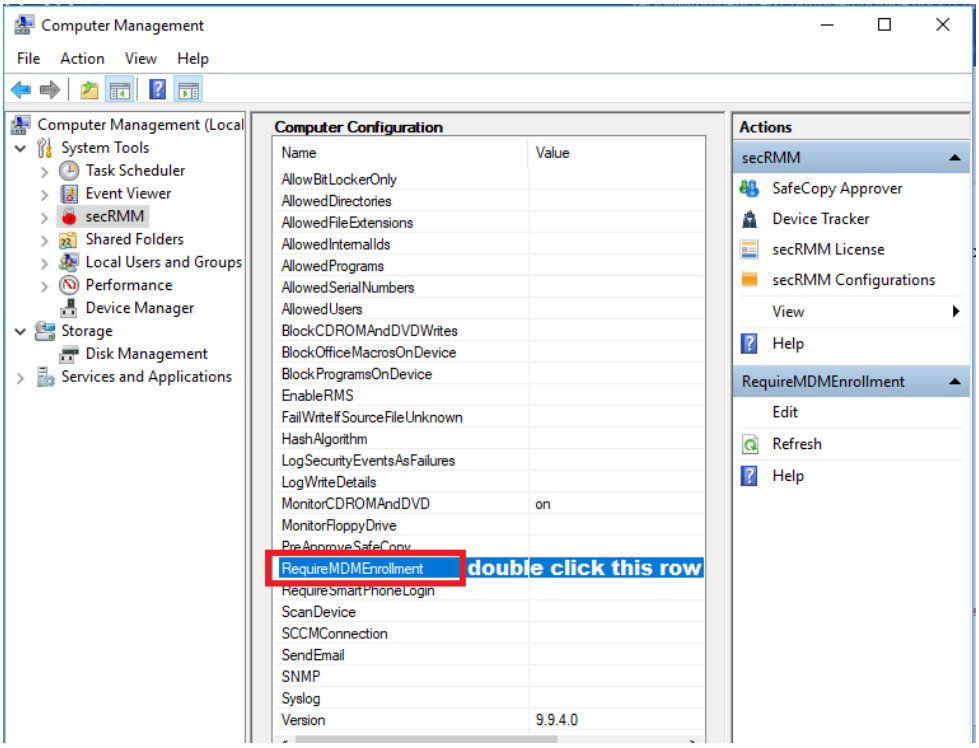
Click the "Save" button as shown in the screenshot below.



Note that if you do not change the "allowPublicClient" to "true" in the Manifest, when you do a "Test Connection" in the secRMM console, you will get the error:

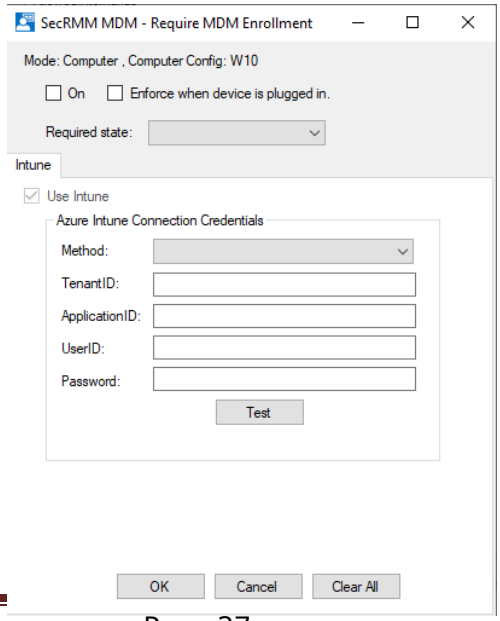
```
Error: AADSTS7000218: The request body must contain the following parameter: 'client_assertion' or
'client_secret'.
```

Configure secRMM to get mobile device state



Now that we have setup where secRMM will get the mobile device state, we will give secRMM the information about how to get the mobile device state data. secRMM can be centrally managed with either System Center Configuration Manager (SCCM) or Azure Intune or Active Directory Group Policy Objects (AD GPO). You can also individually manage a single computer using the Windows "Computer Management" MMC interface (a good tool when you are testing…before you deploy a policy to your entire environment).

Regardless of which interface you use to configure secRMM, there is a secRMM property named "RequireMDMEnrollment". Double click the "RequireMDMEnrollment" row to open the window that lets you configure the Intune connection. As you can see in the screenshot, there are several options available to you. We will break down each option below.

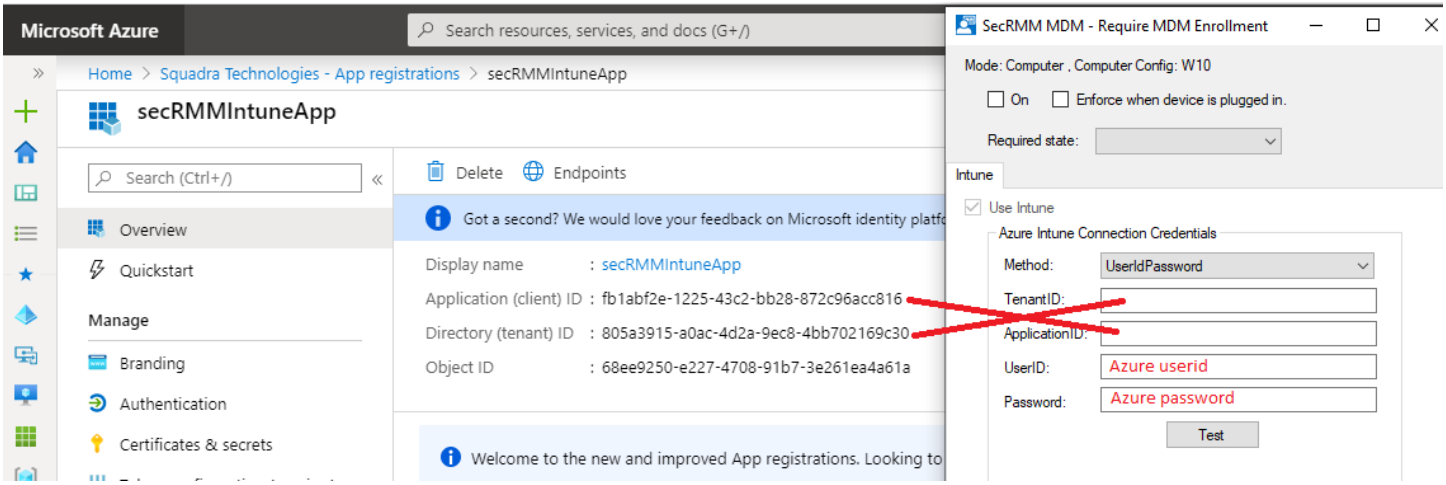


The first checkbox (labeled "On") is required to be checked. It is here to be consistent with all of the other secRMM on/off properties. The second checkbox (labeled "Enforce when device is plugged in.") will make secRMM communicate with Intune as soon as the end-user connects the mobile device using the USB cable. If you do not check this checkbox, secRMM will enforce the rule when an end-user tries to transfer a file to the mobile device. Next, is the drop-down listbox (labeled "Required state"). There are two options: Enrolled and Compliant. This tells secRMM what state is required to allow the mobile device to be used over a USB connection. The compliant state is the most strict since the mobile device must be both enrolled and compliant at the same time. It is not possible for the device to be compliant if it is not enrolled.

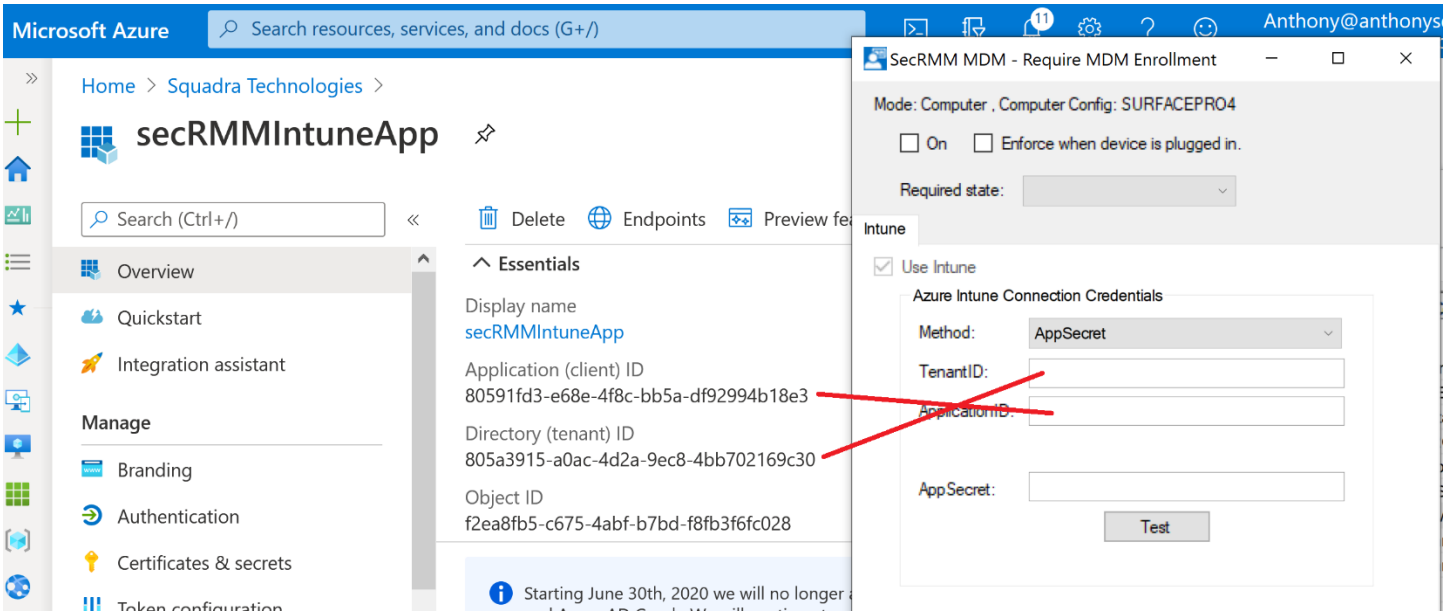
Next, you need to specify the "Azure Intune Connection Credentials" Method.
Select the "UserIdPassword" method if you configured the Azure application permission for "Delegation".
Select the "AppSecret" method if you configured the Azure application permission for "Application".

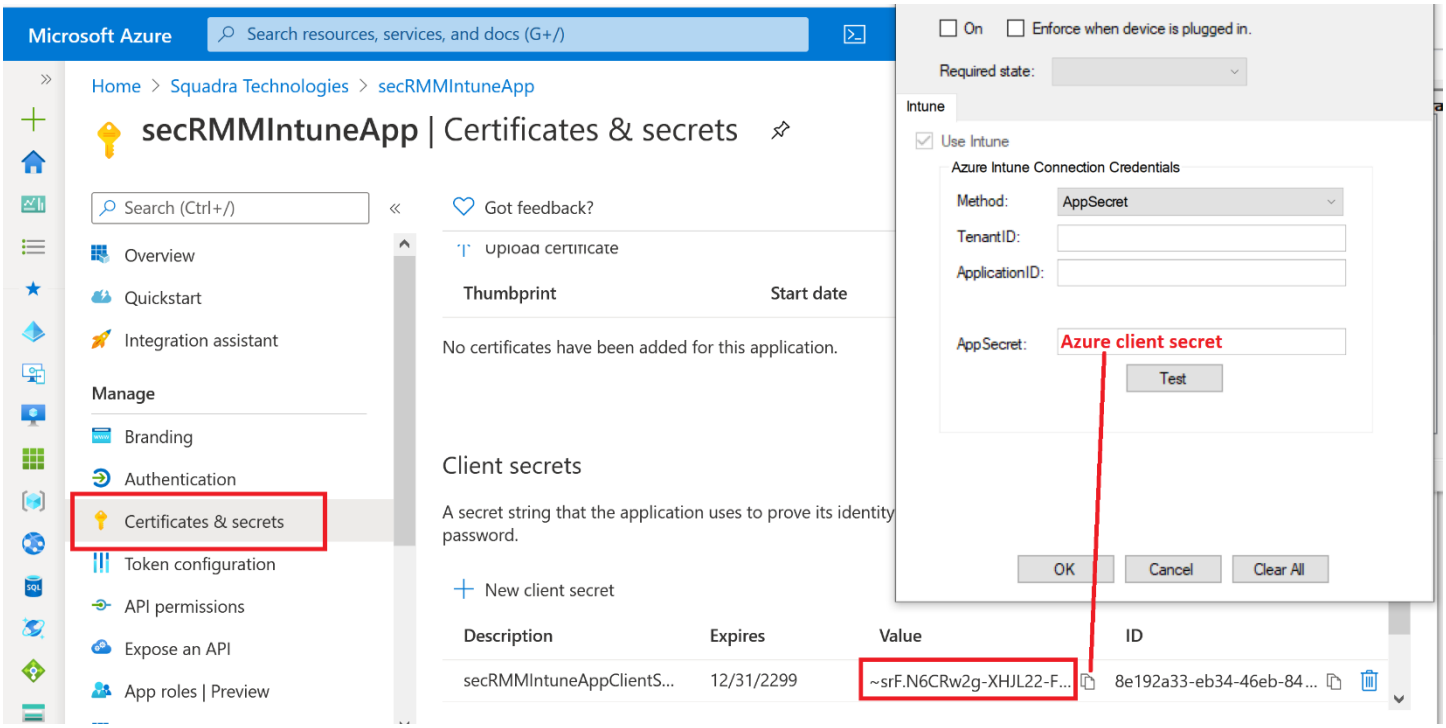
Both methods are shown in the screenshots below.

Delegated permission (userid/password)



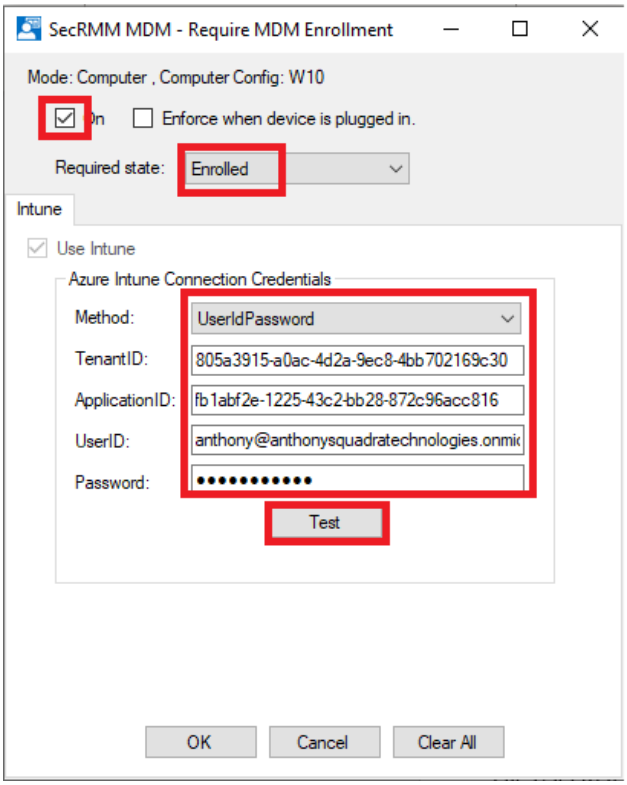
Application permission (client secret)

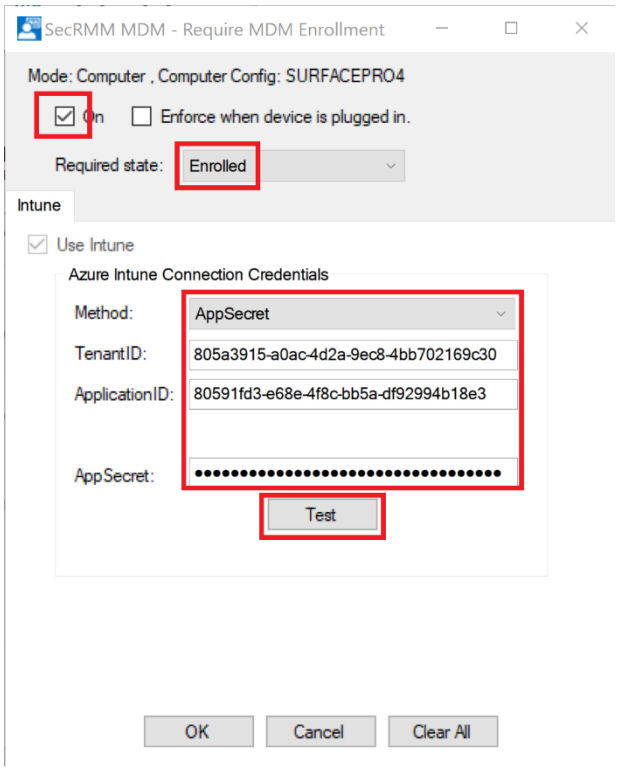


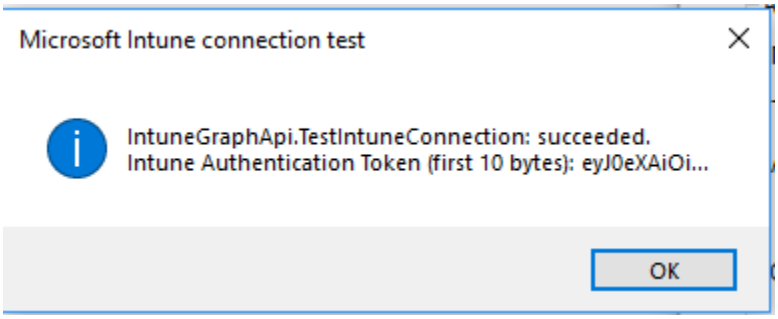


Test the configuration

Once you have all the required properties filled in, click the "Test" button to make sure the connection succeeds. A success message will look like the screenshot below. If the test is unsuccessful, you can look in "C:\Program Files\secRMM\AdminUtils\MDM\Intune\secRMMMDMIntune.log for trace where you can see the detailed errors.



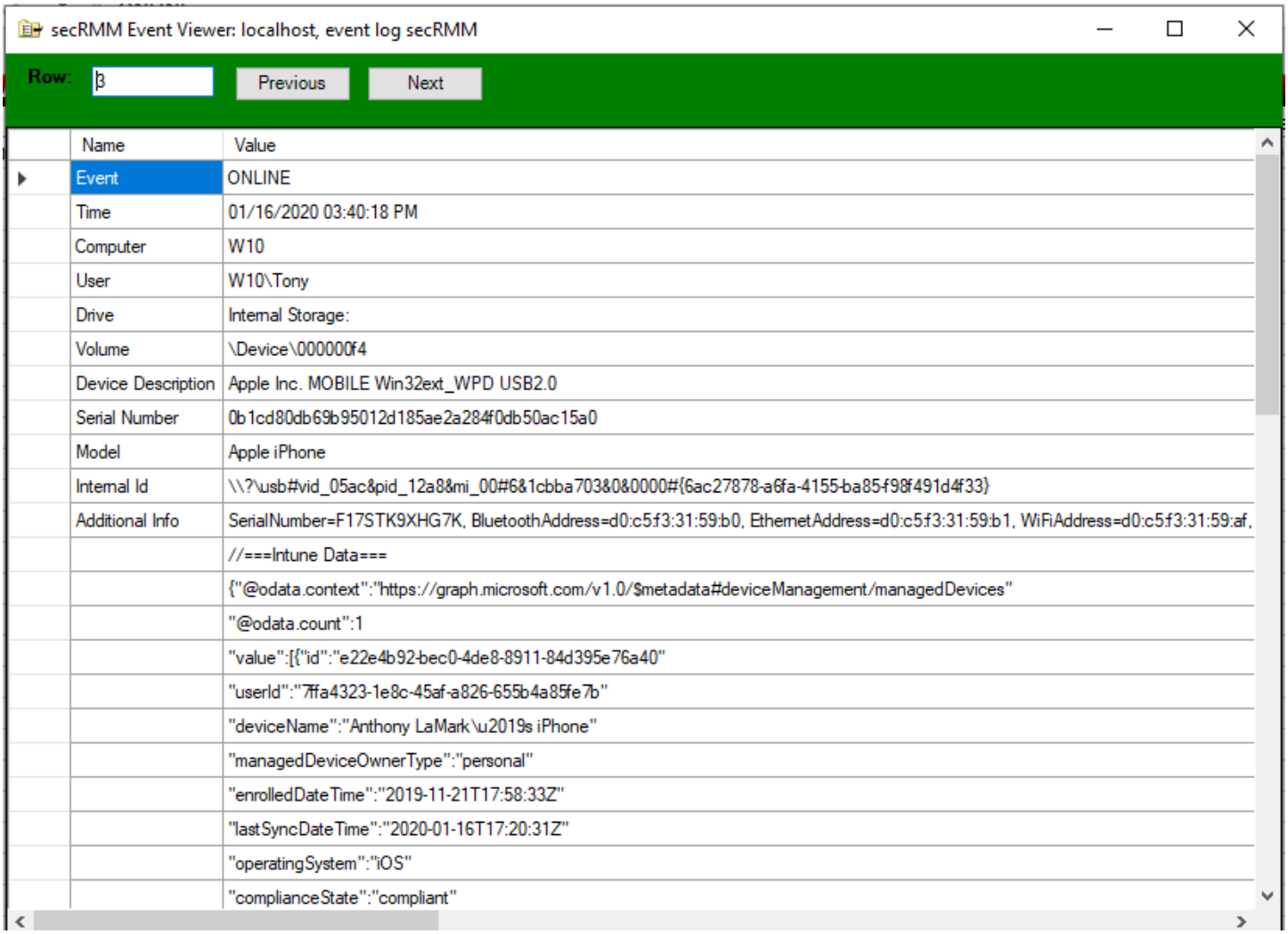




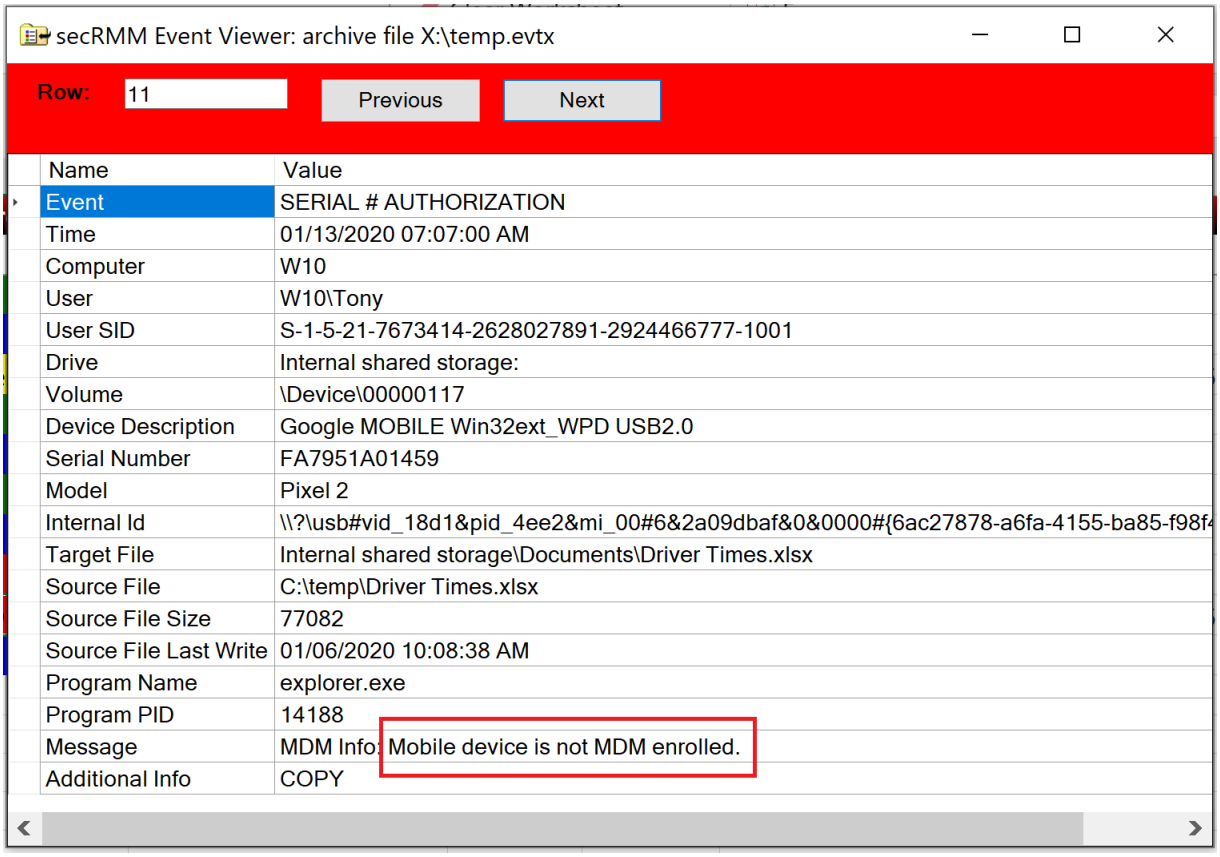
Once you have a successful connection, save the secRMM values by clicking the OK button on the "Require MDM Enrollment" dialog.

Event Data

The secRMM event log will collect data from Azure Intune when mobile devices come ONLINE (over a USB connection) as shown in the screenshot below.



Below is a screenshot of an end-user trying to copy a file (but the file copy was unsuccessful) to the mobile device over a USB connection where the mobile device was not Intune enrolled.



Contacting Squadra Technologies Support

Squadra Technologies Support is available to customers who have purchased a commercial version of secRMM and have a valid maintenance contract or who are in a trial mode of the product.

When you contact Support please include the following information:

1. The version of secRMM you have installed.
2. The Windows versions you have installed: XP, 2003 Server, 2008 Server R2, Vista, Windows 7, etc.
3. Whether the Windows Operating System is 32bit or 64bit.
4. The specific issue you are contacting support for.

About Squadra Technologies, LLC.

Squadra Technologies delivers innovative products that help organizations get more data protection within the computer infrastructure. Through a deep expertise in IT operations and a continued focus on what works best, Squadra Technologies is helping customers worldwide.

Contacting Squadra Technologies, LLC.

Phone 562.221.3079 (United States and Canada)

Email	info@squadratechnologies.com
Mail	Squadra Technologies, LLC. World Headquarters 7575 West Washington Ave. Suite 127-252 Las Vegas, NV 89128 USA
Web site	http://www.squadratechnologies.com/