Comprehensive Customer Setup Instructions

for

CI Synchronizer (Enterprise Edition) for Multi-Source Agent to ServiceNow

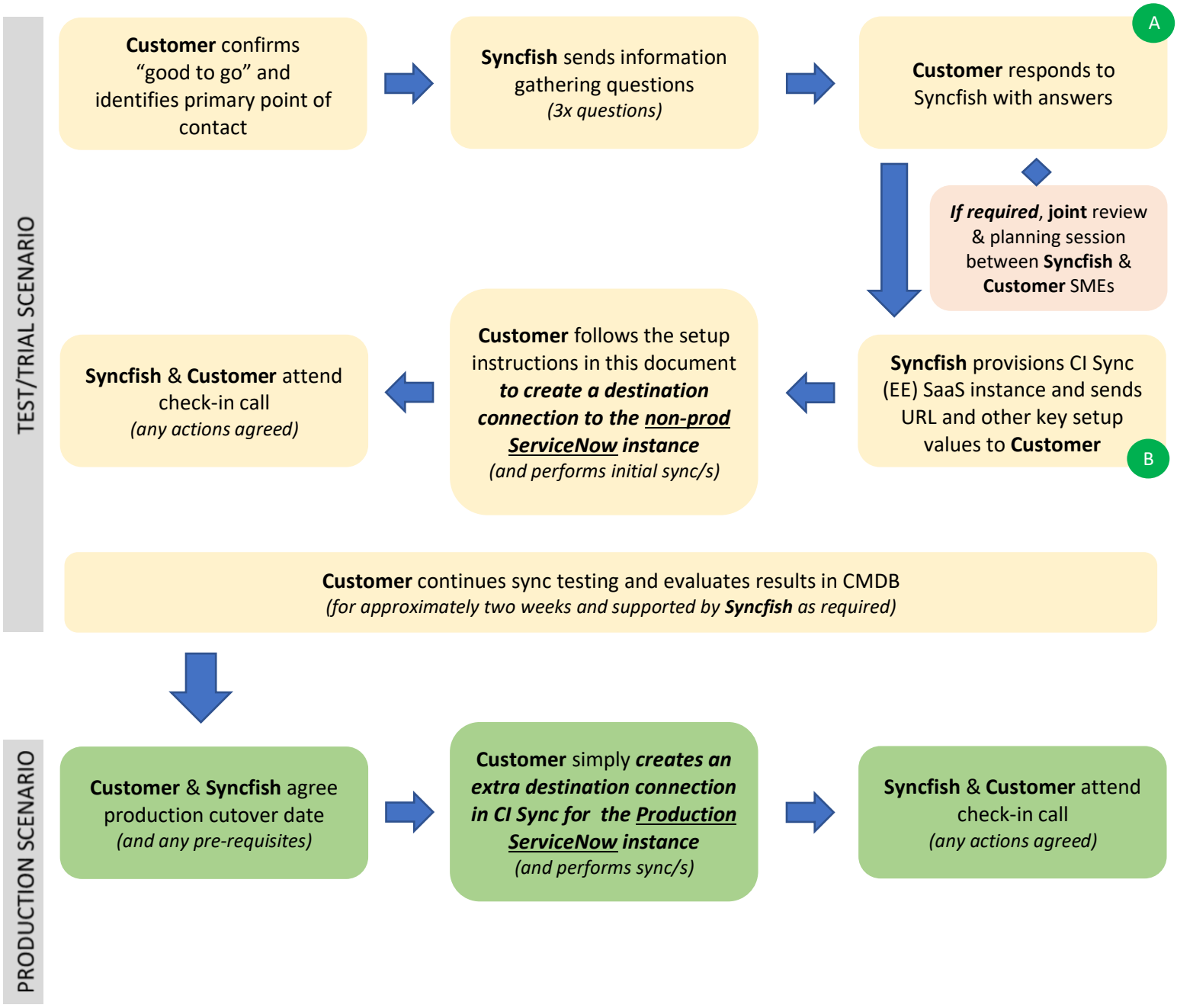
Last Updated: 27 March 2024

Table of Contents

CI Synchronizer (Enterprise Edition) for Multi-Source Agent to ServiceNow
Comprehensive Customer Setup Instructions

Part A - Overall Process after Ordering CI Sync (EE) (Simple Process Diagram)

This diagram shows the overall process of implementing CI Sync (EE) into your environment after you have completed the ordering process with Syncfish.



Guidance Note: The above process can **run in parallel with the ordering** process AND The **duration** between **A and B can be as short as 24 to 48 hours**.

Part B – TEST vs PROD context when using these instructions

For a brand-new implementation (i.e. new Syncfish customer) Syncfish will provision the single instance of CI Sync (EE). These instructions are therefore suitable for setup of a **single instance of CI Sync (EE)**, and the associated components, for use with the following source and destination connections:

1. One Multi-Source Agent will provide connections to source systems for both TEST and PROD data
2. Two ServiceNow instances as the destination CMDBs (i.e. one destination connection in CI Sync (EE) for the TEST ServiceNow CMDB and a second destination connection in CI Sync (EE) for the PROD ServiceNow CMDB).

For initial testing (or for a trial) we anticipate our customers will setup source and destination connections in their CI Sync (EE) instance as follows:

1. A source connection in CI Sync (EE) to their production/live source system.
2. A destination connection in CI Sync (EE) to their TEST (or DEV) ServiceNow instance.

The above configuration would normally be in place for approximately two-weeks to ensure the source asset data is being reflected as expected within the TEST ServiceNow CMDB. During this period Syncfish will assist with any diagnosis or troubleshooting and will discuss any proposed variations to the default correlation, mapping and transform rules within the CI Sync (EE) SaaS engine.

Once testing is complete, we anticipate our customers will setup a second destination connection in CI Sync (EE) to their PROD ServiceNow instance and continue to use the same source connection to their production/live source system instances (i.e. no change).

Part C – Index of Installation Steps

Step #	Additional Details	Section Link	SME Audience
Step 1	Review the Pre-Installation Checklist	Here	• All SMEs
Step 2	Enrol the CI Sync (EE) SaaS application into your AAD	Here	• AAD SME
Step 3	Create an AzureAD App Registration for CI Sync (EE) Agent Authentication	Here	• AAD SME
Step 4	Install the Multi-Source CI Sync (EE) Agent (on the relevant server)	Here	• Infrastructure SME • On-Prem AD SME[#1]
Step 5	Use the CI Sync (EE) Agent Config Utility to setup one or more Source System Connections	Here	• Source System SME • AAD SME[#2] • SQL DBA[#3]
Step 6	Configure your ServiceNow to be ready for CI Sync (EE)	Here	• ServiceNow SME
Step 7	Add your ServiceNow destination connection	Here	• Primary CI Sync (EE) User
Step 8	Run your first synchronization and then progressively run more syncs	Here	• Primary CI Sync (EE) User • ServiceNow SME • Source System SME
Other recommended sections to read			
Appendix A	Understanding how the CI Sync (EE) Agent authenticates to SQL Server during normal operation of the Agent.	Here	• Infrastructure SME • SQL DBA
Appendix B	Configure SQL Maintenance Plans on SQL database(s) for SQL based Source Systems	Here	• SQL DBA

[#1] On-Prem Active Directory SME only required if your SQL server is hosted separately (remotely) from the server that will run the CI Sync (EE) Agent.

[#2] AAD SME is required if you intend to use synchronization source connections for cloud hosted products which require an AAD App Registration (service principal) for authentication.

[#3] A SQL DBA is required if the person performing the CI Sync (EE) Agent installation does not have SQL sysadmin rights on the relevant SQL server.

Step 1 - Review the Pre-Installation Checklist Items

- Checklist Item 1: Make sure you have the CI Sync SaaS Instance and CI Sync Agent install input values.
- Checklist Item 2: Decide on the ServiceNow authentication type.
- Checklist Item 3: Identify the Server that will run the CI Sync (EE) Agent.
- Checklist Item 4: Check HTTPS access is enabled for the Server that will run the CI Sync (EE) Agent.
- Checklist Item 5: Confirm the SMEs are available to perform their individual steps in these instructions.
- Checklist Item 6: Review the SQL Server Edition and Version.
- Checklist Item 7: Review the CI Sync (EE) Lansweeper to ServiceNow - Default Configuration Overview document (applicable to customers using a Lansweeper Source System).

Checklist Item 1: Make sure you have the CI Sync SaaS Instance and CI Sync Agent install input values.

You will need the following:

- The **URL to your company specific instance of the CI Sync (EE)** SaaS application.
- The values needed during the CI Sync (EE) Agent installer are as follows:
  - Your **Azure AD Tenancy ID** (a GUID value from your Azure AD Administrator)
  - The Syncfish **Extractor API URL** (a URL provided to you by Syncfish)
  - The Syncfish **CI Sync (EE) App ID** (a GUID provided to you by Syncfish)
  - The **CI Sync (EE) Instrumentation Key (Customer App Insights/Key)** (a GUID provided to you by Syncfish)

Checklist Item 2: Decide on the ServiceNow authentication type.

Decide which authentication type will be used from the CI Sync (EE) SaaS application to ServiceNow.

- You can use **any one of** Basic Auth, OAuth or MFA.
- The setup steps for all three types are covered in this document. Knowing which one you intend to use will make your installation process easier.

Click here to return to steps index

Checklist Item 3: Identify the Server that will run the CI Sync (EE) Agent.

You need to decide which Server will run the CI Sync (EE) Agent. Within this document, we refer to this as the **CI Sync (EE) Agent Server**.

In most cases this will be an existing server in your environment (i.e. the CI Sync (EE) Agent does not require a dedicated server).

The **CI Sync (EE) Agent can run on any Windows Computer** in your environment **which meets the following criteria:**

- The server **has HTTPS access to the Internet** (at least to the CI Sync (EE) SaaS application URL).
- **If** the **CI Sync (EE) Agent** will be **reading source data from a SQL database** (e.g. for sync sources such as Lansweeper or SCCM) then the **CI Sync (EE) Agent Server should be is geographically near the source system SQL** database (i.e. should be on the same LAN).
- In a **Lansweeper environment** Syncfish make the following general recommendations for **where to run the CI Sync (EE) Agent**:
  - If your Lansweeper Console and Lansweeper SQL DB are the same Server, install the CI Sync (EE) Agent on that same VM (all three can exist together).
  - If your Lansweeper Console and Lansweeper SQL DB are on separate Servers, don't install the CI Sync (EE) Agent on the SQL Server (DBAs don't tend to like this model). Instead, it's fine to install the CI Sync (EE) Agent on your Lansweeper Console Server.

Checklist Item 4: Check HTTPS access is enabled for the Server that will run the CI Sync (EE) Agent.

Verify the **CI Sync (EE) Agent Server** has access to the Internet over HTTPS.

- You should test HTTPS access to the internet from the console of the server. Ideally by testing as the same Windows Service Account the CI Sync (EE) Agent will be logged in as.
- You need to ensure no proxy, firewall or GPO restrictions are in place.

Click here to return to steps index

Checklist Item 5: Confirm the SMEs are available to perform their individual steps in these instructions.

Confirm the following SMEs are available throughout the installation.

- A ServiceNow System Administrator.
- An Azure Active Directory (AAD) Administrator.
- An infrastructure SME who can create a Windows Service Account (either local or in Active Directory) for use by the CI Sync (EE) Agent.
- An infrastructure SME who can install the CI Sync (EE) Agent onto the desired server.
- A SQL Server Database Administrator (if the person performing the CI Sync (EE) Agent installation does not have SQL sysadmin rights on the relevant SQL server).
- Make sure the SMEs have sufficient permissions to complete their tasks described in this document.

Checklist Item 6: Review the relevant CI Sync (EE) Default Configuration Overview document.

Syncfish publish a "Default Configuration Overview" document specific to a given Source and Destination System. These guides cover various topics about the CI Sync (EE) Default Configuration rules, and in particular how the configuration rules will read, map, transform and interact with data and objects between a given Source System and Destination System. If you haven't already been provided with the relevant Default Configuration Overview document, please contact Syncfish.

Click here to return to steps index

Step 2 – Enrol the CI Sync (EE) SaaS application into your AAD

In this step your Azure Active Directory Admin SME will perform the following tasks

- Task 2a: Enrol the CI Sync (EE) SaaS application as an Enterprise Application in your AAD.
- Task 2b: Grant users access to the CI Sync (EE) SaaS application (so they can use the User Interface).

Task 2a: Enrol the CI Sync (EE) SaaS application as an Enterprise Application in your AAD.

Context Note: In this section your customer specific instance of the CI Sync (EE) SaaS application will enrolled into your organisation's Azure Active Directory (AAD) as an Enterprise Application.

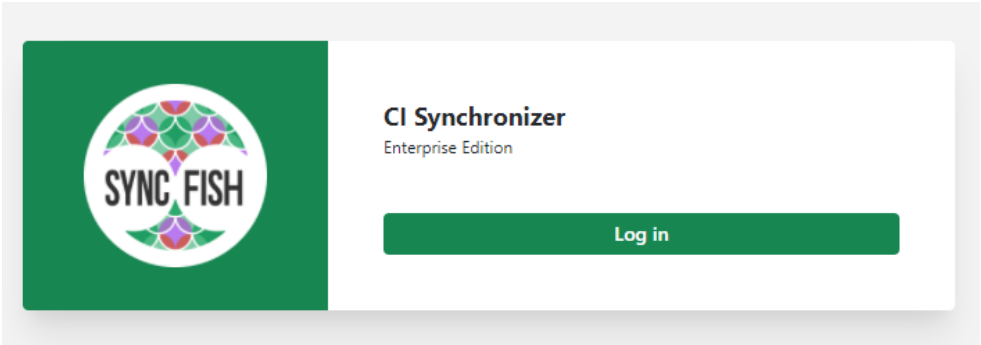
The steps must be performed by an AAD SME with sufficient rights to create and maintain a new Enterprise Application.

By the time your AAD SME commences these steps, Syncfish will have provided you with your company specific URL. The URL will be https://***YourCo***.syncfish.app where "**YourCo**" is your company name or a shorted version.

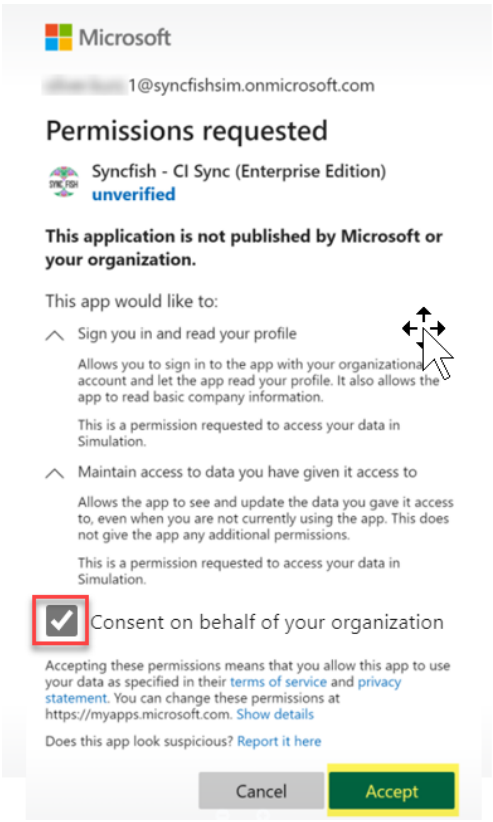
Important: If a non-AAD admin accesses the URL it will initiate the Enterprise Application registration in your AAD and will fail due to not having sufficient AAD permissions. Please do NOT access the URL provided and attempt to login unless you are an AAD SME with sufficient rights to perform all steps below.

1. Your AAD Admin should open a browser and **navigate to the Syncfish provided URL** to your **company specific instance** of the CI Sync (EE) SaaS application User Interface. The URL will be as shown below:
   - https://***YourCo***.syncfish.app

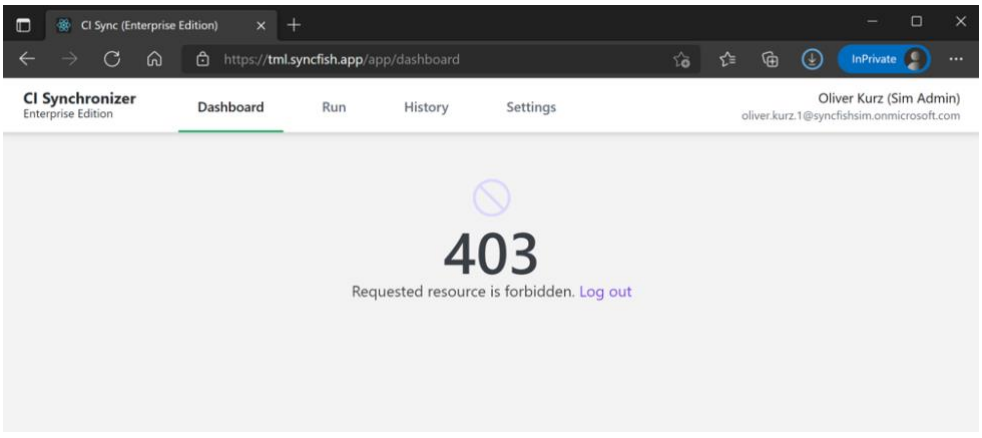
2. **When prompted to sign in**, ensure you login with the AAD Admin account in the same AAD tenancy that you provided to Syncfish.



3. You will be prompted to **Accept the Permissions requested** and to **Consent on behalf of your organisation**. When you click **Accept** it will enrol the CI Sync (EE) SaaS application in your AAD.



4. Once the enrolment is complete, **you will be returned** with a HTTP 403 error "Requested resource is forbidden".

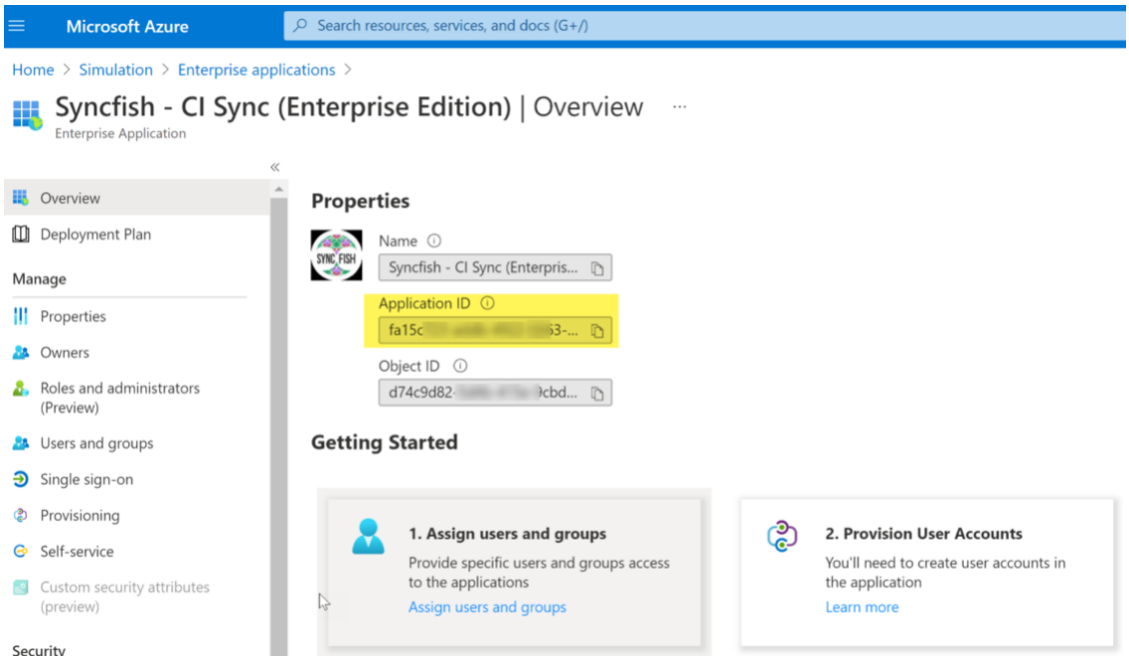


Guidance Note: The 403 error is expected behaviour because the user account (of your AAD Admin) has not yet been granted access to the newly enrolled CI Sync (EE) Enterprise Application.

The steps required to grant individual user access to the CI Sync (EE) SaaS application User Interface are documented below on the subsequent page.

Click here to return to steps index

5. Your AAD Admin should now use the Azure Portal to verify the CI Sync (EE) SaaS application was successfully enrolled as an Enterprise Application in your AAD. To do this:
    a. Login to the **Azure Portal**
    b. Navigate to **Azure Active Directory**
    c. Select **Enterprise Applications**
    d. **Find and select** the newly created CI Sync (EE) Enterprise Application in the list (i.e. "Syncfish – CI Sync (Enterprise Edition)")
    e. From the **Overview** menu item, ensure Properties such as Name, Application ID and Object ID have all be populated.



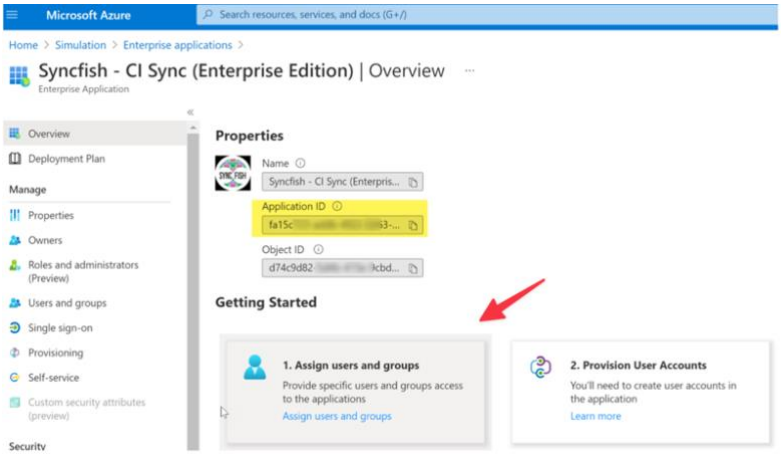
Click here to return to steps index

Task 2b: Grant users' access to the CI Sync (EE) SaaS application (so they can use the User Interface).

Informational Note: In this section your AAD SME will grant those users who will need access to the CI Sync (EE) SaaS application user Interface. In most organisations this is likely to be one or two people only (i.e. those few users expected to create and schedule synchronization jobs via the CI Sync (EE) User Interface).

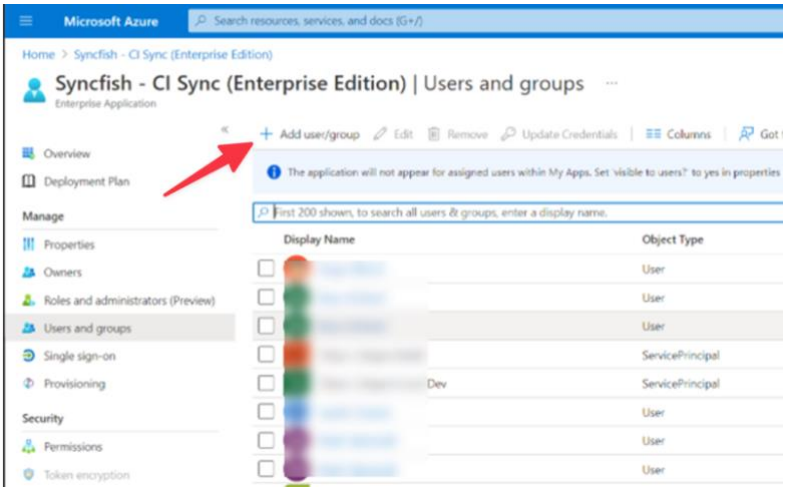
1. In the **Azure Portal**, navigate to **Azure Active Directory** -> **Enterprise Applications** and **select the CI Sync (EE)** application (i.e. "Syncfish - CI Sync (Enterprise Edition)")

2. Click the "**Assign users and groups**" hyperlink under **Getting Started** -> **1. Assign users and groups**.

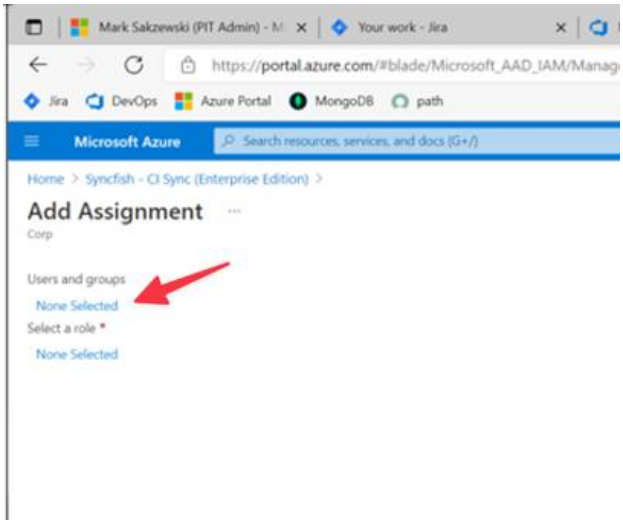


3. There may already be a Role Assigned "**Default Access**" depending on which account enrolled the Enterprise App. This role assignment can be left alone but we still need to add the "administrator" role assignment.

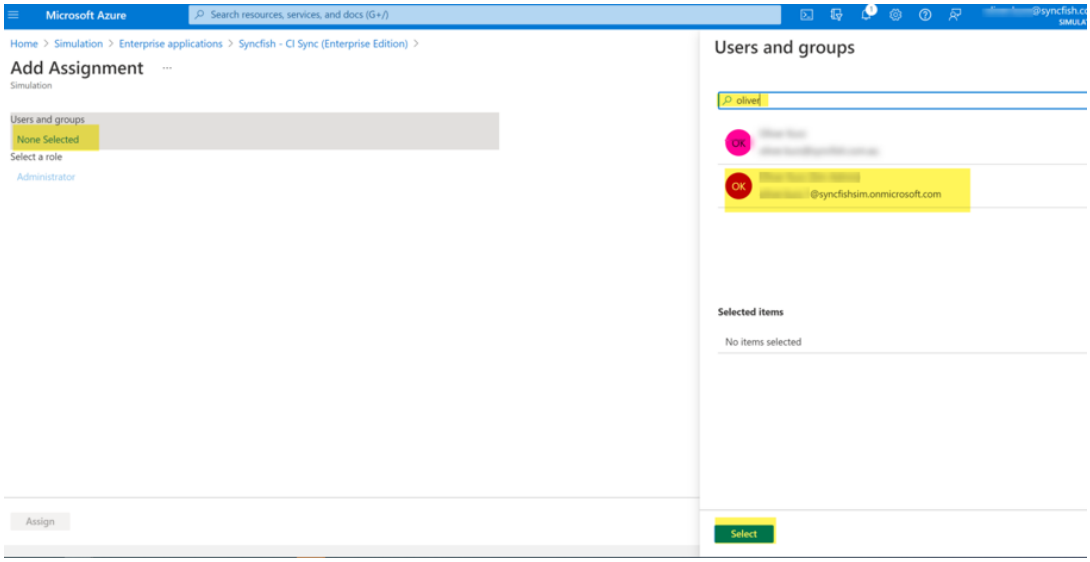
4. Click **Add user/group**



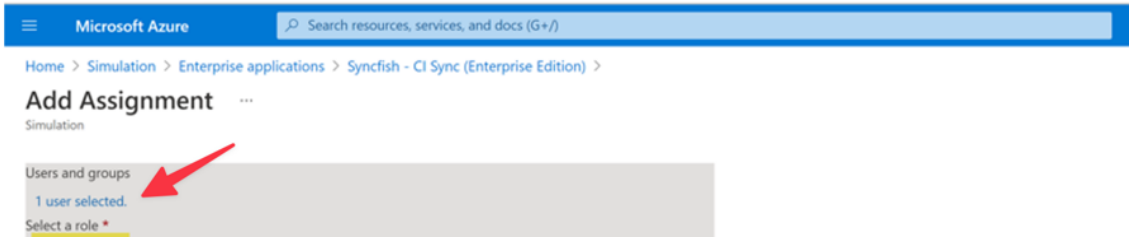
5. Click the **None Selected** link under **Users and Groups**.



6. Use the right-hand pane to **filter/search** and **select** the **user/s** and/or **group/s** you want to grant access to the CI Sync (EE) SaaS application (those few users expected to create and schedule synchronization jobs via the CI Sync (EE) User Interface). When ready, click the **Select** button to complete.



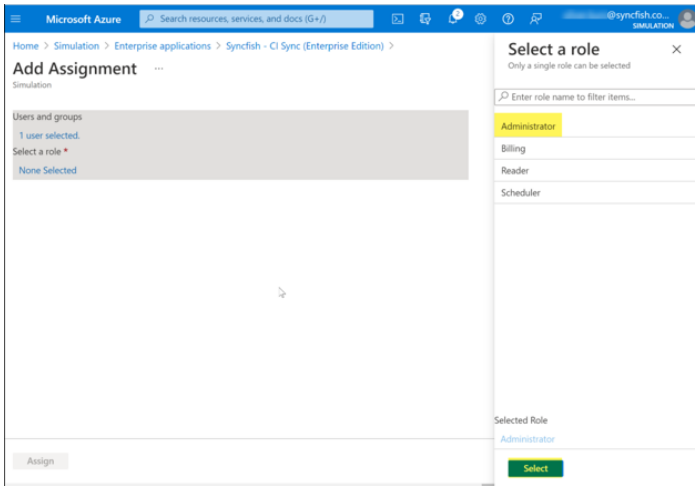
7. You have now granted a user (or a group) access to the CI Sync (EE) User Interface.



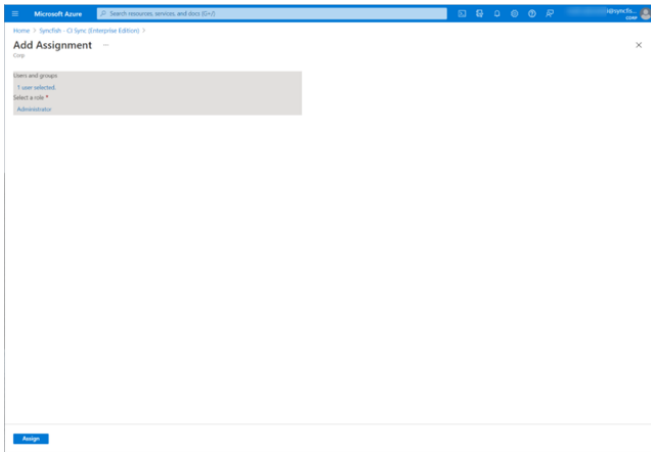
Click here to return to steps index

8. Using similar steps, you now need to grant the CI Sync (EE) Administrator role for those same user/s or group/s (otherwise they will not have sufficient permissions within the CI Sync (EE) SaaS application.
9. Click the **None Selected** link under **Select a role**

10. Use the right-hand pane to select the **Administrator** role. When ready, click the **Select** button to complete.



11. The screen should look as below. When ready, click the **Assign** button at the bottom of the screen.

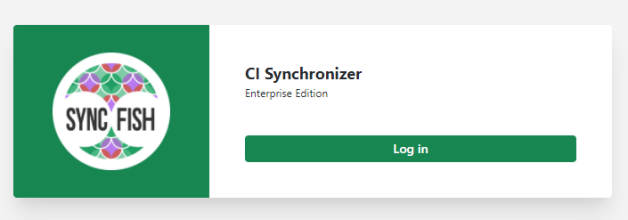


Click here to return to steps index

12. The **users who were granted access** should now be able to login and use the CI Sync (EE) SaaS application User Interface. They can test by **navigating to the Syncfish provided URL** to your company specific instance of the CI Sync (EE) SaaS application User Interface. The URL will be as shown below:

    - https://***YourCo***.syncfish.app

13. **When prompted to sign in**, login with your regular AAD user credentials (including any MFA requirements).



Upon successful login you will be presented with the CI Sync (EE) User Interface.

Click here to return to steps index

Step 3 – Create an AAD App Registration for CI Sync (EE) Agent Authentication

Context Notes:

1. An AAD App Registration object is used to control authentication between the CI Sync (EE) Agent (i.e. the CI Sync (EE) Agent Windows Service) and your organization's Azure Active Directory. This ensures your customer specific CI Sync (EE) SaaS instance will only accept payloads from a CI Sync (EE) Agent that has first authenticated to your organization's Azure AD.
2. CI Sync (EE) supports the following credential types for authentication between the CI Sync (EE) Agent and the App Registration:
    a. Client Secret, or
    b. Certificate
3. The **App Registration created in this section relates to the CI Sync (EE) Agent itself** (i.e. the Windows Service). This CI Sync (EE) Agent App Registration is distinct from any additional App Registrations you may need to create when configuring the CI Sync (EE) Agent to read from sources such as InTune and Azure. The steps for creating one/more App Registrations for data sources such InTune and Azure are described in subsequent sections of this guide.

Informational Note: In AAD an App Registration is used to define a **Service Principal** that will be used to authenticate the CI Sync (EE) Agent when it connects to your customer specific instance of the CI Sync (EE) SaaS application. If you have multiple instances of the same source system (e.g. multiple Lansweeper instances, one for TEST and one for PROD) you will need to create an AAD App Registration for each of the CI Sync (EE) Agent instances you have installed.

In this step your Azure Active Directory (AAD) Admin SME will perform the following tasks:

- Task 3a: Decide which credential type your organization will use for authentication between the CI Sync (EE) Agent against the App Registration that represents the CI Sync (EE) Agent.
- Task 3b (Option 1): Create the App Registration object in your AAD using a Client Secret credential and provide details about the App Registration to the SME performing the CI Sync (EE) Agent installation.
- Task 3b (Option 2): Create the App Registration object in your AAD using a Certificate credential and provide details about the App Registration to the SME performing the CI Sync (EE) Agent installation.

Click here to return to steps index

Task 3a: Decide which credential type your organization will use for authentication between the CI Sync (EE) Agent against the App Registration that represents the CI Sync (EE) Agent.

CI Sync (EE) supports the following credential types for authentication between the CI Sync (EE) Agent and the App Registration:

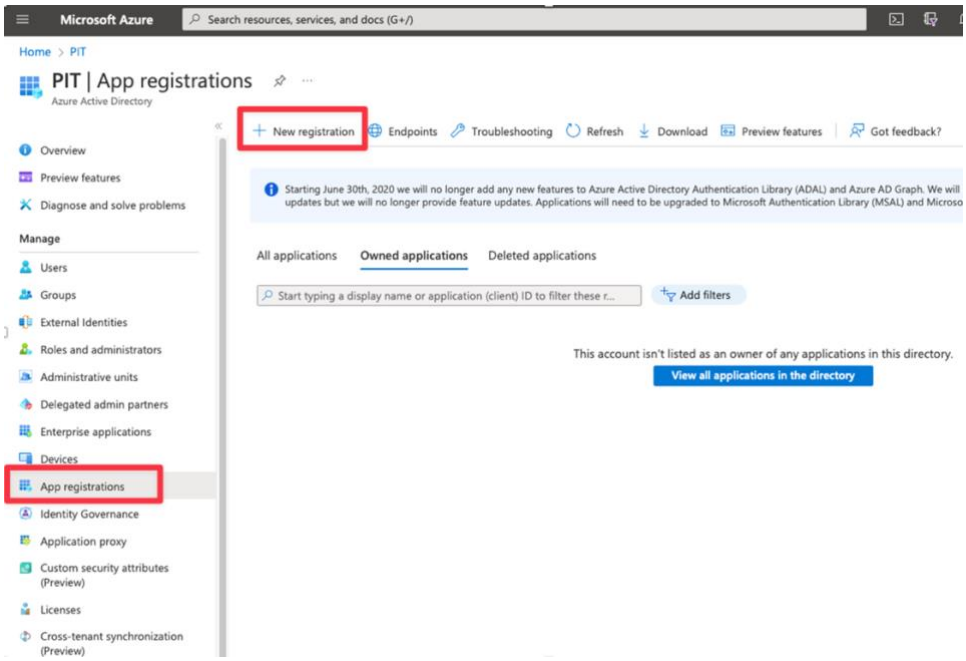
a. Client Secret, or
b. Certificate

Deciding which credential type to use is an organisational decision (often based on your Cyber Security requirements and/or Azure AD administration requirements).

Once you decide with credential type to use, follow the relevant steps in either Task 3b Option 1 or Task 3b Option 2 below.

Task 3b (Option 1): Create the App Registration object in your AAD using a Client Secret credential.

1. In the **Azure Portal**, navigate to **Azure Active Directory** -> **App Registrations** and click **New Registration**



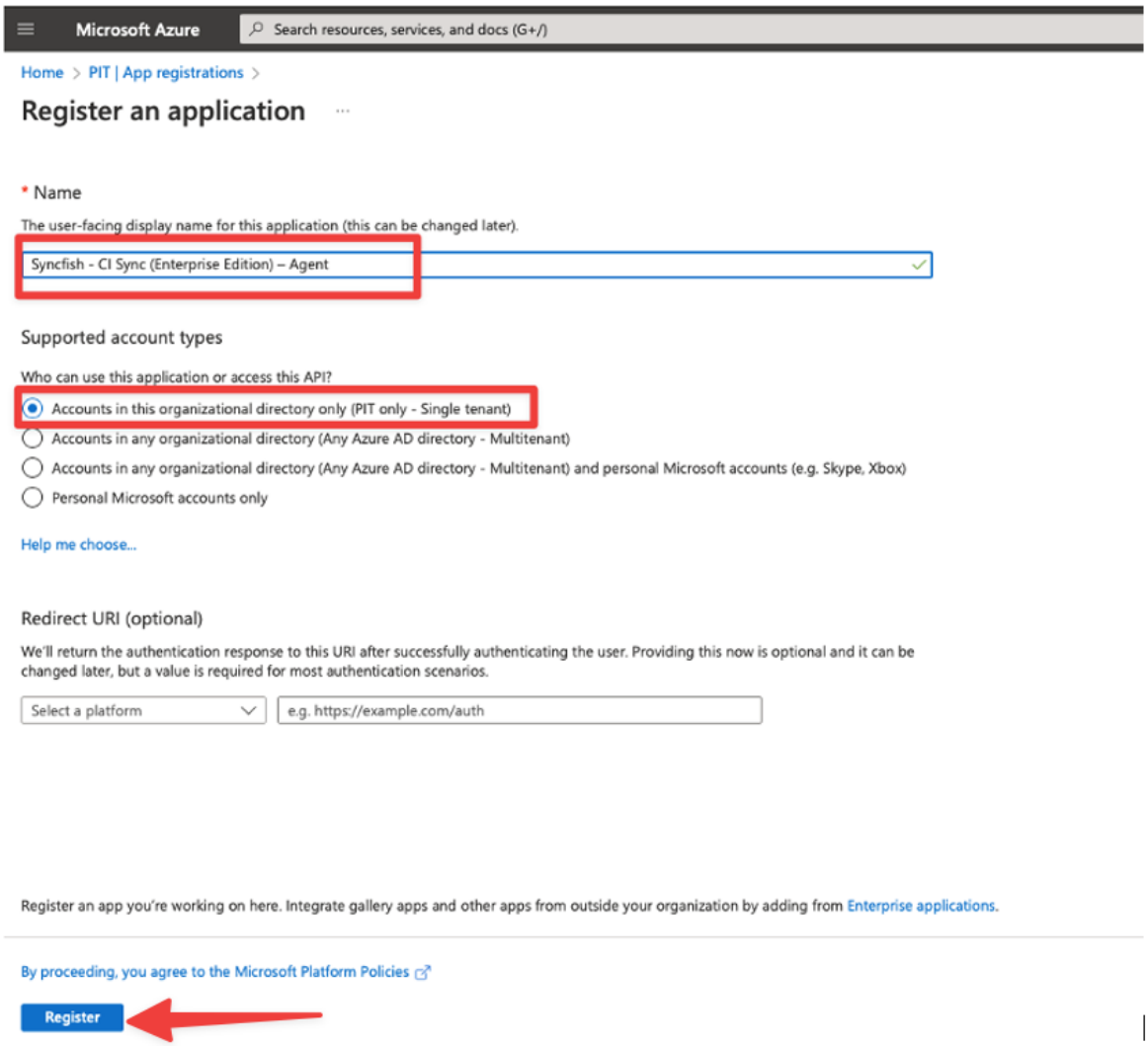
2. On the **Register an application** form complete as follows:

   - Enter the **Name**

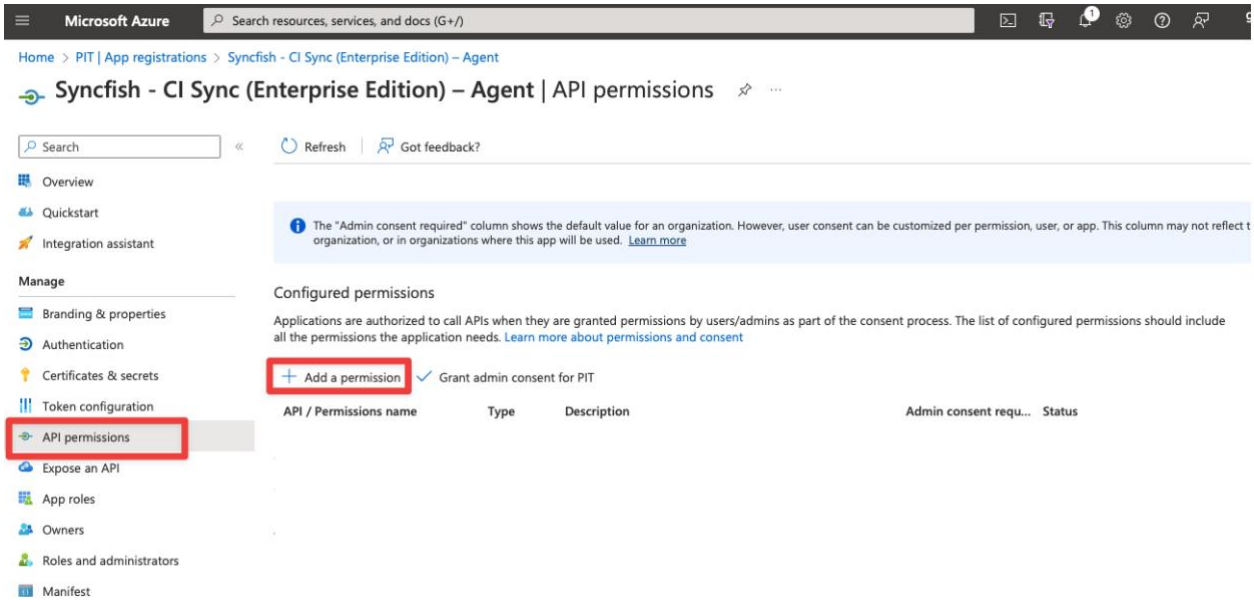
     Guidance Note: Syncfish recommend using "Syncfish - CI Sync (Enterprise Edition) – Agent".

     If you use another name, then **it is highly recommended to include "Agent" in the name** to help distinguish this registration from the CI Sync (EE) SaaS application Enterprise Application registration as AAD will show both in the same list on some forms.

   - Under **Supported account types** select "**Accounts in this organizational directory only (*{Your Domain/Tenant Name}* only - Single tenant)**"
   - Click **Register**

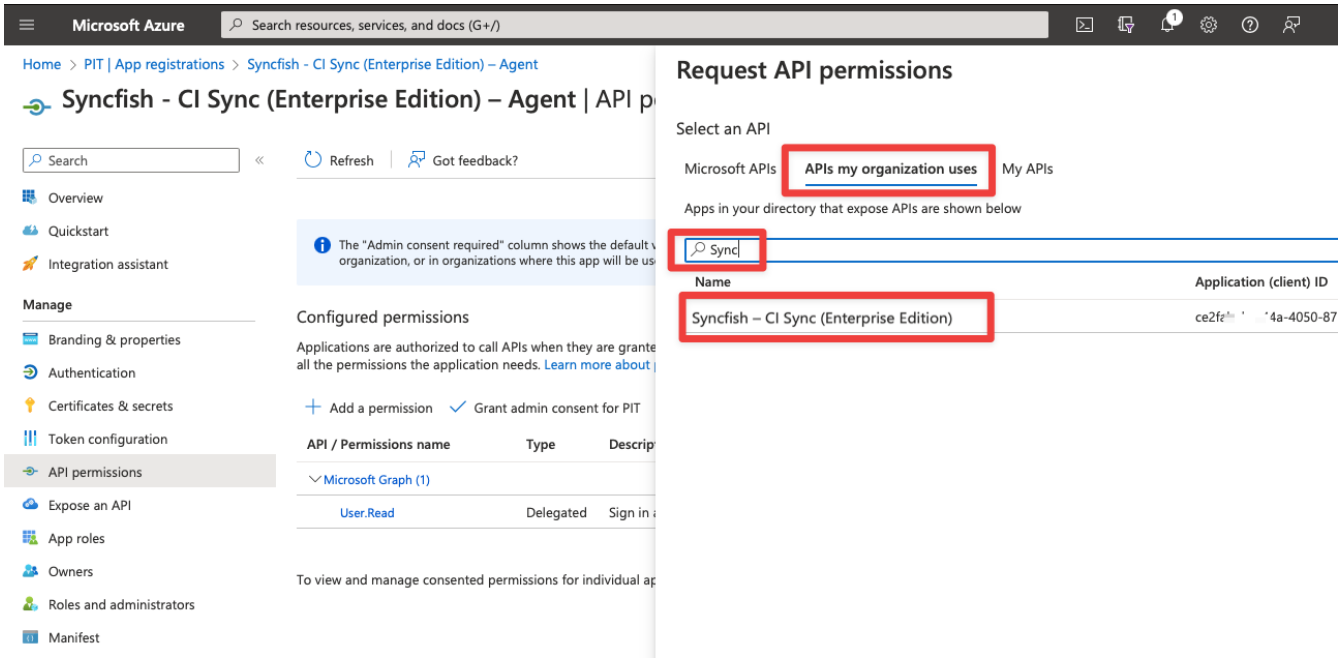


3. Navigate to **API permissions** and click **Add Permission**

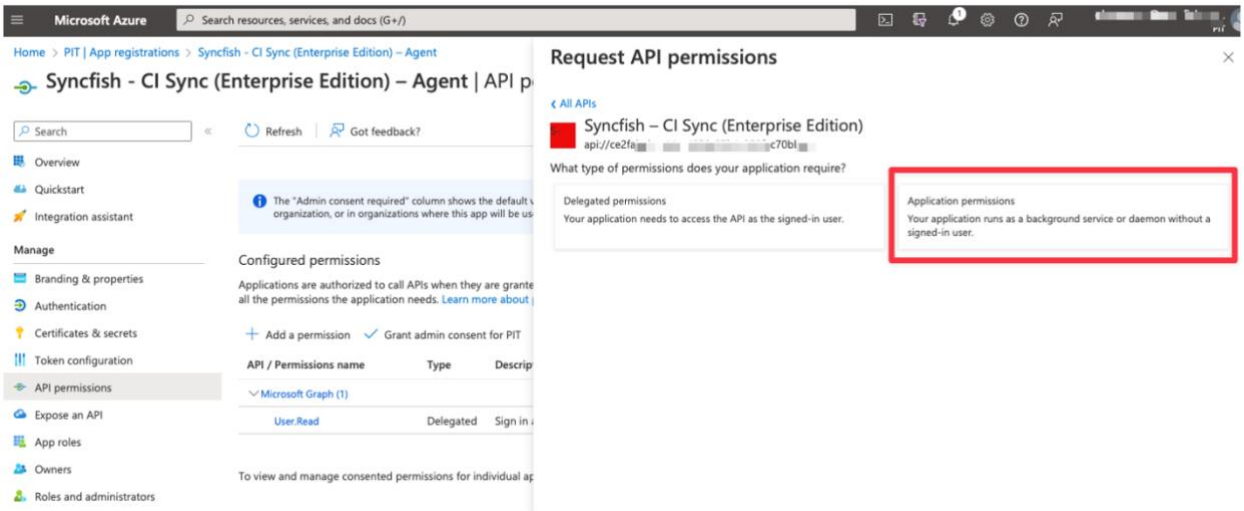


4. Navigate to **APIs my organization uses** and search for Syncfish, and then select the **Syncfish - CI Sync (Enterprise Edition)** entry.

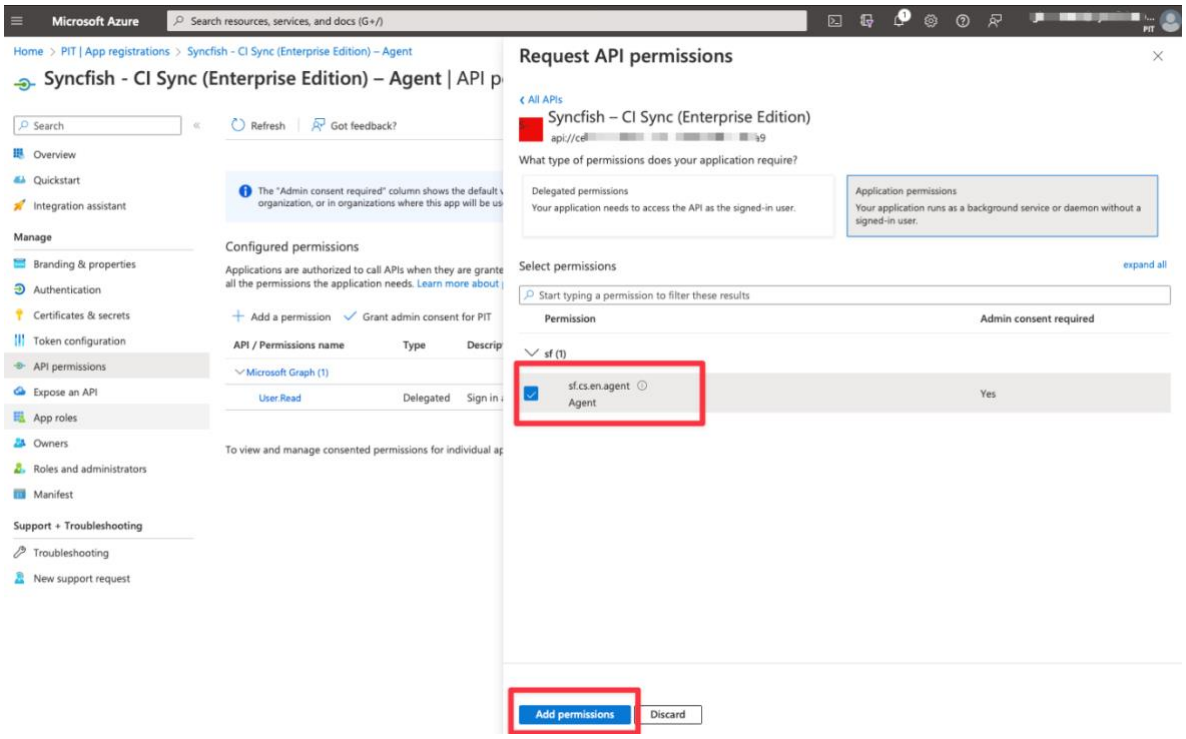
> **Guidance Note:** In the list **you are selecting the CI Sync (EE) <u>Enterprise Application</u>** that was enrolled during Step 2 earlier in this guide.



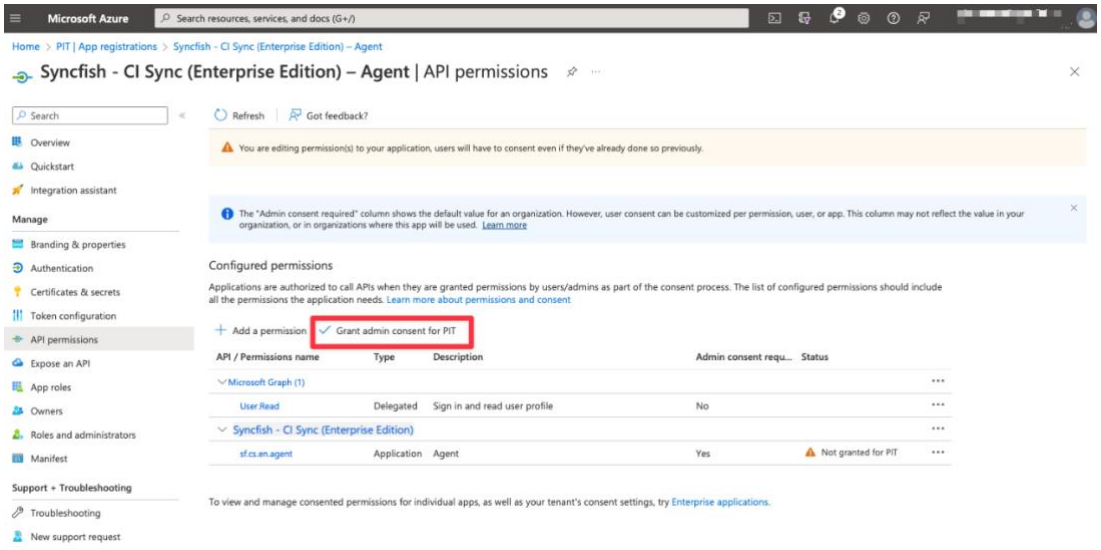
5. Select the "**Application permissions**" button



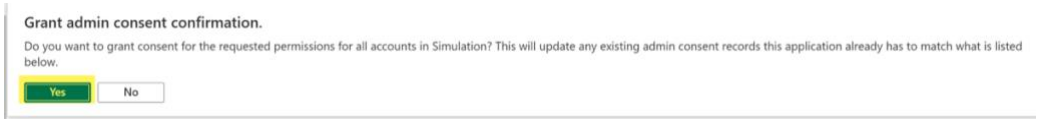
6. Tick the checkbox for **sf.cs.en.agent Agent** and click the **Add permissions** button



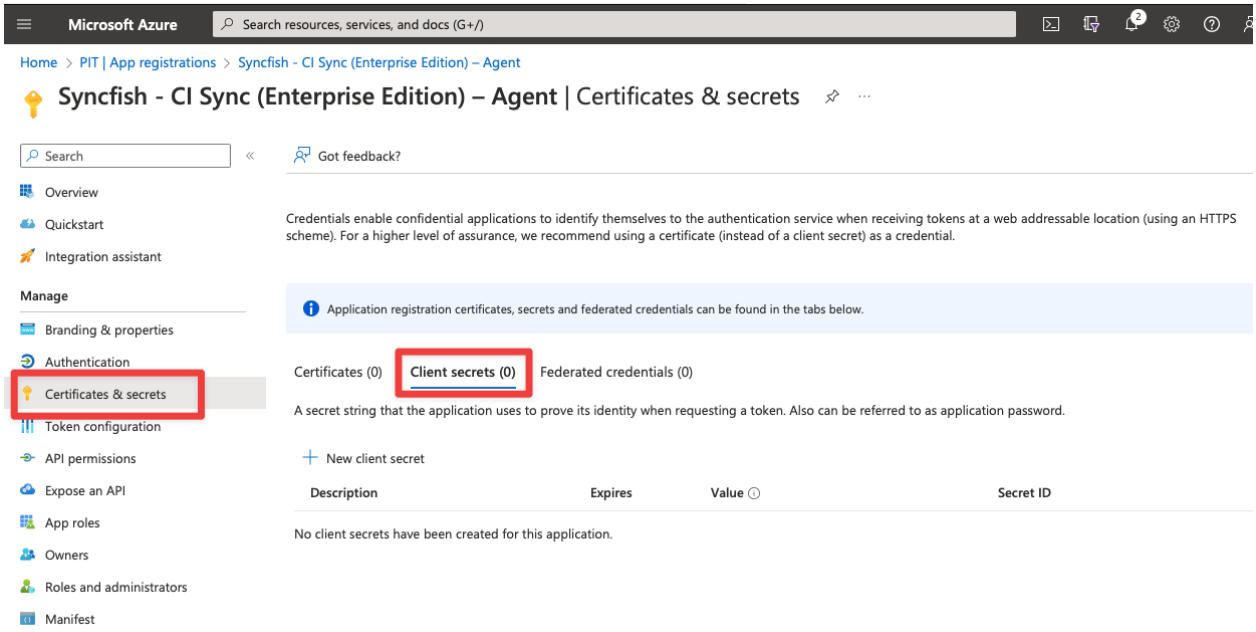
7. Back on the main screen for "API permissions", click the button to "**Grant admin consent for *{Your Domain/Tenant Name}***"



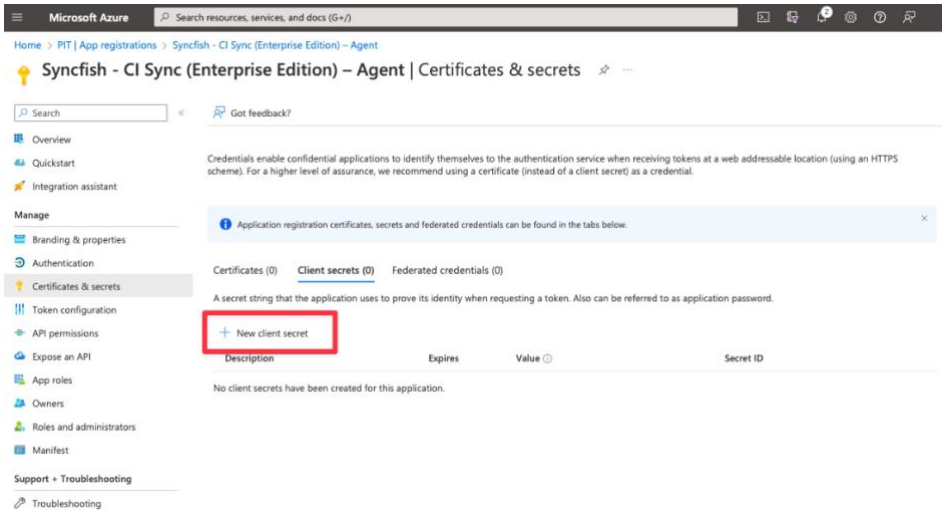
8. Click **Yes** to confirm



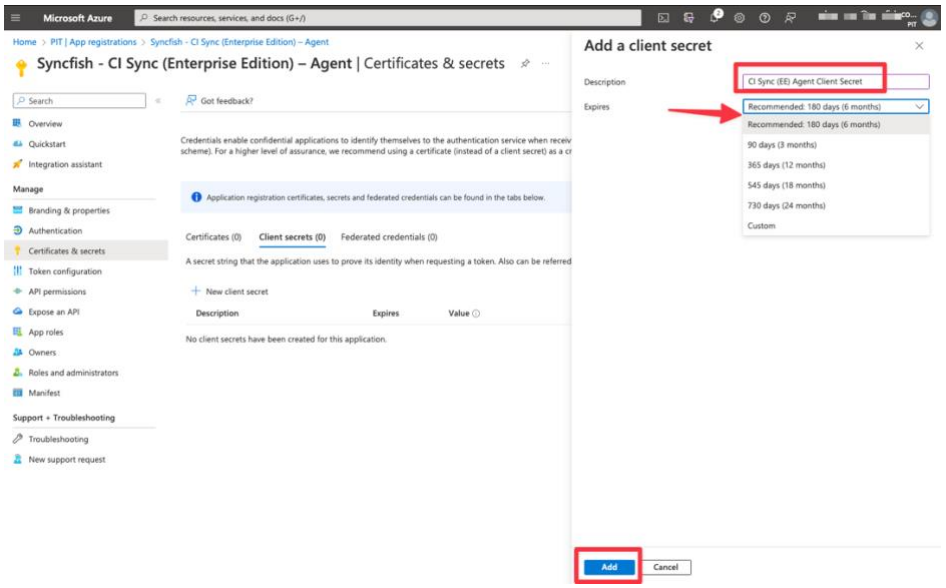
9. Using the left-hand menu, navigate and select **Certificates & secrets**. Select client "**Client secrets (0)**" in the middle of the form.



10. Click the "**New client secret**" button.



11. Enter a unique **Description** for the secret associated with the CI Sync (EE) Agent App Registration.
12. Then, select a suitable **Expires** duration based on your organisational policy. Finally click the **Add** button.



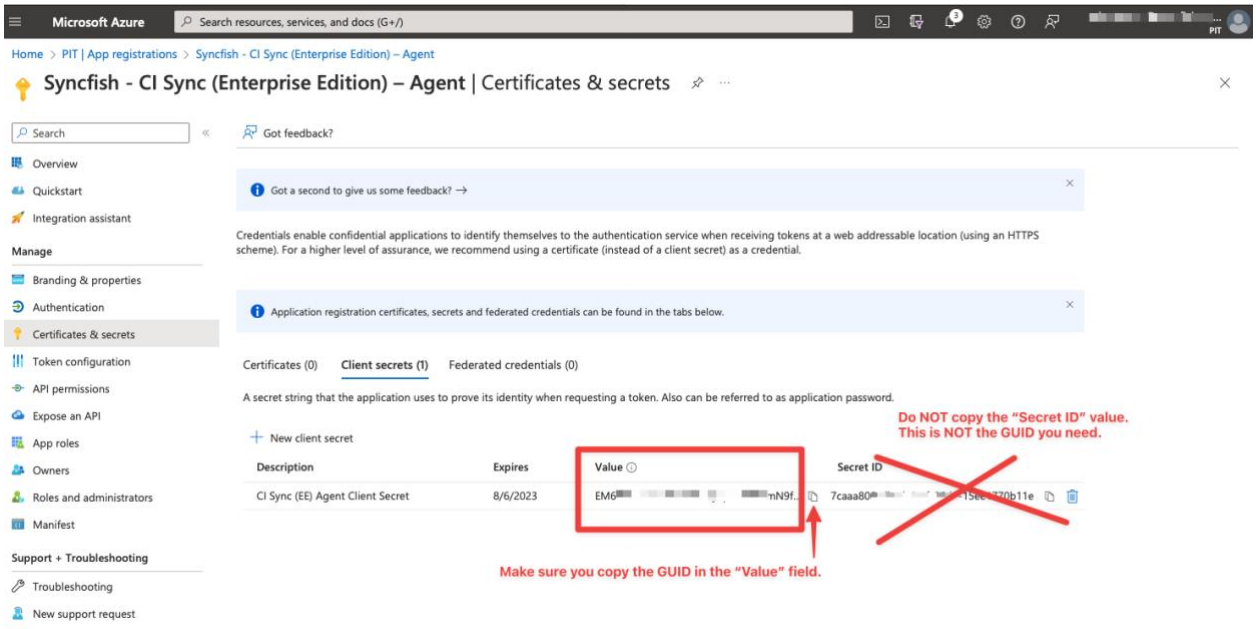
Guidance Note: It is recommended you set a reminder prior to the expiry date of the Secret (i.e. a reminder to regenerate and update the Secret in the CI Sync (EE) Agent configuration. The use of the secret will be clearer once you have read the section on "Install the CI Sync (EE) Agent".

The steps to regenerate the secret and update the secret in the CI Sync (EE) Agent are explained in Appendix C of this document.

13. The form now displays the generated **secret value** (shown in the **Value** field).

> **Guidance Note:** The **Value** is only available while you remain on this screen. You must make a copy of the Value GUID before leaving this form.
>
> Make sure you copy the "Value" and NOT the "Secret ID".

- Use the copy option to make a copy of the GUID in the **Value field.**

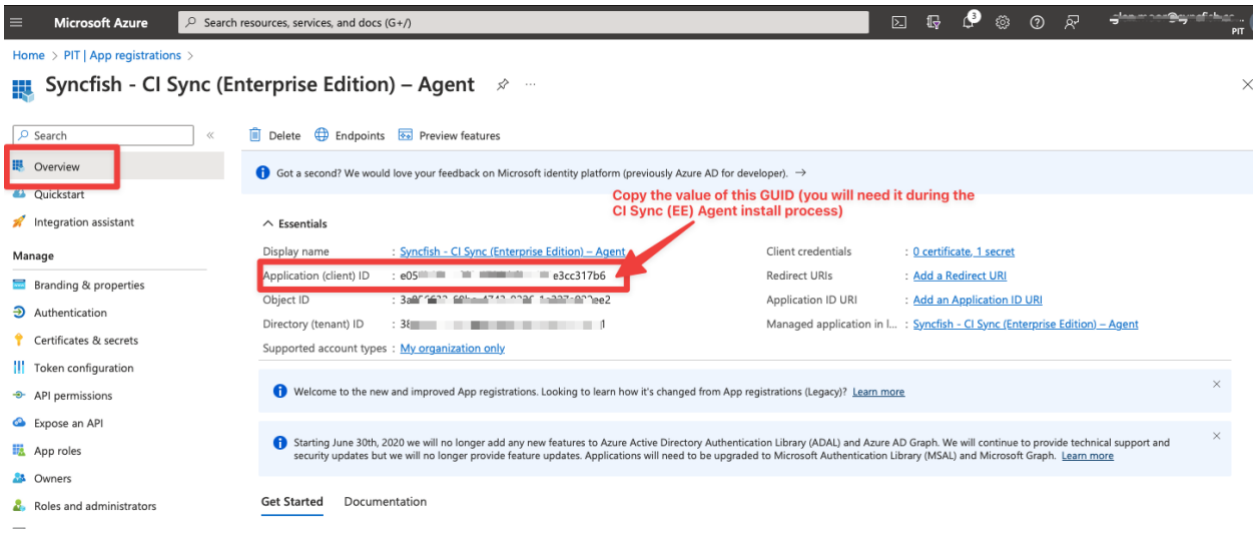


Bookmark: CISyncAgent_AppReg_ClientScrtSecretScrn

> **Data Capture Note:** The secret "**Value**" will used later by the person performing the tasks in Step 4 – Install the Multi-Source CI Sync (EE) Agent. It is recommended you securely store a copy of the secret so it can be shared internally with the SME performing the CI Sync (EE) Agent installation.

Click here to return to steps index

14. Return to the **Overview** page for the App Registration.
    - Use the copy option to make a copy of the "**Application (client) ID**" value.



Bookmark: CISyncAgent_AppReg_ClientScrtAppIDScrn

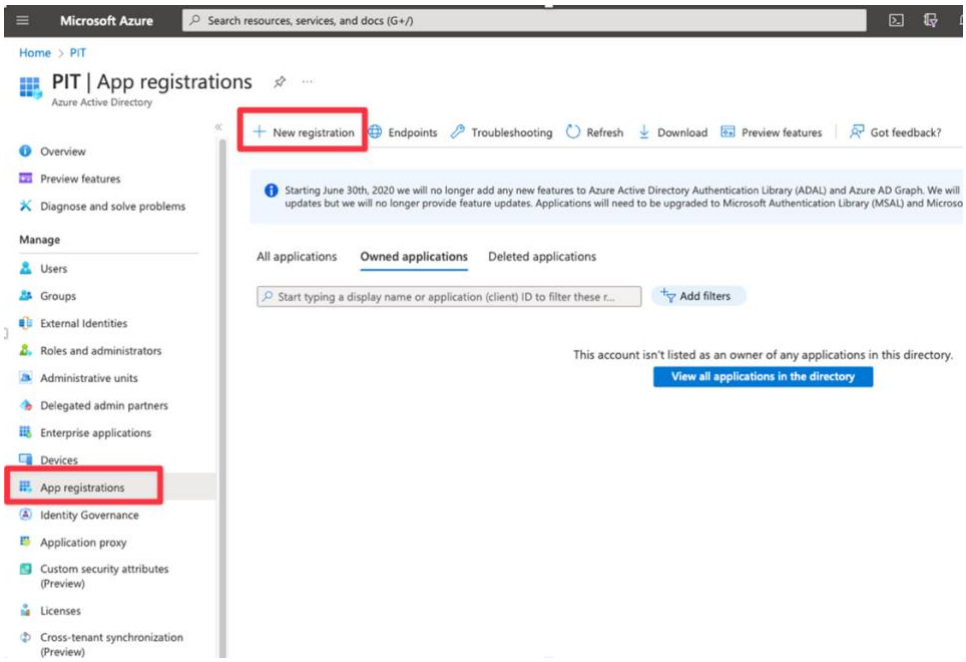
> **Data Capture Note:** The "**Application (client) ID**" will used later by the person performing the tasks in Step 4 – Install the Multi-Source CI Sync (EE) Agent.

15. **For clarity purposes only:** Upon completion of the above tasks, you should have recorded a copy of the two values shown below. These values will be needed by the SME performing the tasks in Step 4 – Install the Multi-Source CI Sync (EE) Agent.
    1. The **Secret Value**
    2. The **Application (client) ID**

At this point Task 3b (Option 1) is now complete. **You can therefore skip** Task 3b (Option 2) steps on the subsequent page and move now to Step 4 – Install the Multi-Source CI Sync (EE) Agent which is typically performed by your infrastructure SME.

Task 3b (Option 2): Create the App Registration object in your AAD using a Certificate credential.

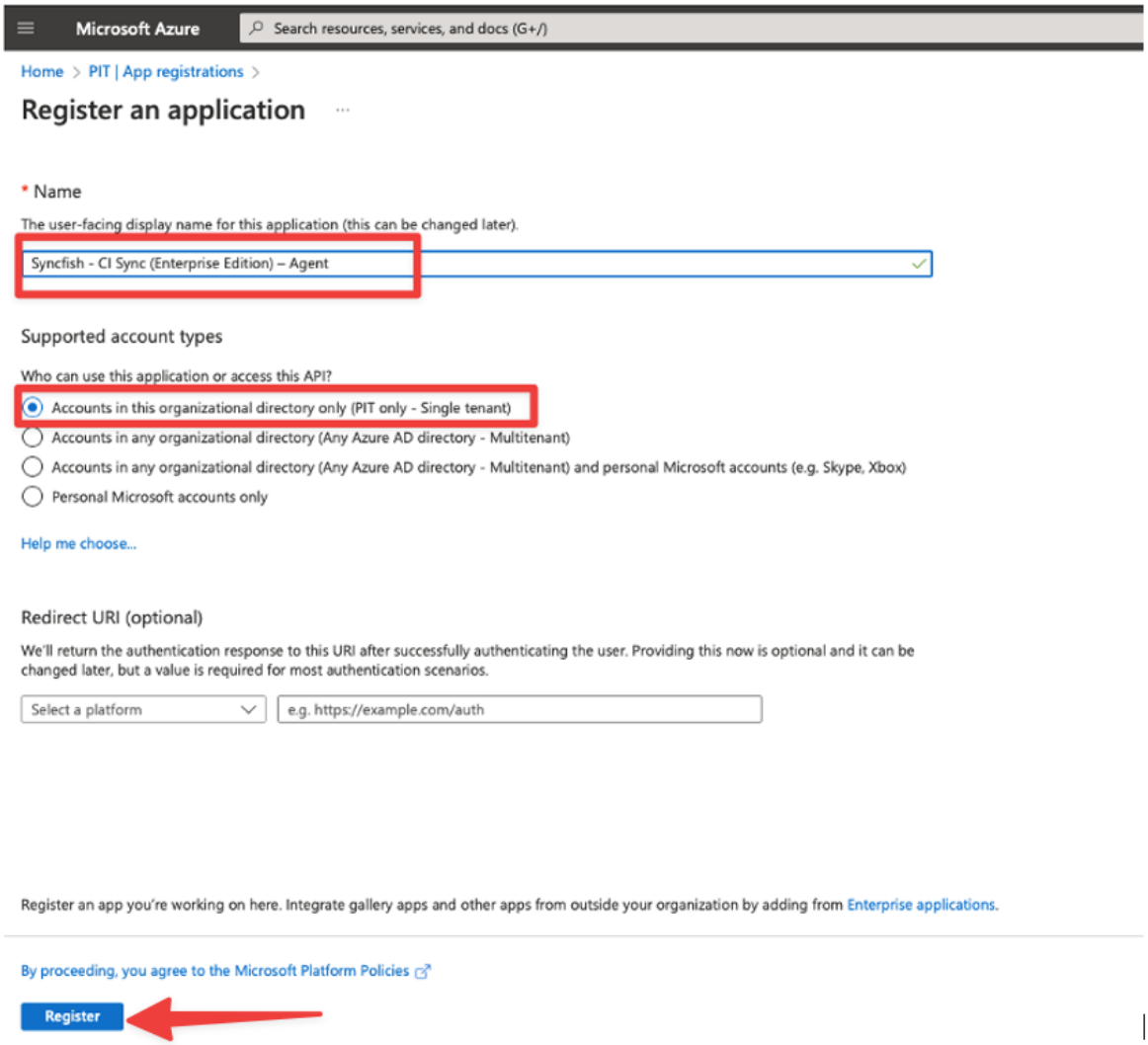
1. In the **Azure Portal**, navigate to **Azure Active Directory** -> **App Registrations** and click **New Registration**



2. On the **Register an application** form complete as follows:
   - Enter the **Name**

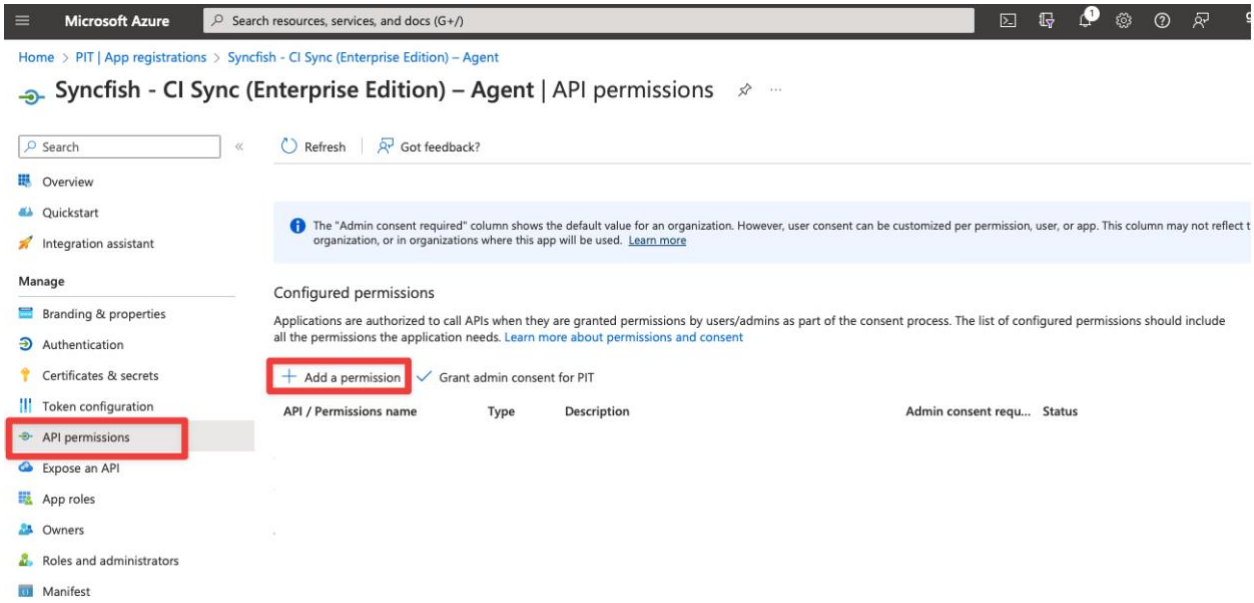
> **Guidance Note:** Syncfish recommend using "Syncfish - CI Sync (Enterprise Edition) – Agent".
>
> If you use another name, then **it is highly recommended to include "Agent" in the name** to help distinguish this registration from the CI Sync (EE) SaaS application Enterprise Application registration as AAD will show both in the same list on some forms.

   - Under **Supported account types** select "**Accounts in this organizational directory only (*{Your Domain/Tenant Name}* only - Single tenant)"**
   - Click **Register**



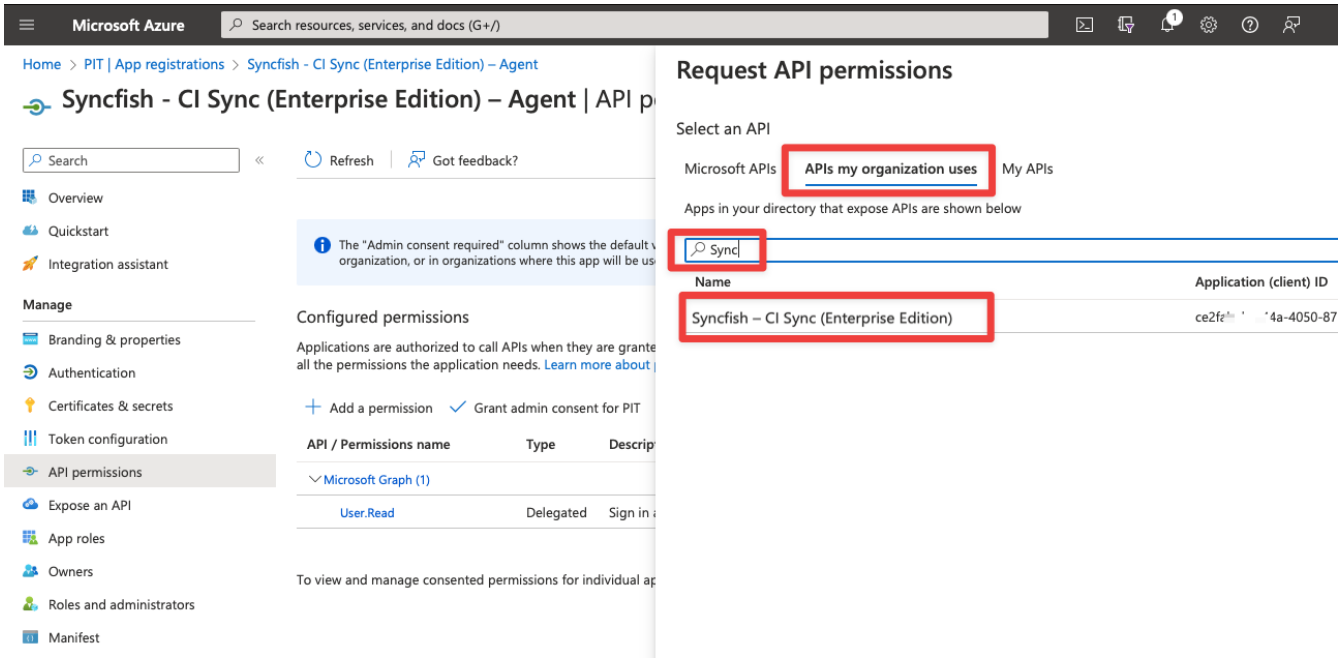
Click here to return to steps index

3. Navigate to **API permissions** and click **Add Permission**

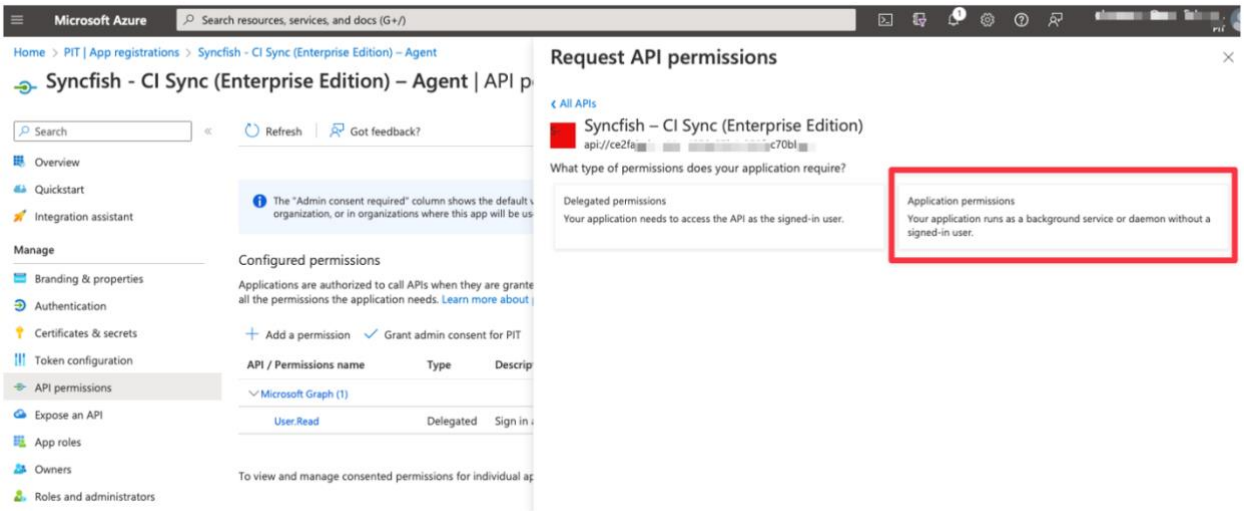


4. Navigate to **APIs my organization uses** and search for Syncfish, and then select the **Syncfish - CI Sync (Enterprise Edition)** entry.

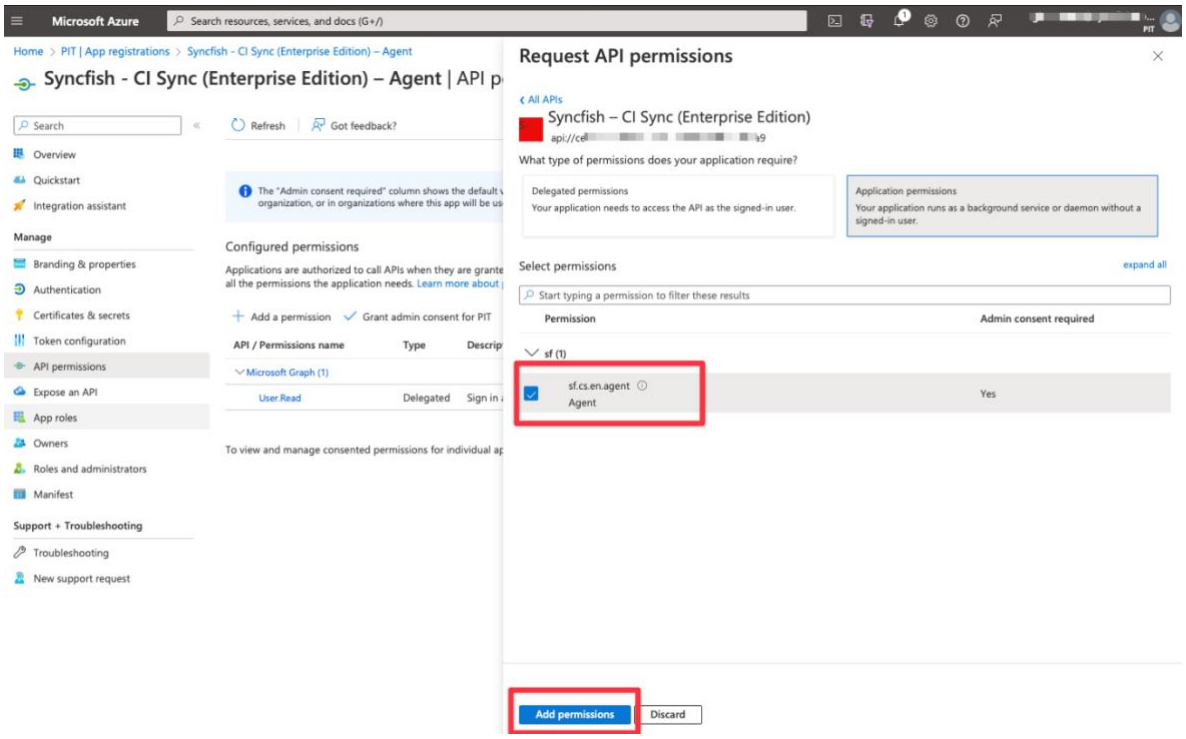
> **Guidance Note:** In the list **you are selecting the CI Sync (EE) Enterprise Application** that was enrolled during Step 2 – Enrol the CI Sync (EE) SaaS application into your AAD earlier in this guide.



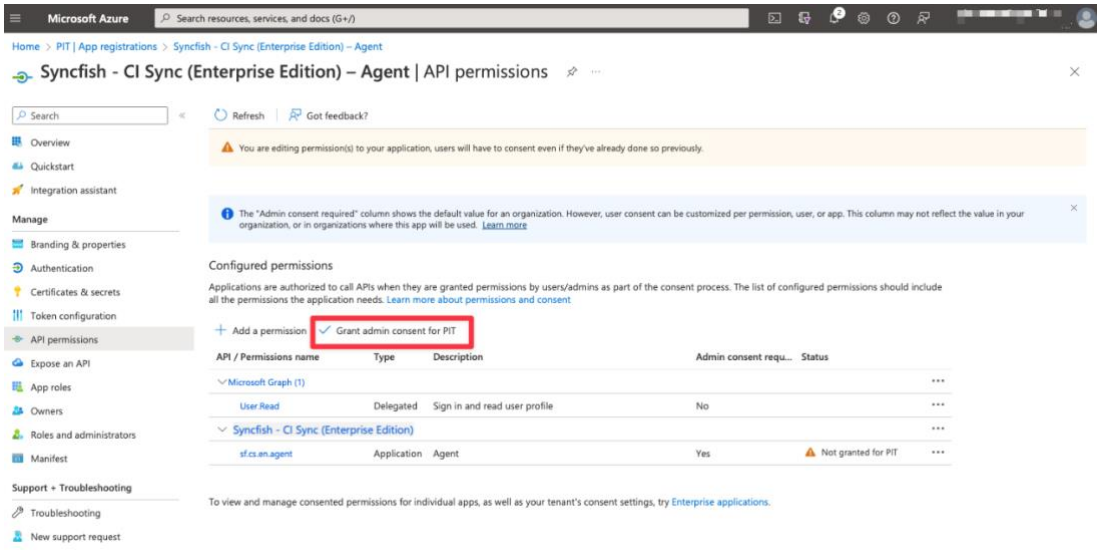
5. Select the "**Application permissions**" button



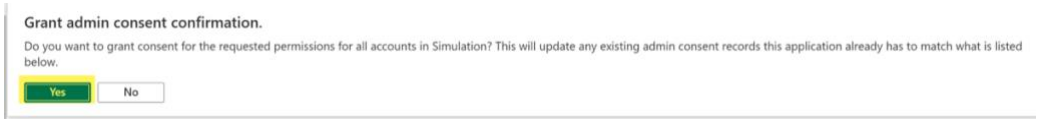
6. Tick the checkbox for **sf.cs.en.agent Agent** and click the **Add permissions** button



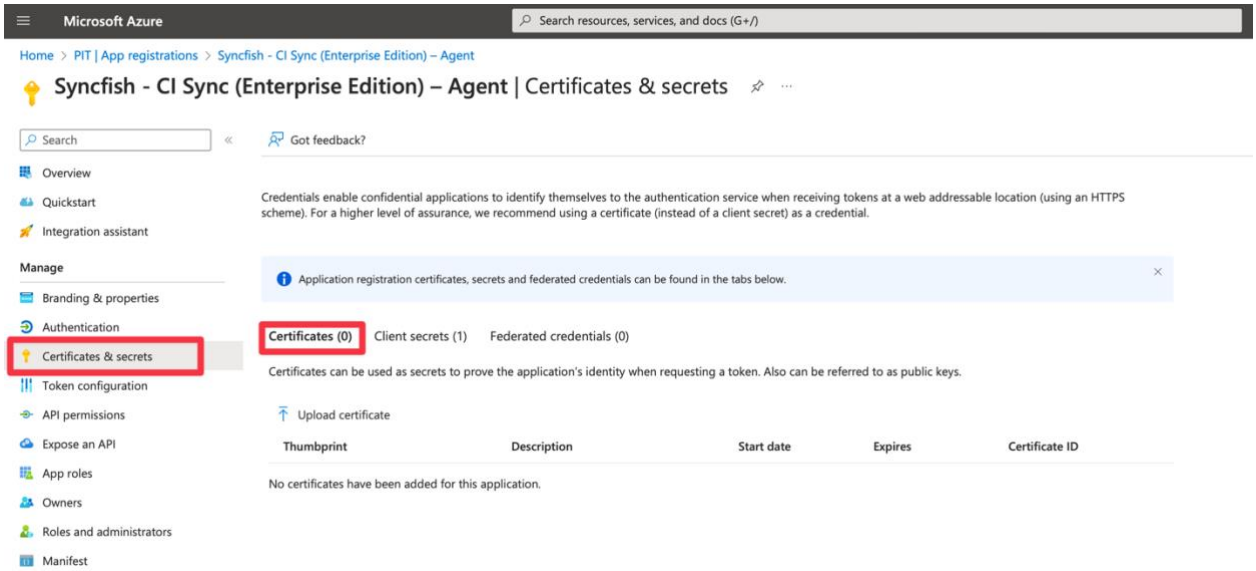
7. Back on the main screen for "API permissions", click the button to "**Grant admin consent for *{Your Domain/Tenant Name}***"



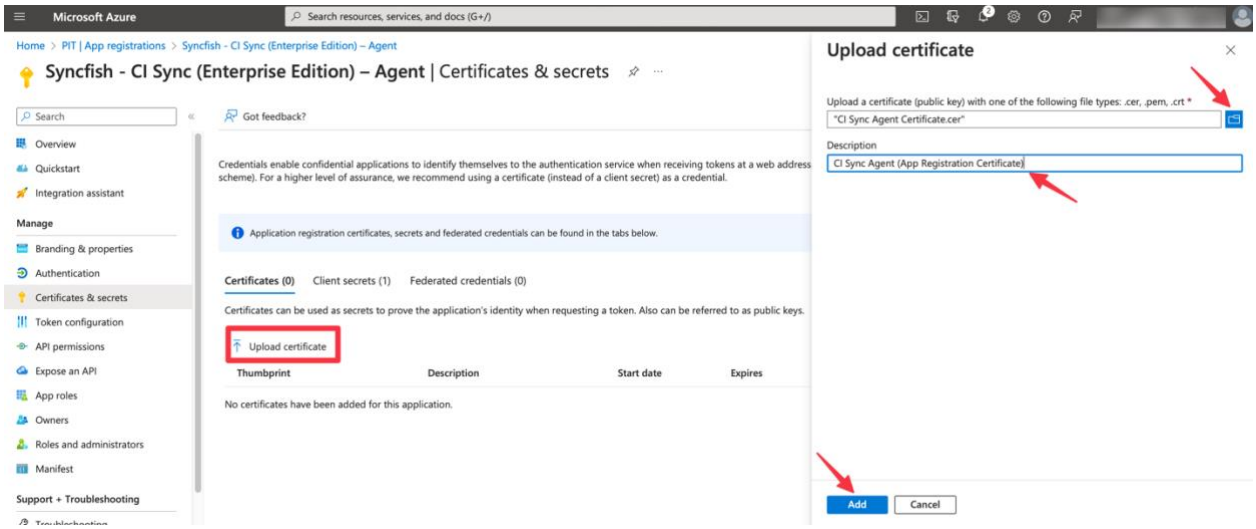
8. Click **Yes** to confirm



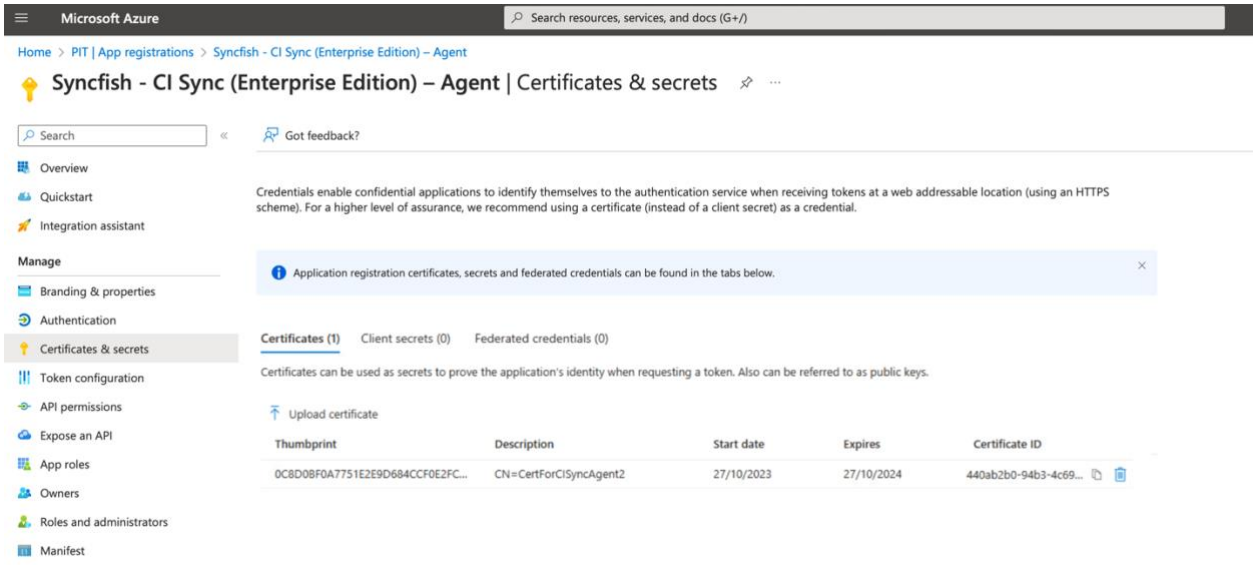
9. Using the left-hand menu, navigate and select **Certificates & secrets**. Select client "**Certificates (0)**" in the middle of the form.



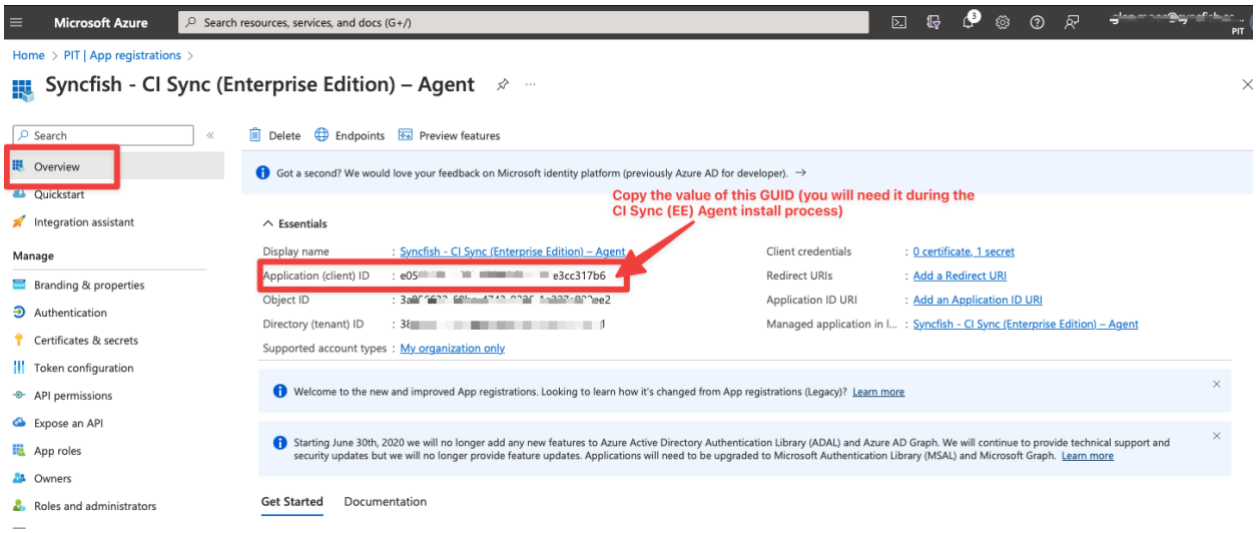
10. Click the "**Upload certificate**" button. Then use the file uploader to import the certificate (public key) file in one of the supported formats. Enter a description and then select the Add **button**.



11. The certificate should now be displayed in the Certificates list for this App Registration.



12. Return to the **Overview** page for the App Registration.
    - Use the copy option to make a copy of the "**Application (client) ID**" value.



Bookmark: CISyncAgent_AppReg_CertAppIDScrn

> **Data Capture Note:** The "**Application (client) ID**" will used later by the person performing the tasks in **Step 4 – Install the Multi-Source CI Sync (EE) Agent**.

Step 4 – Install the Multi-Source CI Sync (EE) Agent

In this step your infrastructure SME will perform the following tasks:

- Task 4a: Create a Windows Service Account for the CI Sync (EE) Agent to use.
- Task 4b: Configure the Windows Service Account.
- Task 4c: Download the CI Sync (EE) Multi-Source Agent from the CI Sync User Interface.
- Task 4d: Run the CI Sync (EE) Agent installer wizard.

Guidance Notes (to prepare for Step 4 and Step 5 tasks)

Syncfish recommends you decide at this point whether to use a Local Windows Account or AD Domain User Account as the Service Account for the CI Sync (EE) Agent Windows Service.

We also recommend you decide which authentication type (either Integrated Security vs SQL Native Login) will be used to authenticate between the CI Sync (EE) Agent and your SQL Server. The CI Sync (EE) Agent uses a SQL Server for two purposes:

1. For read/write access to the RecVer database. The RecVer database is the CI Sync (EE) Agent's delta record tracking SQL DB.
2. For read access to synchronization data sources for products such as Lansweeper and SCCM.

Pros and cons about using a Local Windows User Account as the Service Account

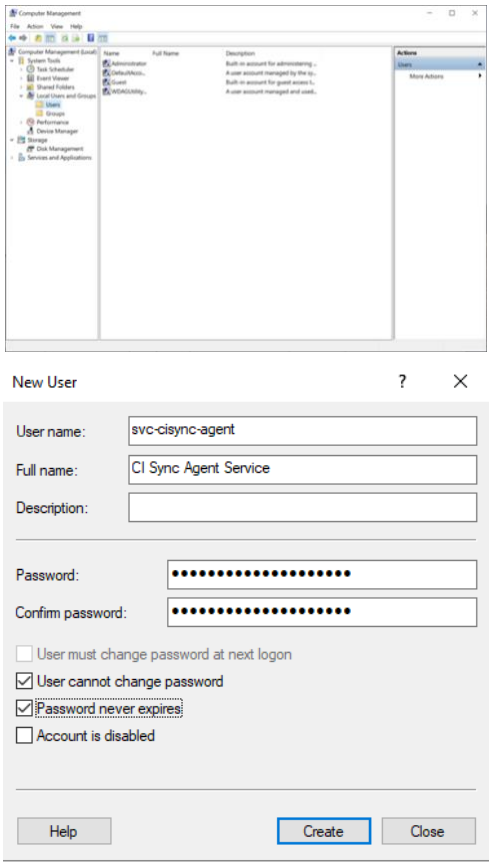
- A Local Account can be quickly created by the person performing the CI Sync (EE) Agent installation.
- A Local Account cannot be used as the SQL authentication account if SQL Server is running on a separate server to the CI Sync (EE) Agent. That is, a Local Account cannot be used for "Integrated Security" login to a remote SQL server. If your SQL Server is remote to the CI Sync (Agent) server then you need to do one of the following:
  - Either run the CI Sync (EE) Agent (i.e. the Windows Service) with an AD Domain User Account
  - Or, run the CI Sync (EE) Agent (i.e. the Windows Service) using a Local Windows User Account and **then use** a SQL Native Login credential for the remote SQL Server when you add your Source Connection(s) during Step 5 – Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection (using Lansweeper as the example) later in this setup guide.

Pros and cons about using an AD Domain User Account as the Service Account

- An AD Domain User account can be used as the authentication account to a remote SQL server.
- An AD Domain User account often needs to setup by a separate team.

Task 4a: Create a Windows Service Account for the CI Sync (EE) Agent to use.

1. **If you are using a Local Windows User Account**, then perform the following (on the target Server):
   - Login to the console of the Server with an account that has Local Administrator rights.
   - Navigate to **Computer Management** -> **Local Users and Groups** -> **Users**
   - From the **Action** menu select **New user…**
   - Enter the **User name** *(e.g. svc-cisync-agent)*
   - Enter the **Full name** *(e.g. see screen shot)*
   - Enter the **Password**
   - **Confirm the Password** when prompted
   - It is recommended to **set the checkbox values** shown below (or amend to your company policy)
   - Click the **Create** button
   - Add the user to the "Users" group.



> **Data Capture Note:** Make sure you record Username/Password as they will be required during Step 5 when you run the CI Sync (EE) Agent Config Utility.
>
> The user only needs to be in the Users group. It does not need to be in any elevated groups.

2. ***Alternatively*** *if you are not using a Local Windows User Account* and instead are using an AD Domain **User**, then perform the following in your Active Directory:
   - Login to your Active Director domain and use the Users and Computers MMC to create a domain user account.
   - Create the account according to your company standards for a Service Account (e.g. svc-cisync-agent or something similar).
   - The user only requires domain user privileges (i.e. be a member of the Domain Users Group). It **does not require** any elevated permissions.

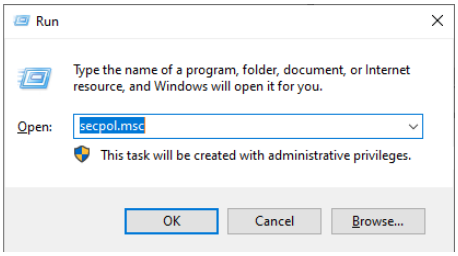
> **Data Capture Note:** Make sure you record Username/Password as they will be required during Step 5 – Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection (using Lansweeper as the example) when you run the CI Sync (EE) Agent Config Utility.

Click here to return to steps index

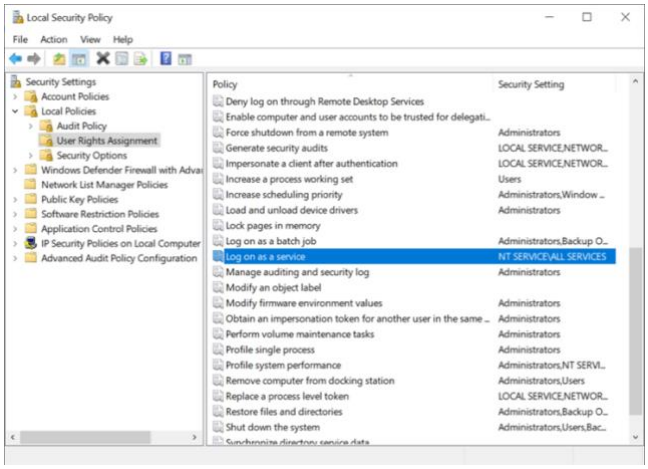
Task 4b: Configure the Windows Service Account.

Perform the following steps on the server that will be used to run the CI Sync (EE) Agent.

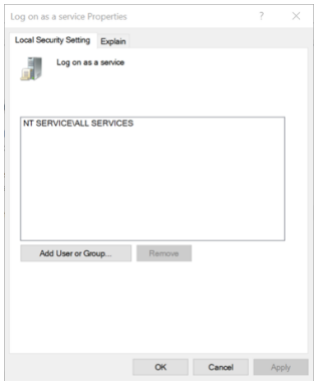
1. Run secpol.msc



2. Navigate to **Local Security Policy**



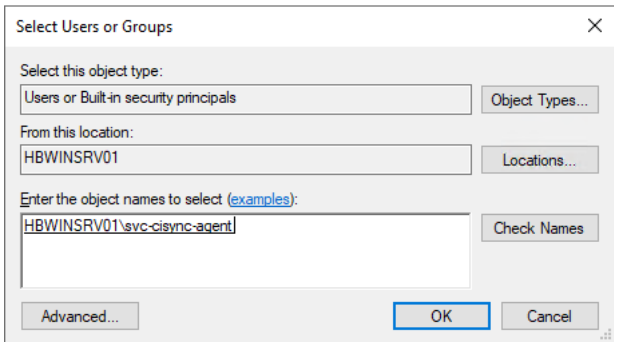
3. Navigate to **Local Policies -> User Rights Assignment** and double-click the **Log on as a service** list entry



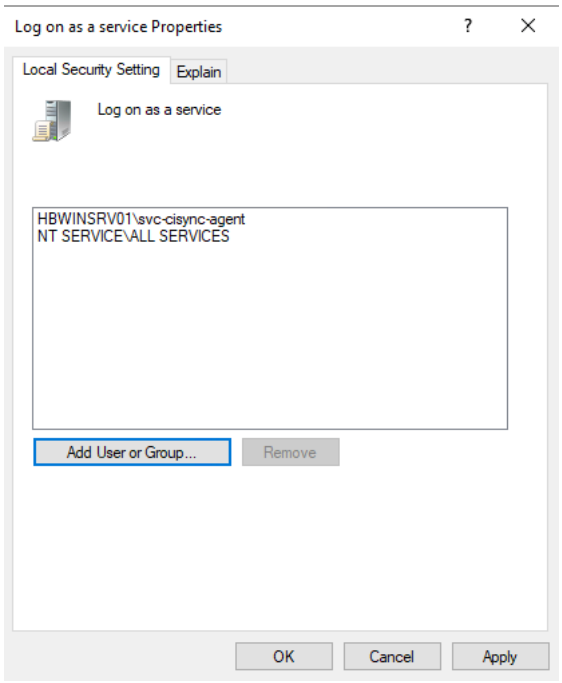
4. Click **Add User or Group** and enter the name of the user created above during (either the **AD Domain User Account** or a **Local Windows User Account**)

Click here to return to steps index

5. Click **Check Names**
6. Click **OK**

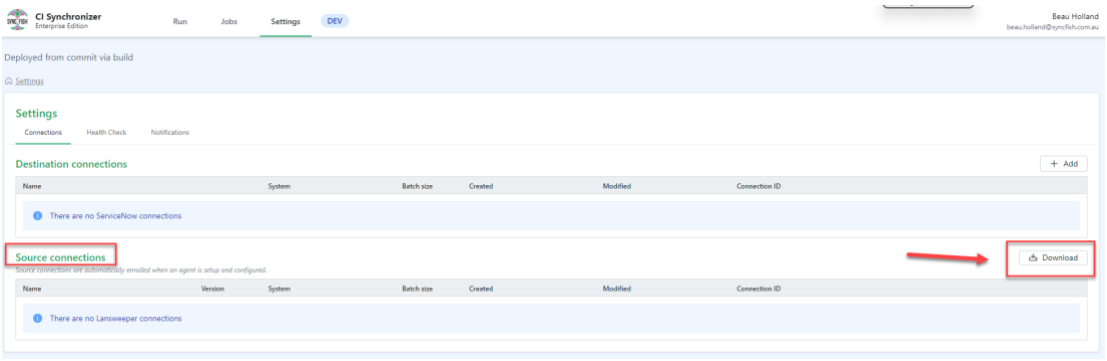


7. Click **Apply and OK**



Task 4c: Download the CI Sync (EE) Multi-Source Agent from the CI Sync User Interface.

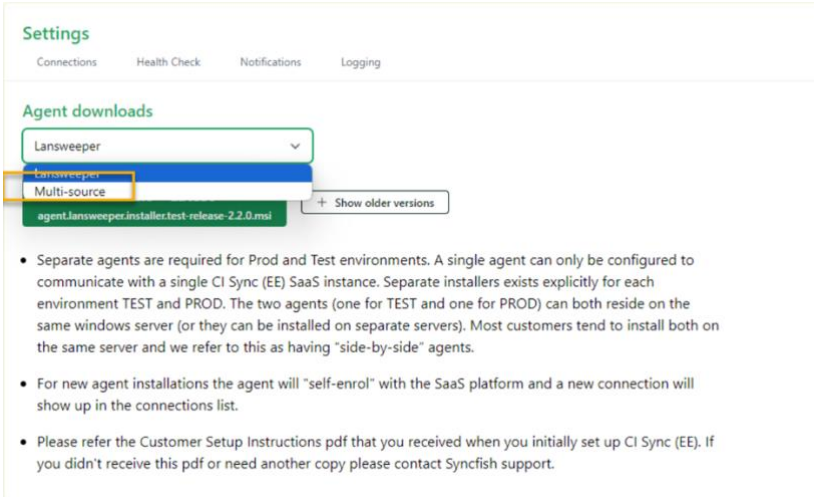
1. Login to your CI Sync (EE) SaaS instance at https://***YourCo***.syncfish.app
2. In the CI Sync UI, navigate to Settings > Connections.
3. Click the "Download" next to the Source Connections list.



4. Use the drop down list and select the "Multi-Source" option to select the CI Sync (EE) Multi-Source Agent described in these instructions, and then select the Download button.

Guidance Notes:

1. The options presented in the drop down list are based on the CI Sync (EE) licence allocated to your organization. Please contact Syncfish if you do not see the "Multi-Source" option.
2. If you see "Lansweeper" in the dropdown list this refers to the original "single-source" CI Sync (EE) Agent. Do not select the "Lansweeper" option if wanting to download/use the new "Multi-Source" agent described in these instructions.

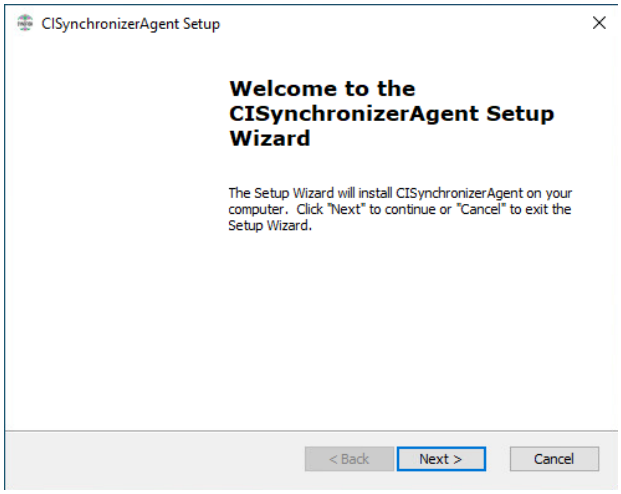


Click here to return to steps index

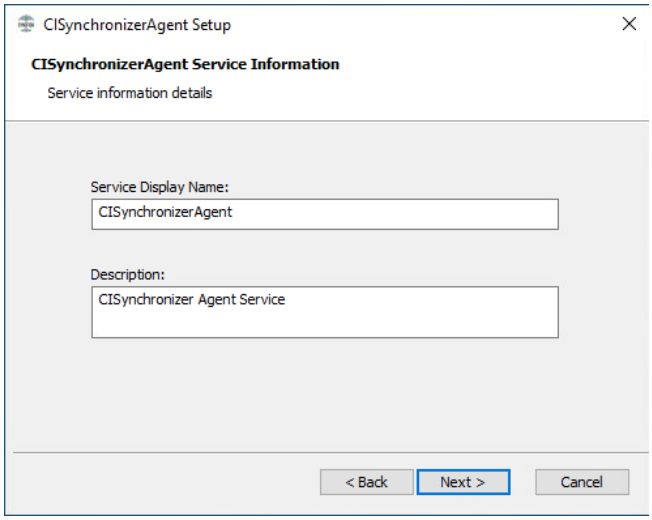
Task 4d: Run the CI Sync (EE) Agent installer wizard.

Perform the following steps on the server that will be used to run the CI Sync (EE) Agent.

1. Run the CI Sync (EE) Agent installer to commence the installation.

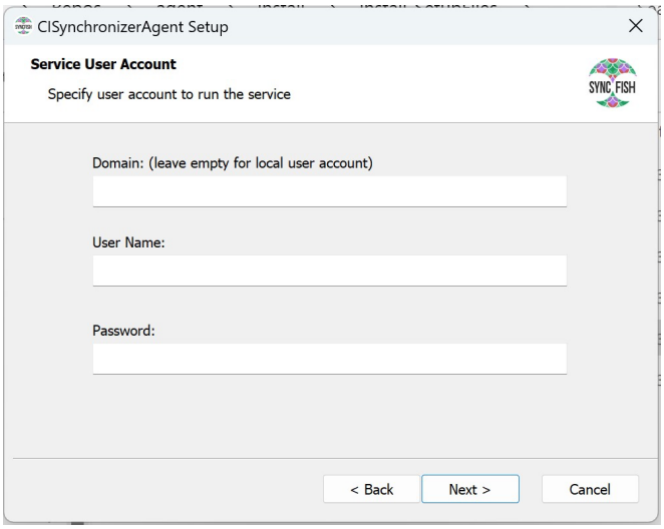


2. Click **Next** and change (if desired) the Service Display Name and Description



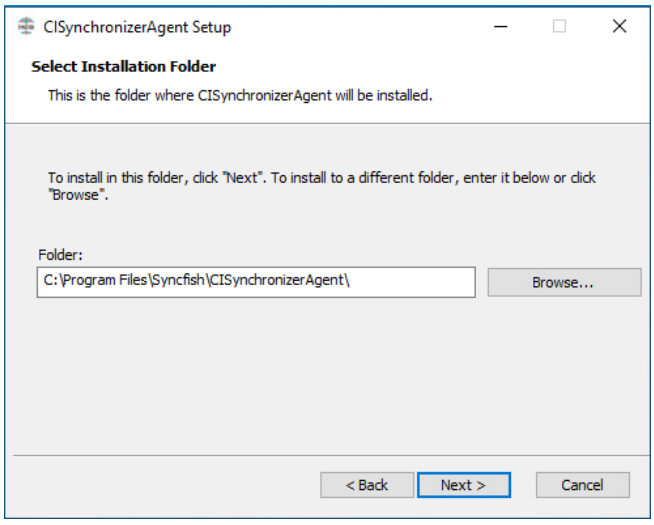
3. Click **Next**

4. Enter the **name for the Service Account** created during Task 4a: Create a Windows Service Account for the CI Sync (EE) Agent to use. (either the Local Windows User Account or the AD Domain User Account)
   - If using a **domain account**, enter the AD Domain in the first field and the username in the next field.
   - If using a **local account**, leave the first field (Domain) blank and just enter the local user account the second field (e.g. **svc-cisync-agent**).
   - Enter the **password** for the account.

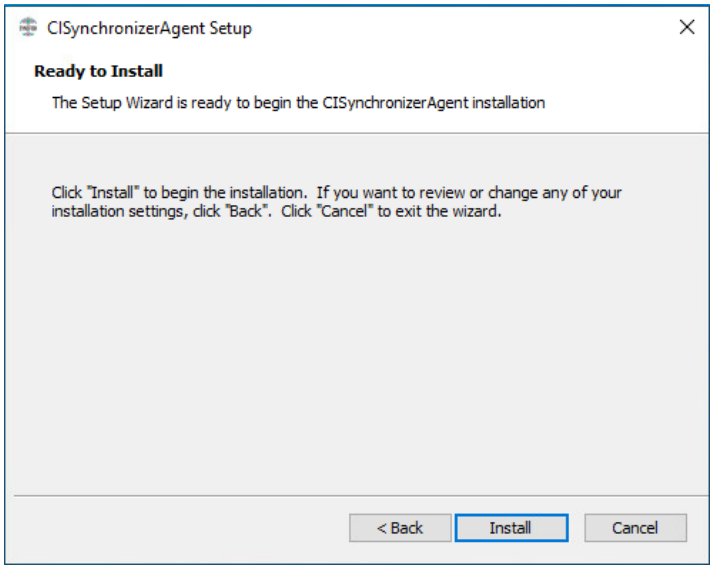


   - Click the **Next** button

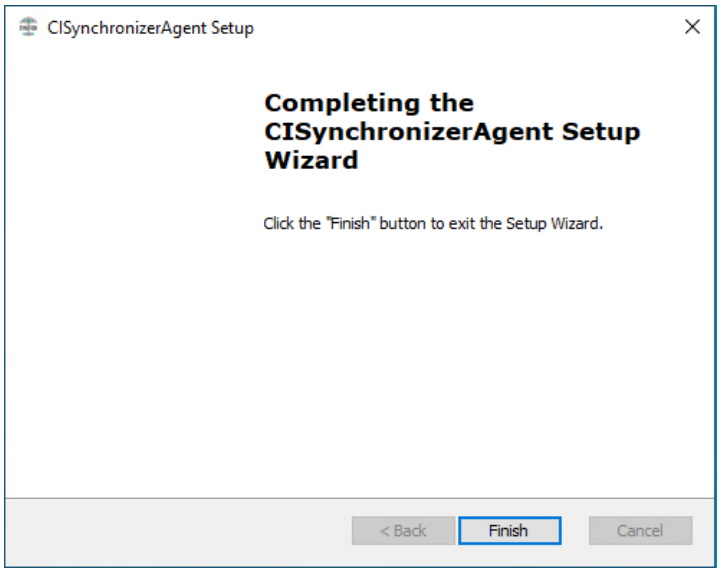
5. At Select an Installation **Folder**, change if required, and click **Next**



6. Click **Install** on the **Ready to Install** prompt



7. Wait for the installation to complete (several minutes).



- Click **Finish**

Guidance Note: Upon completion the **CI Sync Agent Config Utility** will be automatically launched. Instructions for using the CI Sync Agent Config Utility are in Step 5 – Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection (using Lansweeper as the example) (the next section).

Step 5 – Use the CI Sync (EE) Agent Config Utility to Setup one or more Source System Connections

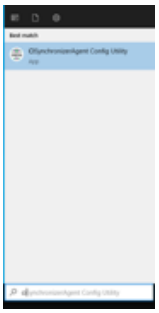
In this step your Source System (or infrastructure) SME will perform the following tasks. Depending on your specific circumstances you may also require the assistance of an AAD SME and a SQL DBA.

- Task 5a: Use the **CI Sync (EE) Agent Config Utility** to enter connection values.
- Task 5b: Use the **CI Sync (EE) Agent Config Utility** to test the connection values and register the CI Sync (EE) Agent with your CI Sync (EE) SaaS Instance.
- Task 5c: Use the **CI Sync (EE) Agent Config Utility** to setup Source System Connections.

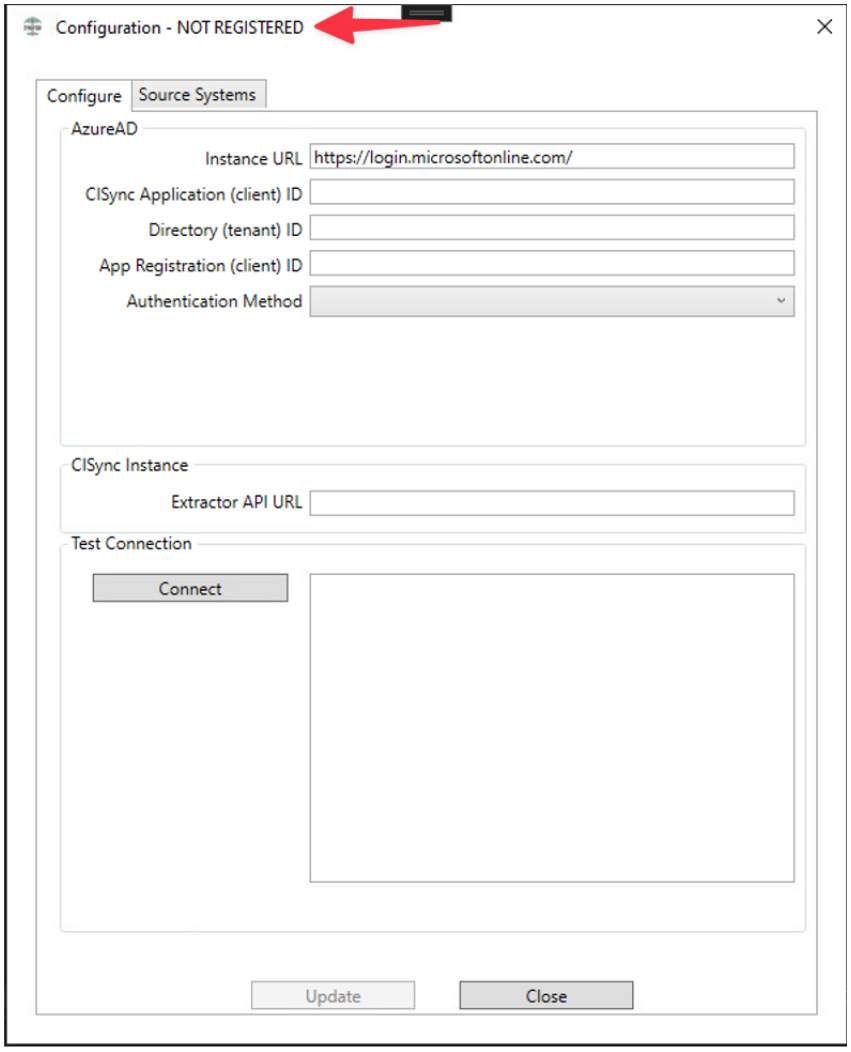
Task 5a: Use the CI Sync (EE) Agent Config Utility to register the CI Sync (EE) Agent with your CI Sync (EE) SaaS Instance.

Perform the following steps on the server where the CI Sync (EE) Agent was installed (referred to as the "CI Sync (EE) Agent Server").

1. If you have just completed Step 4 – Install the Multi-Source CI Sync (EE) Agent (to install the Multi-Source CI Sync (EE) Agent) the **CI Sync (EE) Agent Config Utility** will have automatically loaded. If the Config Utility is not running the locate and run from the Start Menu (i.e. run the "**CISynchronizerAgent Config Utility**" program).



2. If you are running the Config Utility for the first time the CI Sync (EE) Agent (i.e. the Windows Service) needs to be registered with your customer specific CI Sync (EE) SaaS instance. You can see this from the "NOT REGISTERED" text at the top of the form.



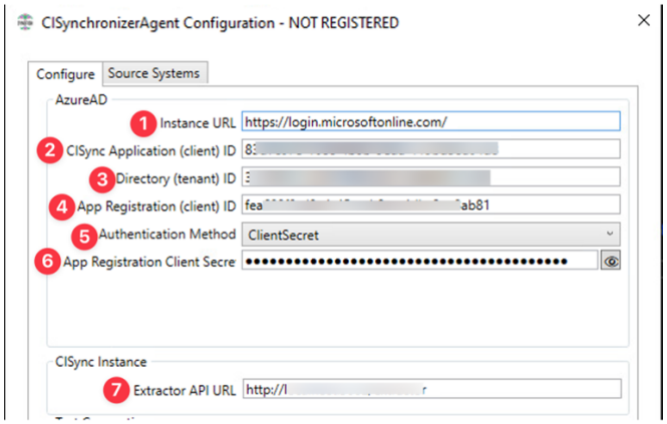
Follow the guidance on the subsequent pages to enter the required values. Two sets of instructions are provided depending on which authentication method you are using.

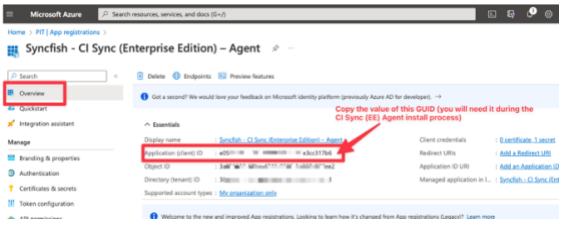
- Follow Task 5a (Option 1): Client Secret Authentication Steps if your AAD App Registration object was configured by your AAD SME to use a Client Secret.
- Follow Task 5a (Option 2): Certificate Authentication Steps if your AAD App Registration object was configured by your AAD SME to use a Certificate.

Click here to return to steps index

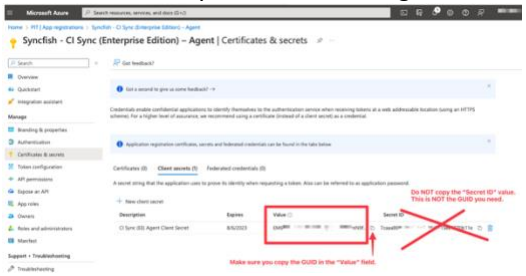
Task 5a (Option 1): Client Secret Authentication Steps

1. Use the CI Sync Configuration Utility, navigate to the **Configure Tab** (as shown below) to enter the relevant values. The table below explains each field and the required value.



#	Field Name on the Configure Tab	Value you need to enter
1	**Instance URL** *Note: This is the URL to the directory provider. In all cases use the value shown in the next column.*	https://login.microsoftonline.com/
2	**CISync Application (client) ID** *Note: This is a static value provided by Syncfish. In all cases use the value shown in the next column.*	fa15c723-addb-4922-8463-8d0fbce47cb1
3	**Directory (tenant) ID** *Note: This is your Azure Active Directory Tenancy ID (the same one you provided to Syncfish when your CI Sync (EE) SaaS instance was being provisioned.*	This value is obtained via your **Azure Portal** under **Active Directory -> Overview**. 
4	**App Registration (client) ID** *Note: Enter the client ID of the App Registration in your Azure AD you created during Step 3 – Create an AAD App Registration for CI Sync (EE) Agent Authentication.*	See the screen shot below for reference or you can click here to skip back to the task in Step 3 where you recorded this value. 

Click here to return to steps index

#	Field Name on the Configure Tab	Value you need to enter
5	**Authentication Method**	Select **ClientSecret**
6	**App Registration Client Secret** *Note: Enter the **Value** field (NOT the Secret ID GUID) for the CI Sync (EE) Agent you recorded when creating the Azure AD App Registration for the Agent.*	See the screen shot below for reference or click here to skip back to the tasks where you first created and recorded the secret **Value**. If you didn't record the secret **Value** at the time you will need to regenerate it. 
7	**Extractor API URL** *Note: This is your customer specific CI Sync API URL (i.e. the "Extractor API" URL) sent to you by Syncfish.*	This is your company specific CI Sync (EE) URL with an **/extractor** suffix. That is: https://*YourCo*.syncfish.app**/extractor**

Informational Note: Secrets entered into the **CI Sync (EE) Agent Configuration Utility** are stored in the Windows Credential Store.

2. Finally, skip past Task 5a Option 2 (on the next page) and instead proceed straight to Task 5b: Use the CI Sync (EE) Agent Config Utility to test the connection values and register the CI Sync (EE) Agent with your CI Sync (EE) SaaS Instance.

Task 5a (Option 2): Certificate Authentication Steps

Informational Note: This section assumes the organization has a certificate management solution in place and the certificate related to this activity (which was also used for the AAD App Registration object creation) is available to the Windows Server being used to install/run the CI Sync (EE) Agent (i.e. the Windows Service).

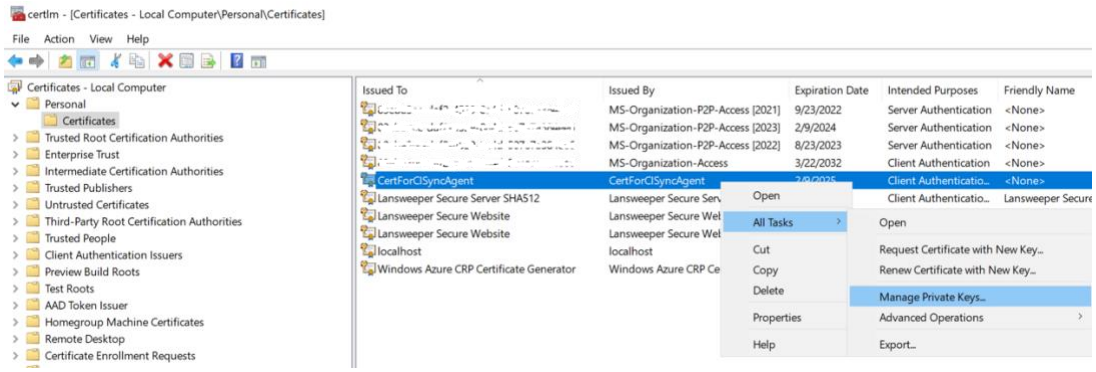
1. Import the Digital Certificate into the Windows Local Machine Certificate Store. It is recommended you do **not** import into the Current User Certificate Store.

   Informational Notes:

   - The Digital Certificate referred to above is the provided to you by your AAD SME (i.e. the SME who performed Task 3b (Option 2): Create the App Registration object in your AAD using a Certificate credential. in your AAD using a Certificate Credential).
   - When importing the certificate, it is recommended you import to the Local Machine Certificate Store. If you import it to the Current User you will almost certainly strike errors when the CI Sync (EE) Agent user account (i.e. the user account used by the Windows Service) tries to access the certificate.
   - Only consider importing to the Current User Certificate if you are running the CI Sync (EE) Config Utility with "Run As" in the context of the CI Sync (EE) Agent user account (i.e. the user account used by the Windows Service).

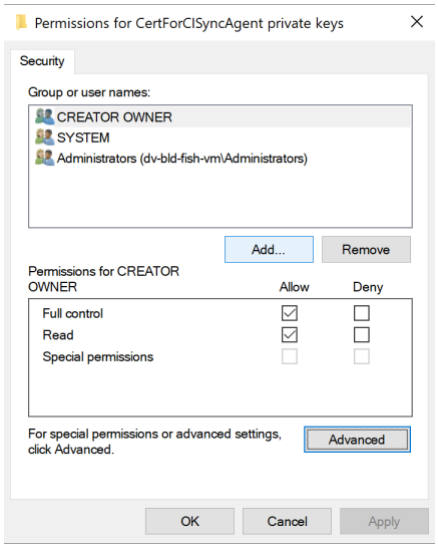
2. Ensure the CI Sync (EE) Agent user account (i.e. the user account used by the Windows Service) has sufficient permissions to read the Digital Certificate you imported. The following steps explain how to check/set the right permissions.

   a. Open Microsoft Management Console
   b. Navigate to *Certificates - Local Computer* → *Personal* → *Certificates*
   c. Right-click on the relevant certificate (the one you imported) and select *All Tasks* → *Manage Private Keys*

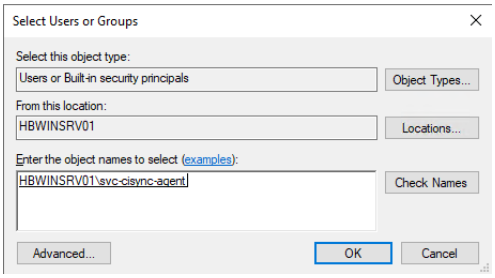


Click here to return to steps index

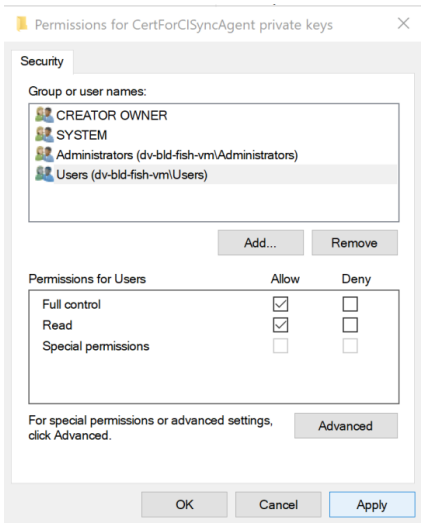
d. On the permissions window, click the Add button.



e. Search for Users, locate the CI Sync (EE) Agent User *(e.g. svc-cisync-agent)*, press Check Names, then press the OK button.

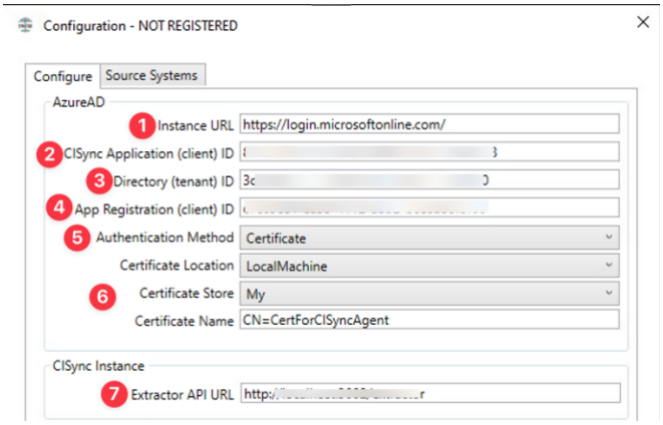


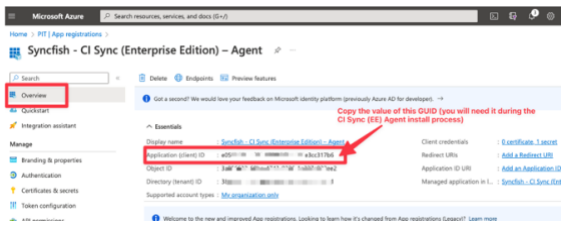
f. Back on the permissions window, make sure Full control and Read are ticked. Then click Apply and then Ok.



Click here to return to steps index

3. Use the CI Sync Configuration Utility, navigate to the **Configure Tab** (as shown below) to enter the relevant values. The table below explains each field and the required value.



#	Field Name on the Configure Tab	Value you need to enter
1	**Instance URL** *Note: This is the URL to the directory provider. In all cases use the value shown in the next column.*	https://login.microsoftonline.com/
2	**CISync Application (client) ID** *Note: This is a static value provided by Syncfish. In all cases use the value shown in the next column.*	fa15c723-addb-4922-8463-8d0fbce47cb1
3	**Directory (tenant) ID** *Note: This is your Azure Active Directory Tenancy ID (the same one you provided to Syncfish when your CI Sync (EE) SaaS instance was being provisioned.*	This value is obtained via your **Azure Portal** under **Active Directory -> Overview**. 
4	**App Registration (client) ID** *Note: Enter the client ID of the App Registration in your Azure AD you created during Step 3 – Create an AAD App Registration for CI Sync (EE) Agent Authentication.*	See the screen shot below for reference or you can click here to skip back to the task in Step 3 where you recorded this value. 
5	**Authentication Method**	Select **Certificate**

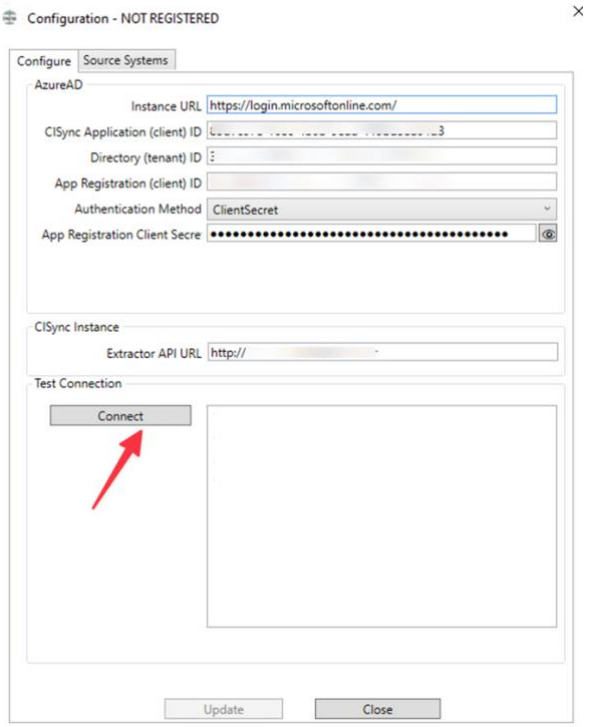
#	Field Name on the Configure Tab	Value you need to enter
6	**Certificate Location**	**Options:** LocalMachine, CurrentUser **Recommendation:** It is strongly recommended you use LocalMachine, as the current user account (the person installing/running the CI Sync (EE) Agent Config Utility) is not the same as the CI Sync (EE) Agent user account (i.e. the user account used by the Windows Service). If you have saved the certificate in the CurrentUser context, rather than the LocalMachine context (of the Windows Certificate Store) it will cause an error when the CI Sync (EE) Agent user account (i.e. the user account used by the Windows Service) tries (and fails) to read the certificate from the Windows Certificate Store.
	Certificate Store	**Options:** AddressBook, AuthRoot, CertificateAuthority, My, Root, TrustedPeople, TrustedPublisher The Certificate Store selections are a standard set of values provided by the Microsoft and correspond to the folders in the certificate store. **Recommendation:** It is recommended this value be set to the "My" store (this translates to the "Personal/Certificates" path in the Windows Certificate Store).
	Certificate Name	Enter the subject of the certificate related to the CI Sync (EE) Agent authentication to the App Registration object.
7	**Extractor API URL** *Note: This is your customer specific CI Sync API URL (i.e. the "Extractor API" URL) sent to you by Syncfish.*	This is your company specific CI Sync (EE) URL with an **/extractor** suffix. That is: https://*YourCo*.syncfish.app**/extractor**

4. Finally, proceed to Task 5b: Use the CI Sync (EE) Agent Config Utility to test the connection values and register the CI Sync (EE) Agent with your CI Sync (EE) SaaS Instance.

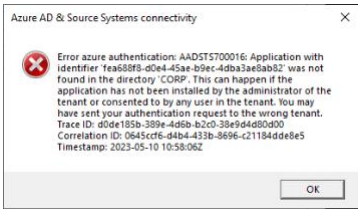
Click here to return to steps index

Task 5b: Use the CI Sync (EE) Agent Config Utility to test the connection values and register the CI Sync (EE) Agent with your CI Sync (EE) SaaS Instance.

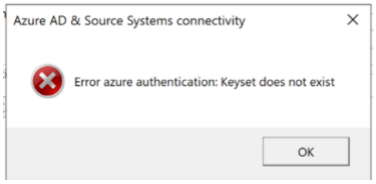
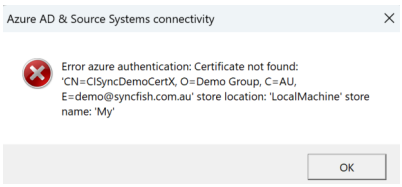
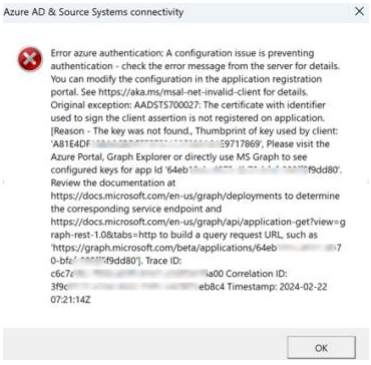
1. After entering the various values, click the **Connect** button under the "Test Connection" heading.



2. If the connection test fails an error message dialog will be displayed containing details of the error (see examples below).

Example #	Screen Shot	Applicable Authentication Method and Possible Cause
1		**Applicable Authentication Method** ☑ Client Secret Authentication ☑ Certificate Based Authentication **Possible Cause** The App Registration (client) ID value entered into the CI Sync (EE) Config UI does not match (cannot be located) in Azure AD. Recheck the App Registration in AAD and ensure you have copy/pasted the correct App Registration as per the steps and tasks from Section 4 of this document. ☒

Click here to return to steps index

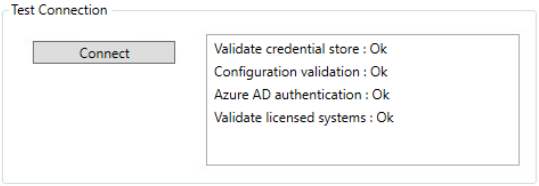
Example #	Screen Shot	Applicable Authentication Method and Possible Cause
2		**Applicable Authentication Method** ☒ Client Secret Authentication ☑ Certificate Based Authentication **Possible Cause** The CI Sync (EE) Agent cannot locate the Digital Certificate in the Windows Certificate Store due to a permissions error. There are several reasons this might occur. Refer to Appendix Z and specifically Trouble Shooting Topic Z1: Keyset does not exist error using CI Sync Agent "Test Connection" to validate Azure authentication with certificate based authentication.
3		**Applicable Authentication Method** ☒ Client Secret Authentication ☑ Certificate Based Authentication **Possible Cause** The CI Sync (EE) Agent cannot locate the Digital Certificate in the Windows Certificate Store due to a mismatch in the Certificate Name attribute. Refer to Appendix Z and specifically Trouble Shooting Topic Z2: Certificate not found error using CI Sync Agent "Test Connection" to validate Azure authentication with certificate based authentication.
4		**Applicable Authentication Method** ☒ Client Secret Authentication ☑ Certificate Based Authentication **Possible Cause** The CI Sync (EE) Agent cannot locate the Digital Certificate associated with the App Registration in the Azure AD (based on the certificat thumbprint value). Recheck the App Registration in AAD and ensure you have the corect certificate details stored against the App Registration and the correct Digital Certificate has been imported into the Windows Certificate Store.

Guidance Note: In most cases an error here will be one of the following:

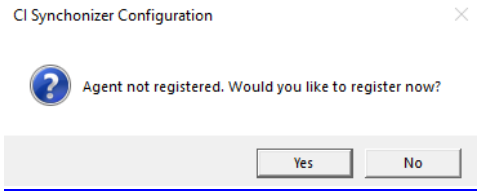
- The server running the CI Sync (EE) Agent (the Windows Service) does not have HTTPS outbound access to the Internet.
- One of the values entered on the Configure Tab is incorrect (e.g. the details for the App Registration object in AAD are not correct).
- There is a mismatch in the format of the Certificate Name entered into the CI Sync (EE) Config Utility vs the actual name of the Certificate in the Windows Certificate Store.
- Appendix Z of this document contains trouble shooting notes on certain error messages.
- If the error cannot be resolved, please reach out to your Syncfish contact or email support@syncfish.com.au.

Click here to return to steps index

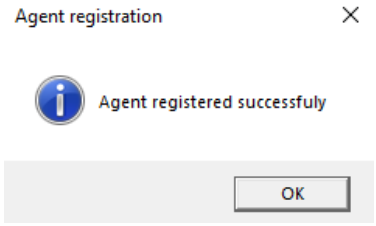
3. If the Agent successfully authenticates to your customer specific CI Sync (EE) SaaS instance the Test Connection result will look as shown below.



4. Click **Yes** to register the CI Sync (EE) Agent with your customer specific CI Sync (EE) SaaS instance.

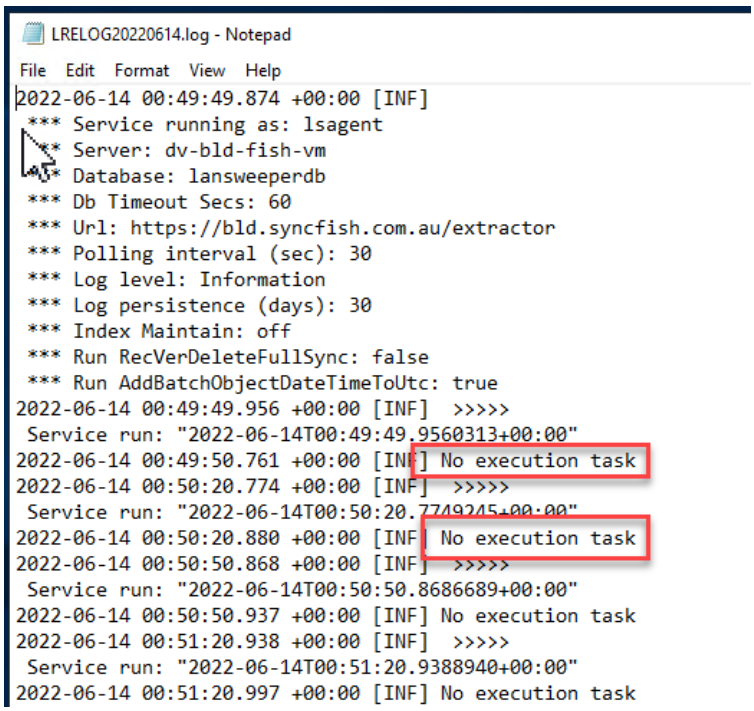


5. Click **OK** to agent being registered successfully.



6. Optionally, view the View the **CI Sync (EE) Agent log file** content as an initial check there are no errors:
   - Open the latest **CI Sync (EE) Agent log file** in the folder:
     C:\Program Files\Syncfish\CISynchronizerAgent\service\logs

   - If you see “No execution task” in the logs and no related ERROR logs the Agent is successfully communicating with your CI Sync (EE) SaaS instance



Informational Note: Log files are organised by day eg: LRELOG20220615.log = logs for the 15th June 2022.

Click here to return to steps index

Task 5c: Use the CI Sync (EE) Agent Config Utility to setup Source System Connections.

The CI Sync (EE) Agent technology allows you to sync from multiple different source systems in a single agent. Each Source System is represented in the CI Sync (EE) Agent as a Source System Connection. Source Systems include products such as Lansweeper, SCCM, Intune, Azure and so on.

Each source system (i.e. source system product) implements its own authentication technology and requirements. Therefore, the steps to setup a Source System Connection using the CI Sync (EE) Agent Config Utility are different for each system source.

Please now skip to the relevant Appendix listed below for instructions on using the CI Sync (EE) Agent Config Utility to create one or more Source System Connections within the CI Sync (EE) Agent.

Skip to Appendix D for How to add **Lansweeper** as a Source Connection to the CI Sync (EE) Agent.
Skip to Appendix E for How to add **Intune** as a Source Connection to the CI Sync (EE) Agent.
Skip to Appendix F for How to add **Azure** as a Source Connection to the CI Sync (EE) Agent.

After completing the tasks in the relevant Appendix, return here have your ServiceNow SME proceed with Step 6 – Configure your ServiceNow to be ready for CI Sync (EE).

Step 6 - Configure your ServiceNow to be ready for CI Sync (EE)

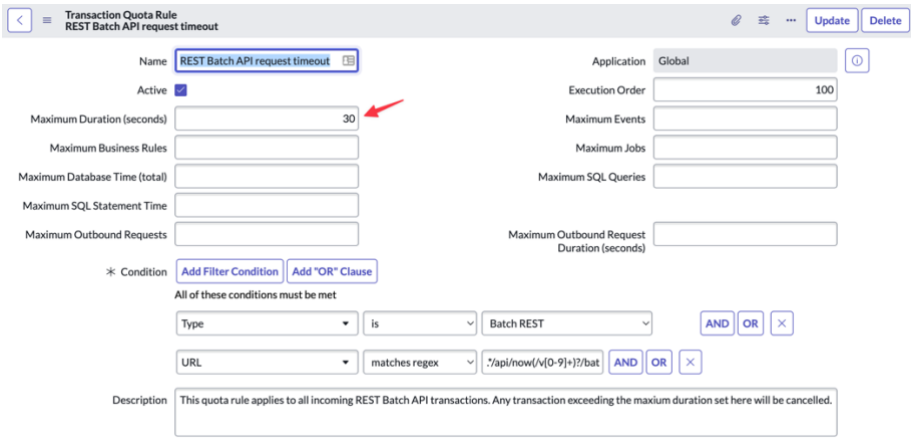
In this step your ServiceNow Admin SME will perform the following tasks

- Task 6a: Configure the Batch API timeout.
- Task 6b: Configure the CMDB CI Software Package Name field for increased performance.
- Task 6c: Create a User Account (to be used by the CI Sync (EE) SaaS application).
- Task 6d: Configure permissions on the new User Account.
- Task 6e: (**Optional**) Configure additional authentication controls for OAuth or MFA.
- Task 6f: **(Optional though recommended)** Update your ServiceNow CI forms to include additional Related Lists populated by CI Sync (EE).

Task 6a: Configure the Batch API timeout.

1. **Navigate** to your ServiceNow at this URL:

   https://**YourInstance**.service-now.com/sysrule_quota_list.do?sysparm_query=nameLIKEbatch
2. From the list**,** locate the entry where the **Name** is "**Rest Batch API request timeout"**
3. Change the "**Maximum Duration (seconds)**" to 60, then click **Update**



Click here to return to steps index

Task 6b: Configure the CMDB CI Software Package Name field for increased performance.

Context Note: The standard configuration of the "Software" (table: cmdb_ci_spkg) table has the "Package name" (field: package_name) field set as the Display field for the table.

The Package name field is a calculated fields which results in the table API performing additional redundant lookups.

Unchecking the Display property results in a ten (10) times improvement in synchronization performance.

There are two options for completing the configuration of your ServiceNow instance:

Option	Summary and Link to Detailed Instructions
Option 1 – Automated Steps using Update Set	• Download a Syncfish supplied Update Set, import, preview and commit. • See subsequent pages for Task 6b (Option 1): Automated Steps using UpdateSet.
Option 2 – Manual Steps using ServiceNow UI Steps	• Your SME uses the ServiceNow UI to manually modify the dictionary configuration and updates the various related list fields. • See subsequent pages for Task 6b (Option 2): Manual Steps using ServiceNow UI.

Each of the above options achieves the same end-result. That is, each option applies the following changes:

- **Updates the dictionary record** to display=False on the table "cmdb_ci_spkg" for column "package_name"
- **Add the "Package name" field to the "List Layout" on** each of **the following CI Class default views:**
  - AIX Server
  - Computer
  - ESX Server
  - HPUX Server
  - Hyper-V Server
  - Linux Server
  - Netware Server
  - OS/X Server
  - Solaris Server
  - UNIX Server
  - Windows Server

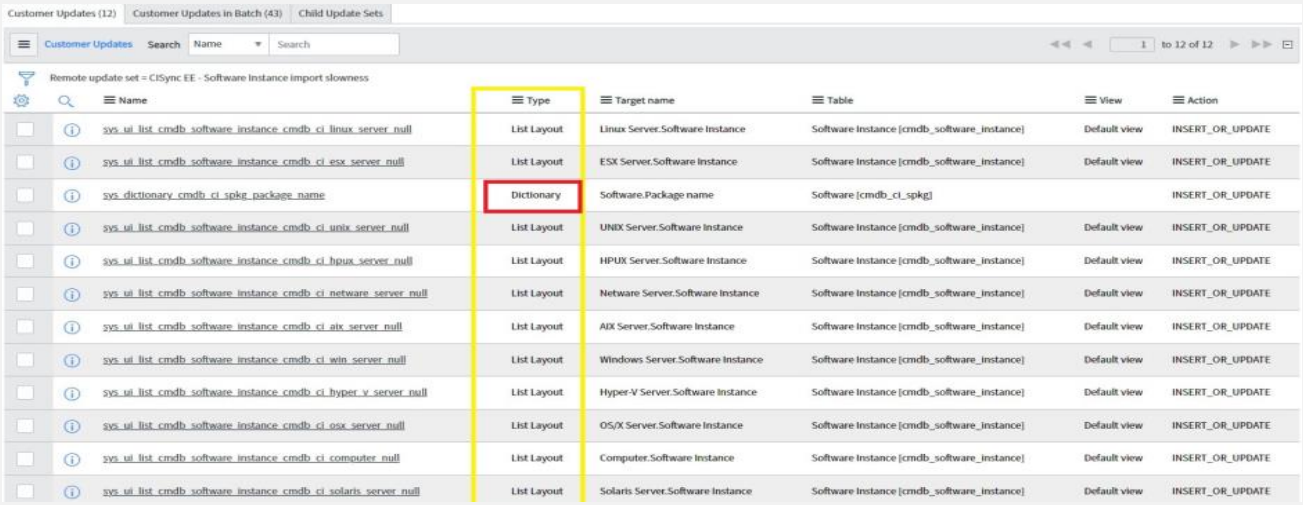
Task 6b (Option 1): Automated Steps using Update Set

Informational Note:

The update set includes two types of update as follows.

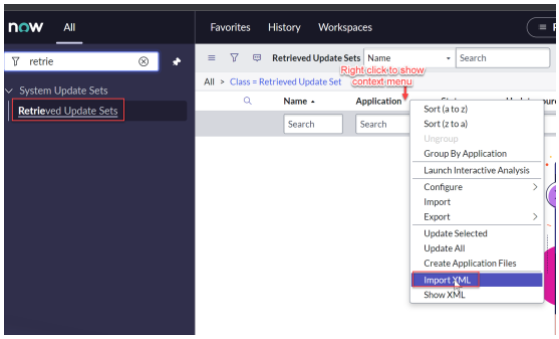
1. It updates to the **Dictionary** on the table "cmdb_ci_spkg".
2. It **adds the "Package name" field to the default "List Layout"** on a few CI Classes

These two updates are shown in the screen below.

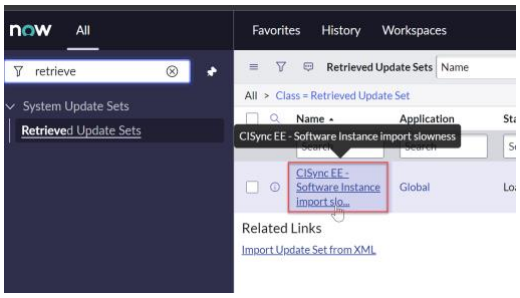


When you preview the update set (as part of Step 9 below) you may see preview errors if the above List Layout have already been modified in your ServiceNow instance. If you receive preview errors and use the instruction in Step 9 to decide how best to proceed. If in doubt contact Syncfish for assistance.

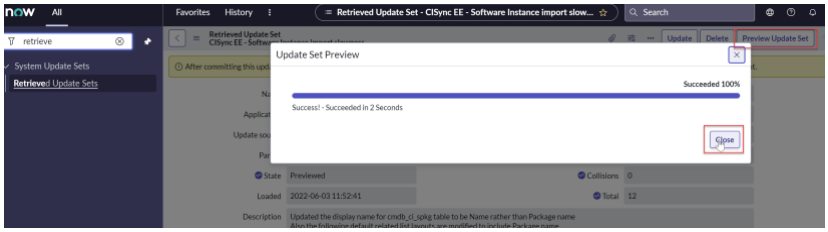
1. **Download the update set** from Syncfish at the below URL: https://downloads.syncfish.app/servicenow/cisync-fix-software_instance-import-slowness.xml
2. Open a browser and navigate to **your ServiceNow instance**
3. In the left nav menu search for "**Retrieved Update Sets**" and click to open
4. Right click on the column heading row and select "**Import XML**"



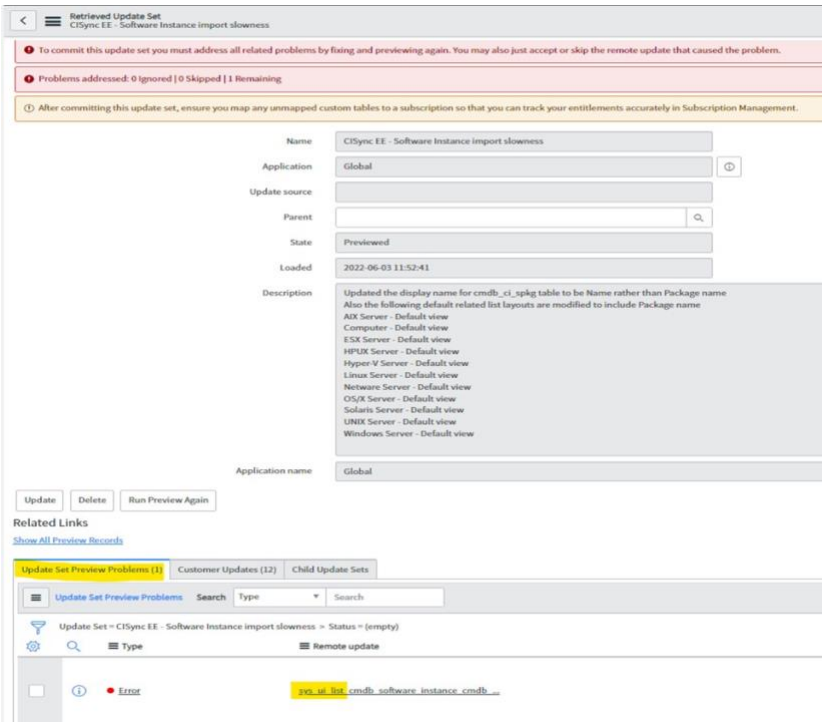
6. Select "**Choose File**"
7. Select the downloaded file "**cisync-fix-software_instance-import-slowness.xml**"
8. Click to **open the Update Set**



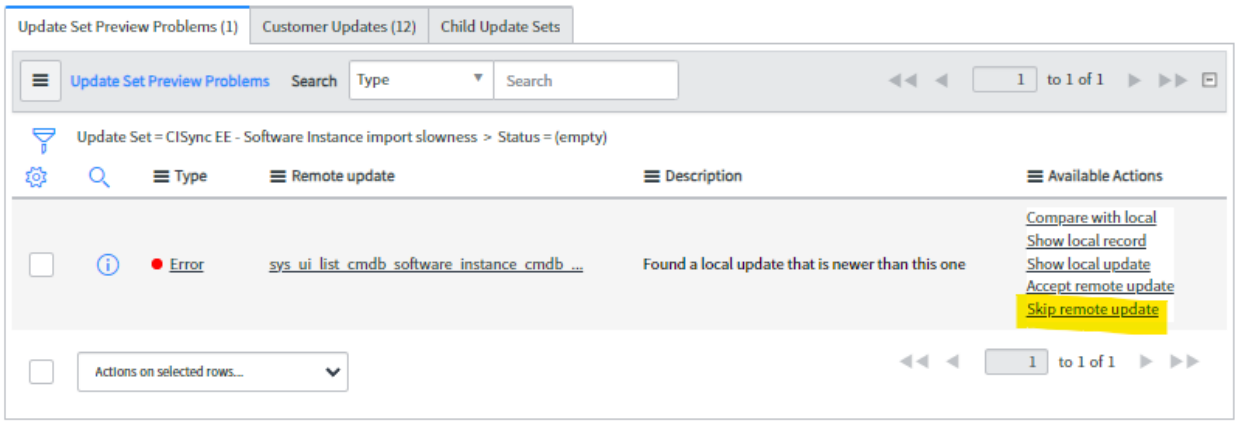
9. Click "**Preview Update Set**"
10. If there are no preview errors, Click "**Close**" and proceed to Step 12 below to Commit the Update Set.



11. If there are preview errors, they are likely to be related to the List Layout update (as described in the Informational Note at the start of this task). Use the details below to diagnose and resolve the errors and if in doubt contact Syncfish for assistance.
    - If there are preview errors, view the errors in the 'Update Set Preview Problems' shown below.



- Make sure all the errors are related to the UI List view updates and the error is not for the Dictionary update then you can Skip the remote update in the Available Actions list.



12. Click "**Commit Update Set**"

> **Guidance Note:** You can skip the next few pages and move directly to Task 6c: Create a User Account (to be used by the CI Sync (EE) SaaS application).

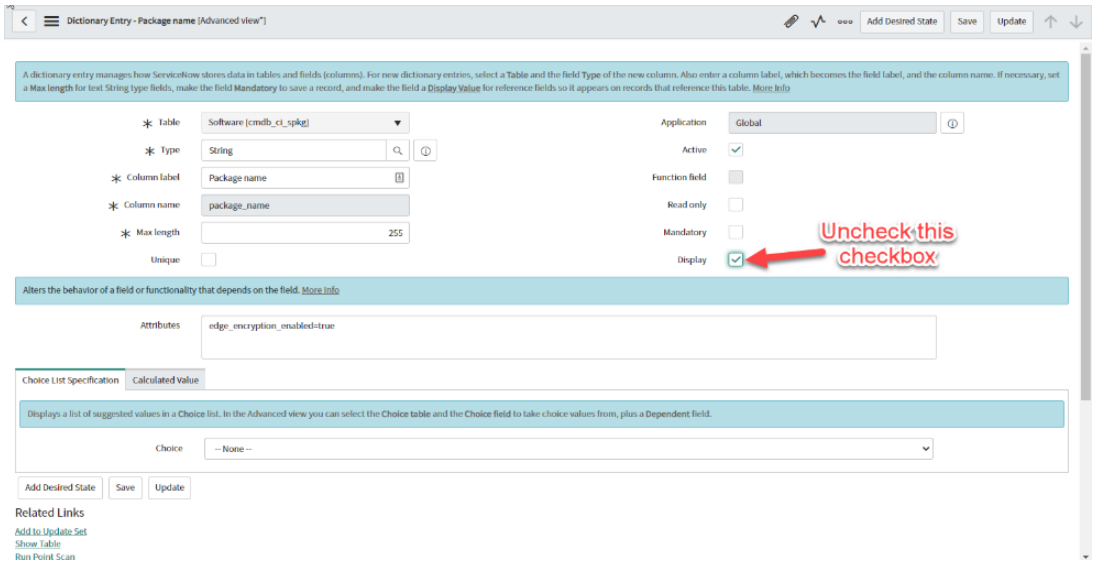
Task 6b (Option 2): Manual Steps using ServiceNow UI

Guidance Note:

Only perform the below steps **if you didn't use Option 1** above (i.e. didn't use the Update Set option). Option 2 consists of bullet points 1 through 15 on the next couple of pages.

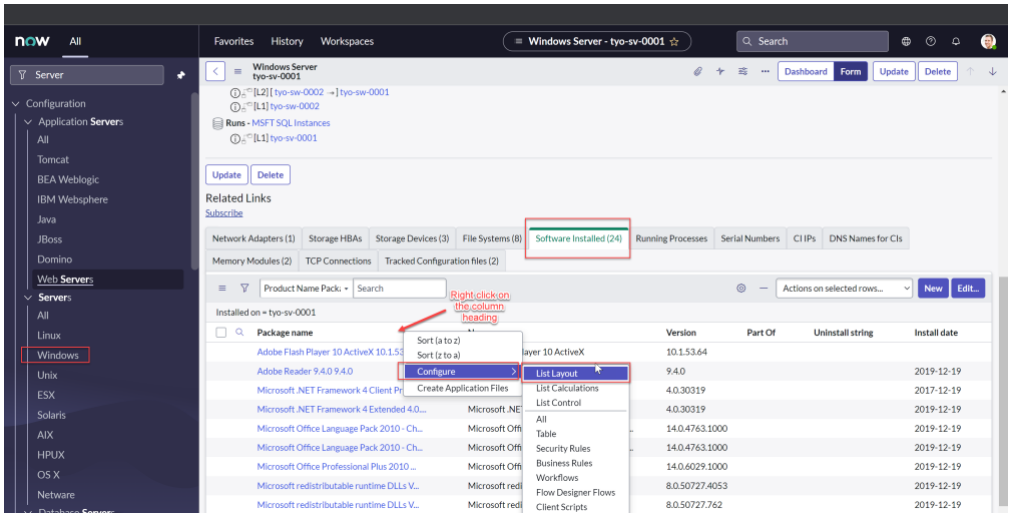
If you have already used Option 1 (the automated method) then skip through all 15 items below and continue from Task 6c: Create a User Account (to be used by the CI Sync (EE) SaaS application) onwards.

1. Navigate to the dictionary record for Table: cmdb_ci_spkg, Column name: package_name using the following URL: https://**yourinstance**.service-now.com/sys_dictionary_list.do?sysparm_query=nameSTARTSWITHcmdb_ci_spkg%5EelementSTARTSWITHpackage_name
2. Open the record
3. Uncheck **Display**
4. Click **Update**

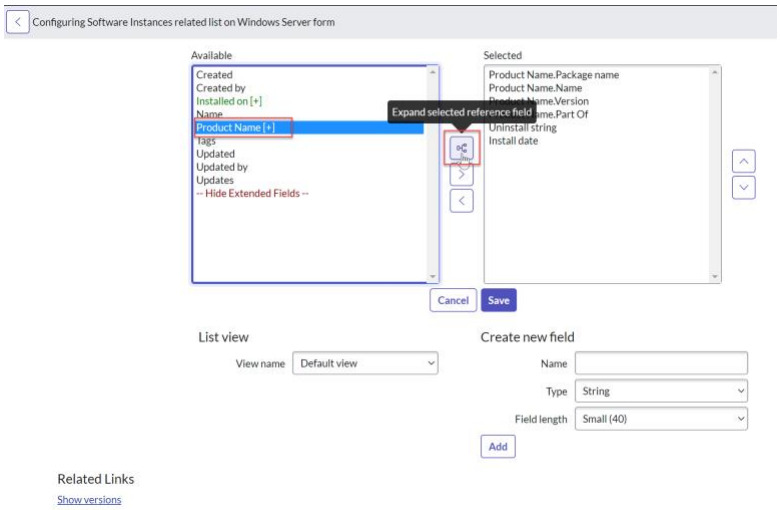


5. Next, you need to update the "**Software Installed**" **Related Lists column layout** for a number of CMDB CI record types. To begin with, execute the steps below to update the **Windows Server – Default view** *(and the repeat these steps for the additional CI record types noted at the end of this step*)

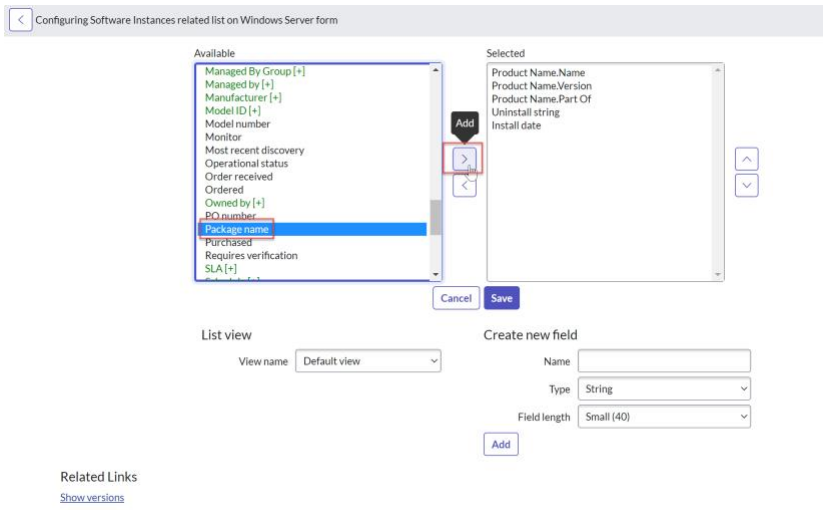
   - Navigate to "**Windows Servers**"
   - Scroll down to the form to see the **Related Lists** (the set of tabs at the bottom of the form)
   - Click "**Software Installed**"
   - Right click the **column heading** (e.g. Package Name) within the "Software Installed" Related List
   - Select "**Configure**"
   - Select "**List Layout**"



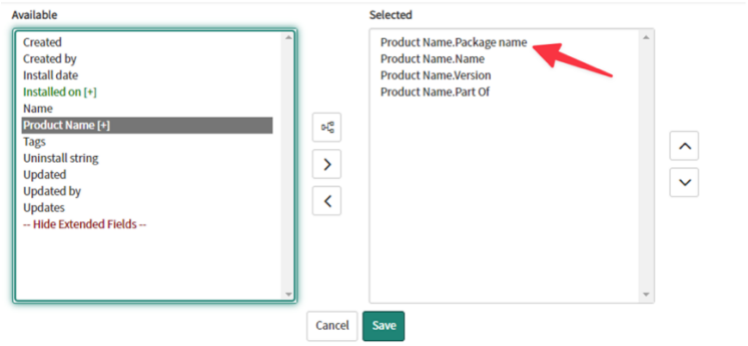
- In the Available list select "**Product Name**"
- Select "**Expand selected reference field**" in the middle buttons between the "Available" list and "Select" list



- Scroll down to find "**Package name**" under the "**.Product Name --> Software Fields**"
- Select "**Package name**"
- Select the "**Add**" button



- "**Package name**" should now be added to the "**Selected**" list on the right-hand side
- Click "Save"



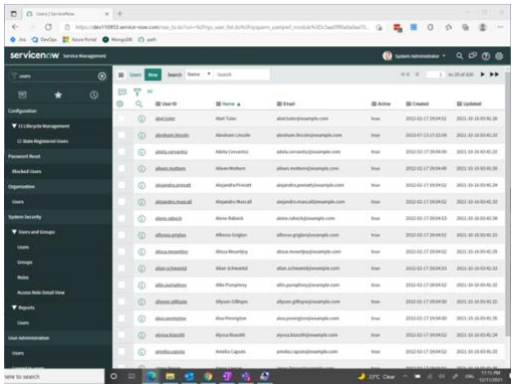
15. **Repeat the above steps** by navigating to each of the following CMDB CI record types and amending each of their "**Software Installed"** Related List
    - AIX Server - Default view
    - Computer - Default view
    - ESX Server - Default view
    - HPUX Server - Default view
    - Hyper-V Server - Default view
    - Linux Server - Default view
    - Netware Server - Default view
    - OS/X Server - Default view
    - Solaris Server - Default view
    - UNIX Server - Default view

Click here to return to steps index

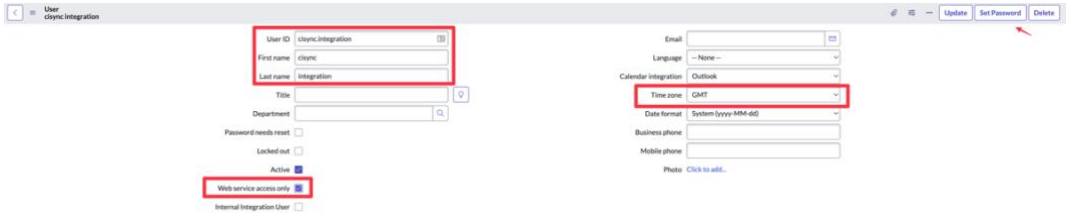
Task 6c: Create a User Account (to be used by the CI Sync (EE) SaaS application)

Create a ServiceNow user account. It is recommended that it be named accordingly so users can identify the records created/updated by the integration. e.g. “cisync.integration”.

1. Log into ServiceNow as an Administrator
2. Navigate to **System Security / Users**



3. Click **New,** then enter the **Username, First name, Last name** and **Password**



Informational Note: The way you set the Password will vary depending on the edition of ServiceNow you are using. The above screen is from San Diego.

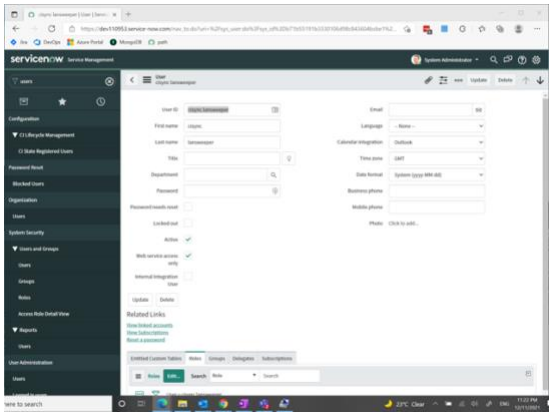
Data Capture Note: Make sure you record Username/Password as they will be required when configuring the ServiceNow connection properties in the CI Sync (EE) platform later in this document.

4. **Tick** the “**Web service access only**” checkbox
5. Set the **Time zone** to **GMT**
6. Click **Submit**

Click here to return to steps index

Task 6d: Configure permissions on the new User Account

1. **Navigate** to the **cisync** user account (e.g. “cisync.integration” or the name you used in Task 6b above)



2. Select the **Roles** tab and click the **Edit…** button
3. **Filter/Select** the roles below and click the **Save** button
   - user_admin
   - personalize_choices
   - snc_platform_rest_api_access
   - cloud_admin
   - tracked_file_reader
   - asset
   - model_manager

Important Note: By granting these ServiceNow Out-of-the-Box (OOTB) Roles you are permitting the cisync user account, and therefore the CI Sync (EE) SaaS application access to your ServiceNow environment to the extent afforded by these roles.

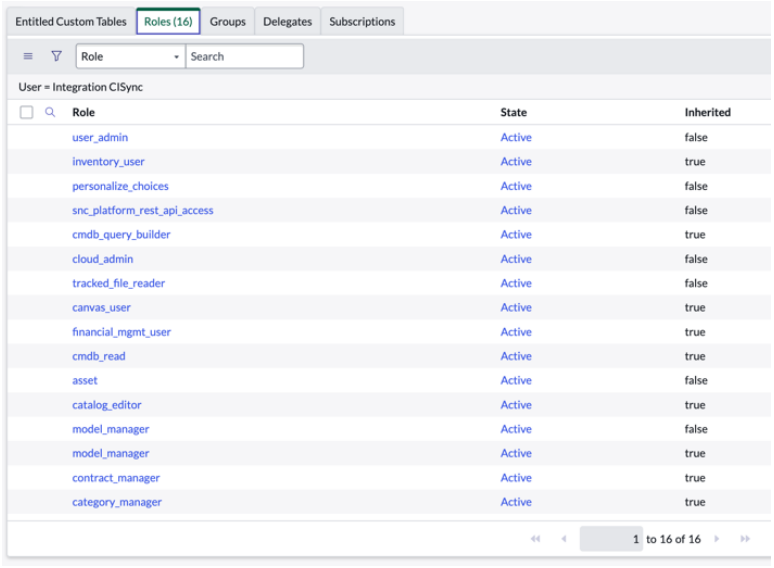
It is probably these OOTB Roles grant CI Sync (EE) access to ServiceNow tables that are not needed or in-scope for CI Sync (EE). A good example is the sys_user table in ServiceNow. The default CI Sync (EE) configuration rules do require the CI Sync (EE) SaaS application to access the sys_user, however the user_admin role grants such access (the user_admin role is needed for access to other reference tables (such as the core_company table) that stores manufacture reference data).

Syncfish recommend you review the ServiceNow system documentation (and system itself) to understand the permissions these roles provide to your CI Sync (EE) SaaS application.

Syncfish provide further details on this topic in the document titled “**CI Sync (EE) - Overview of Source and Destination Fine Grain Permission Option for Personal Data**”. The document also includes non-authoritative guidance on how to assess and potentially apply fine-grain permissions to further restrict CI Sync (EE)’s access within ServiceNow, in particular if your organization has concerns about Personal Data.

Click here to return to steps index

4. Click **Save**. Then use the "Roles" tab to check the above roles and the various inherited ones have been applied by comparing to screen shot below.



Task 6e: (Optional) Configure additional authentication controls for OAuth or MFA

The CI Sync (EE) SaaS application supports the following authentication methods provided by the ServiceNow platform for API integrations:

- Basic Auth
- OAuth
- MFA

Use the sections and steps below to configure additional authentication controls as defined by your organisational requirements.

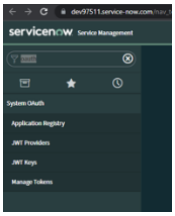
Steps for Basic Auth

If you are using **Basic authentication**, there are **no further configuration steps** required for ServiceNow.

Steps for the Optional Use of OAuth (<u>if you are using Basic Auth or MFA you don't need to do these steps</u>)

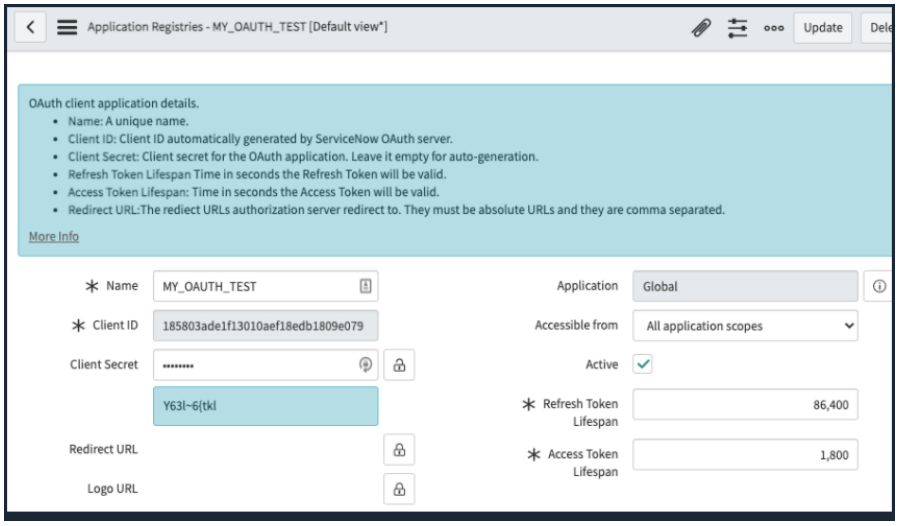
1. Setup in ServiceNow as follows (the hyperlinks below will take you to the related ServiceNow documentation)
    - Activate OAuth
    - Create and endpoint for clients to access the instance

2. Create an OAuth API endpoint for external clients
   a. Search for **OAuth** in the "filter navigation", click on **Application Registry**



   b. Click **New** then "**Create an OAuth API endpoint for external clients**"
   c. Once the OAuth setup is completed, open the application registry record created and copy the **Client ID** and **Secret** to setup the connection in the CI Sync (EE) SaaS application User Interface.

> **Guidance Note:** Click the Client Secret padlock to view the secret as per the following screenshot.



> **Data Capture Note:** Take note of the Client Id and Client Secret as they will be required when configuring the ServiceNow connection within the CI Sync (EE) SaaS application (via the CI Sync (EE) User Interface) later in the document (in Task 7a: Add a ServiceNow Destination Connection and confirm the Source System Connection (via the CI Sync (EE) Agent) is visible and active.).

Steps for the Optional Use of MFA (if you are using Basic Auth or OAuth you don't need to do these steps)

1. Setup in ServiceNow as follows (the hyperlinks below will take you to the related ServiceNow documentation)
   - Exploring Multi-factor authentication
     - Configure user-based multi-factor criteria
     - Log on with multifactor authentication for the first time (will validate the secret in ServiceNow)
2. Once the MFA setup is completed go to the URL below and copy the sys_id of the record just created

   https://yourinstance.service-now.com/user_multifactor_auth_list.do
3. Using the copied sys_id, run the background script below to extract & decrypt the secret.

```
1 var encr = new GlideEncrypter();
2 var grLDAP = new GlideRecord("user_multifactor_auth");
3 grLDAP.get("ENTER_SYS_ID_HERE");
4 var clearString = encr.decrypt(grLDAP.multi_factor_secret);
5 gs.print("TFA secret for " + grLDAP.user.getDisplayValue()+": "+clearString);
```

> **Data Capture Note:** Take note of the secret as it will be required when configuring the ServiceNow connection within the CI Sync (EE) SaaS application (via the CI Sync (EE) User Interface) later in the document.

Click here to return to steps index

Task 6f: (Optional though recommended) Update your ServiceNow CI forms to include additional Related Lists populated by CI Sync (EE).

The default/out-of-the-box configuration rules of CI Sync (EE) populate a number of ServiceNow "Related Lists" for each of the various CI Classes. A number the Related Lists populated by CI Sync (EE) may not be exposed in your ServiceNow UI by default and therefore you should consider updating the ServiceNow CI forms to show the additional Related Lists.

In this section we explain the following:

1. The list of Related Lists populated by CI Sync (EE).
2. How to modify a ServiceNow CI form to expose a new Related List (if it isn't showing already).

The table below shows the Related Lists populated by CI Sync (EE). Not all Related Lists are relevant to all CI Classes (if in doubt try adding the Related List to a given CI in the ServiceNow UI).

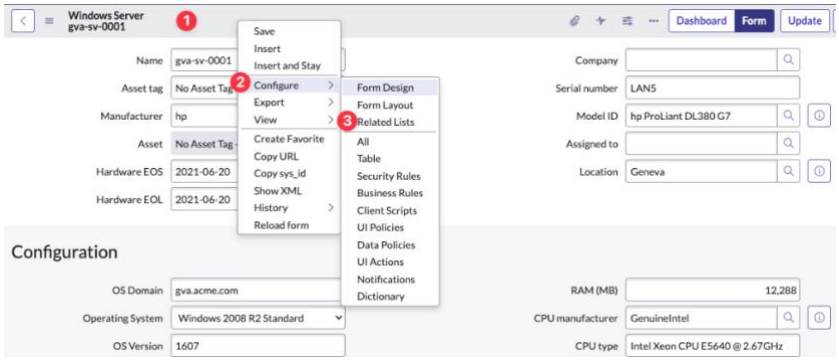
Related List	Related List Name in the ServiceNow UI	Applicable CI Classes
Memory Modules	Memory Module->Configuration Item	Apple Macs Windows PC Windows Server Linux Server
Network Adapters	Network Adapter->Configuration Item	Apple Macs Windows PC Windows Server Linux Server ESXI Server
IP Addresses	CI IPs	Windows Server Linux Server VMWare ESXI Server
Physical Disks	Storage Device-> Computer	Windows PC Windows Server Linux Server
File Systems	File System-> Computer	Windows PC Windows Server Linux Server
Mapped Network Drives	File System->Computer	Windows PC Windows Server
Monitors	Computer Peripheral->Computer	Windows PC Windows Server
Software Installations	Software Installed	Windows PC Windows Server Linux Server
Software (via Airwatch)	Software Installed	Android iPad iPhone
Patches	Patch->Configuration Item	Windows PC Windows Server
Windows Services	Windows Service->Configuration Item	Windows PC Windows Server

Click here to return to steps index

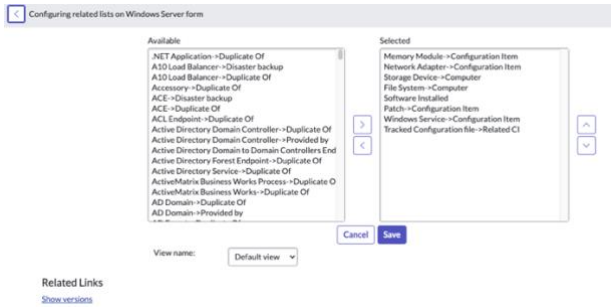
Related List	Related List Name in the ServiceNow UI	Applicable CI Classes
Registry Entries	Tracked Configuration File->Related CI	Windows PC Windows Server
Switch Ports	Switch Port -> CMDB CI	IP Switch
Hyper-V Instances	Hyper-V Virtual Machine Instance->Server	Hyper-V Server
Hyper-V Networks	Hyper-V Virtual Network->Server	Hyper-V Server
Clusters	VMware vCenter Cluster->vCenter Reference	VMWare vCenter
Datacentres	VMware vCenter Datacenter->vCenter Reference	VMWare vCenter
Datastores	VMware vCenter Datastore->vCenter Reference	VMWare vCenter
Virtual Machine Instances	VMware Virtual Machine Instance->vCenter Reference	VMWare vCenter
vCenter Network	VMware vCenter Network->vCenter Reference	VMWare vCenter
VMware Network Adapter	VMware Network Adapter->Configuration Item	VMWare vCenter

Below are basic steps to modify a ServiceNow CI form to expose a new Related List.

1. Login to your ServiceNow instance with Admin permissions.
2. Navigate to any CI in the relevant CI Class (e.g. navigate to a Windows Server CI)
3. **Right-click in the heading area** of the form, then click **Configure** and then **Related Lists** from the sub-menus.



4. Find the Related List you want to expose on the form using the left hand column which lists all **Available** Related Lists. Use the table above to identify the Related Lists populated by CI Sync (EE).
5. Click the Related List and then click add (the **selection arrow)** to move the item to the Selected column and then click **Save.**



Click here to return to steps index

Step 7 – Add your ServiceNow destination connection

In this step, the individual/s responsible for using the CI Sync (EE) User Interface, will perform the following tasks:

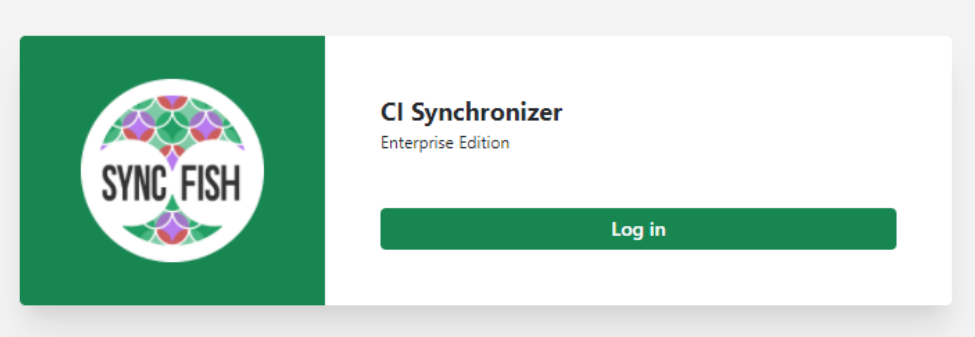
- Task 7a: Add a ServiceNow Destination Connection and confirm the Source System Connection (via the CI Sync (EE) Agent) is visible and active.
- Task 7b: Execute a Test Access check against the ServiceNow instance.
- Task 7c: Execute a Sync Config Test against the ServiceNow instance.

Task 7a: Add a ServiceNow Destination Connection and confirm the Source System Connection (via the CI Sync (EE) Agent) is visible and active.

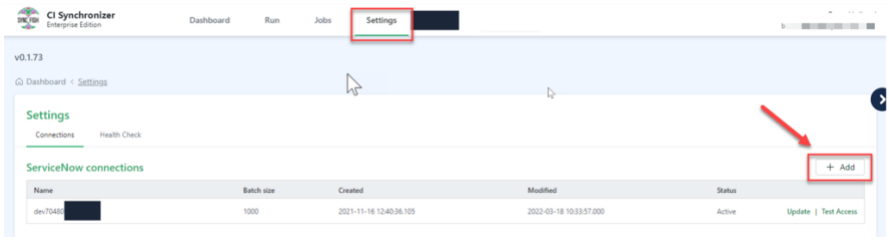
Context Note: In this step you will:

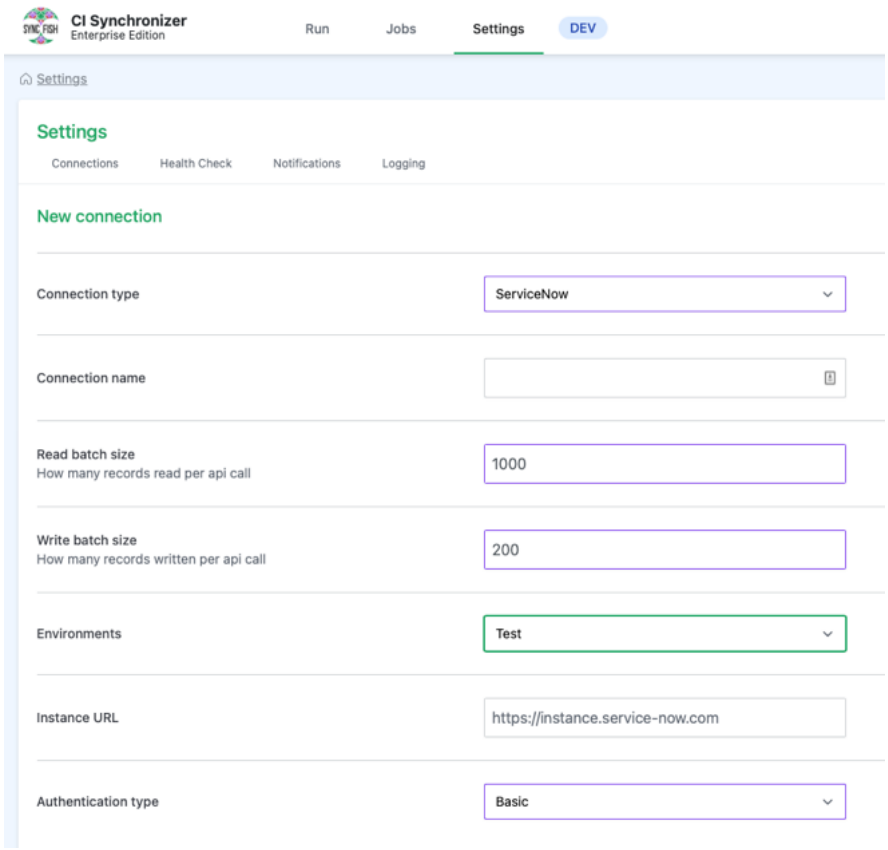
- Add your ServiceNow instance as a Destination Connection in the CI Sync (EE) SaaS application UI and make sure the connection to ServiceNow is active and visible to the CI Sync (EE) SaaS application.
- Check your Source System connection is active and visible to the CI Sync (EE) SaaS application.

1. **Login to your CI Sync (EE) SaaS application user interface** using your company specific URL and with your user account in the same AAD tenancy that the CI Sync (EE) SaaS application was registered in. The URL will be in the format of: https://**YourCo**.syncfish.app



2. Add your ServiceNow instance/connection as follows:
   - Go to the **Settings** page and under the **ServiceNow Connections** heading, click the **[+ Add]** button





   - Enter a friendly name into the **Connection name** field.
   - Leave the default **Read Batch size** of **1000**
   - Leave the default **Write Batch size** of **200**
   - Select the relevant value for the **Environment** related to this connection. The options are:
     - Test
     - Production

> **Informational Note:** The Connection Name and Environment values entered here control the Destinations you see on the "Run" page in the CI Sync (EE) User Interface where you are setting up a CI Sync (EE) job.

Click here to return to steps index

- Enter the **Instance URL** for the ServiceNow instance. e.g. https://**yourinstance**.service-now.com

> **Guidance Note:**
> - If this connection is for your TEST ServiceNow instance, enter the **non-production** ServiceNow instance URL.
> - If this connection is for your PROD ServiceNow instance, enter the **production** ServiceNow instance URL.

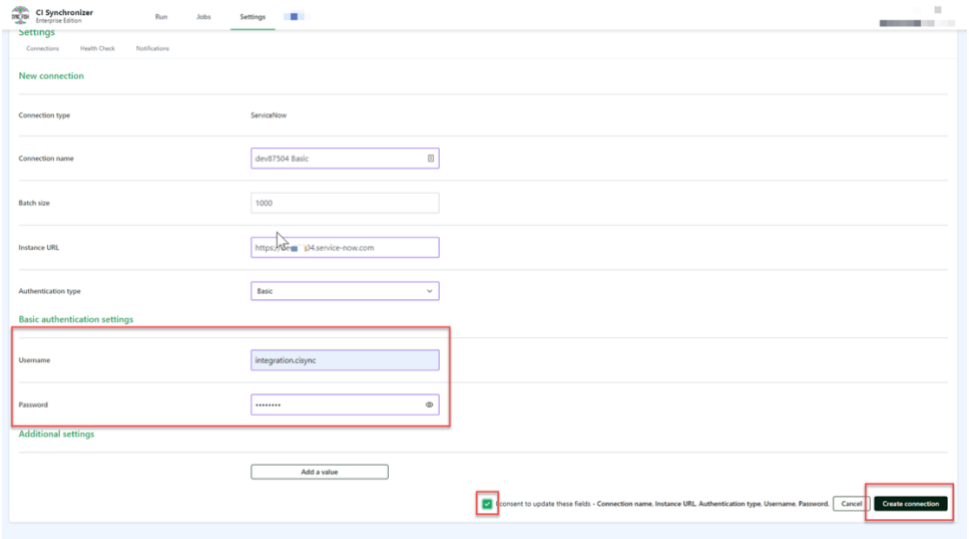
- Select the **Authentication type**
  - **Basic**
  - **OAuth**
  - **Multi-factor Authentication**

Instructions for Basic authentication

- Enter **username, password**

> **Guidance Note:** This is the username and password created in Task 6c: Create a User Account (to be used by the CI Sync (EE) SaaS application).

- Click "**Consent**"
- Click "**Create connection**"

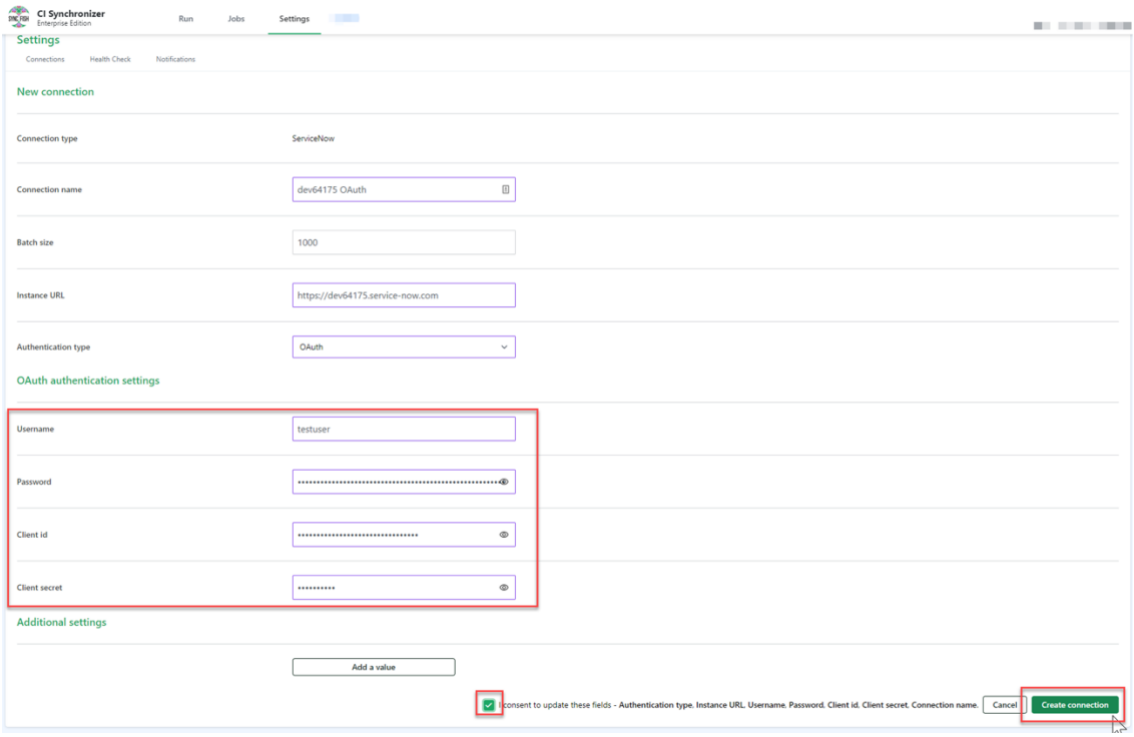


Instructions for OAuth authentication

- Enter **username, password, client id** and **client secret**

> **Guidance Note:** This is the username and password created in Task 6c: Create a User Account (to be used by the CI Sync (EE) SaaS application).

- Click "**Consent**"
- Click "**Create connection**"



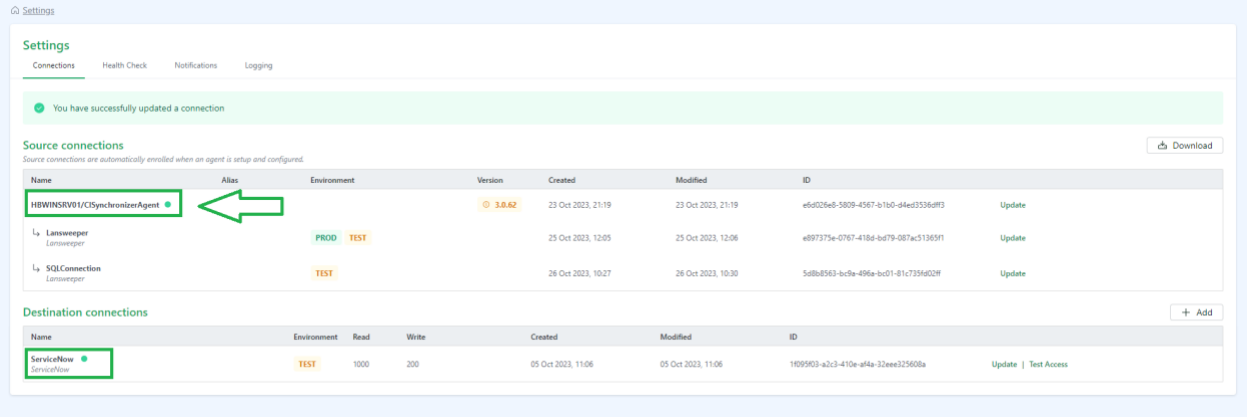
Instructions for Multi-factor authentication

- Enter **username, password,** and **decrypted secret**

> **Guidance Note:** This is the username and password created in Task 6c: Create a User Account (to be used by the CI Sync (EE) SaaS application).

- Click "**Consent**"
- Click "**Create connection**"

7. After finishing the setup of authentication, go back to the **Settings** page and under the **Connections** heading check you have a green light against your ServiceNow connection and your one or more Source System Connections (i.e. the CI Sync (EE) Agent Source System Connections you created during Step 5).



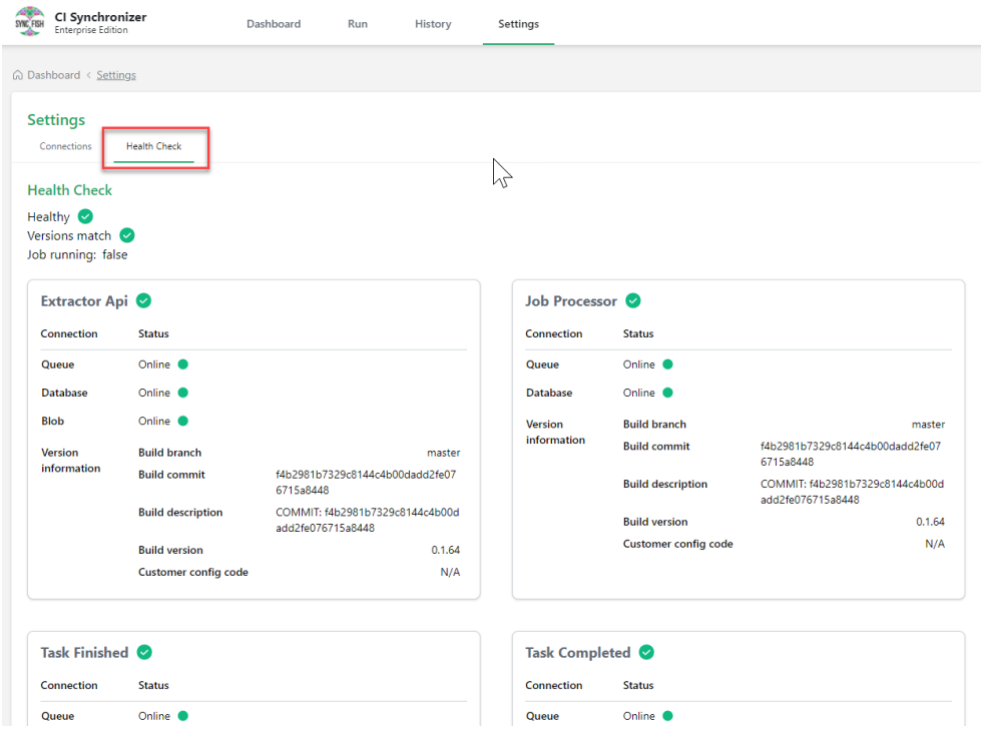
8. If you don't see green against your **ServiceNow connection**, then
   - Recheck the connection settings for ServiceNow in the CI Sync (EE) UI
   - Recheck the authentication details are correct on the ServiceNow end

9. If you don't see green against your **Source System Connection**(s) (**i.e.** CI Sync (EE) Agent Source System Connections you created during Step 5), then
   - Verify the VM which hosts the CI Sync (EE) Agent is up and running
   - Verify the CI Sync (EE) Agent itself (i.e. the windows service) is running
   - Verify the VM which hosts CI Sync (EE) Agent has https access to the Extractor API of your company specific instance of the CI Sync (EE) SaaS application. This can be done by RDP'ing to the VM, opening a browser and accessing the Extract API URL. Ensure no Proxy policy on the server is blocking access to this URL.

> **Guidance Note:** The URL used here is simply your company specific CI Sync (EE) URL with an **/extractor** suffix. For example, the URL will be similar to: https://*YourCo*.syncfish.app**/extractor**

Click here to return to steps index

10. After you have green lights on the source and destination connections, click on the **Health Check** heading and ensure you have green lights/ticks against the various sections on the page



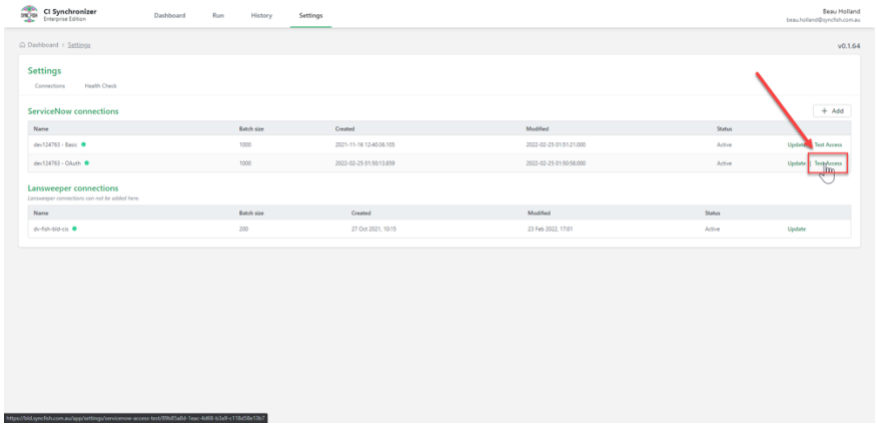
11. If any of the indicators are not green, please contact the Syncfish Support team.

Click here to return to steps index

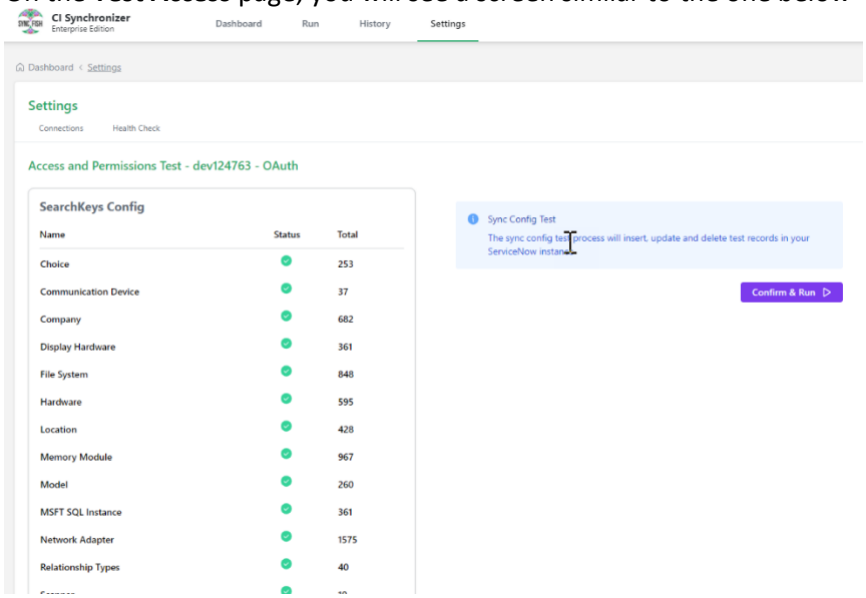
Task 7b: Execute a Test Access check against the ServiceNow instance.

> **Context Note:** In this step you will confirm the ServiceNow user account (i.e. the account configured in the CI Sync (EE) SaaS application) has sufficient permissions to read the various CMDB CI tables.

1. Login to your CI Sync (EE) SaaS application user interface, then:
   - Go to the **Settings** page
   - Locate your **ServiceNow** connection under the **Connections** heading
   - Click the **Test Access** link



1. On the **Test Access** page, you will see a screen similar to the one below



2. If you have **all green lights**, then the **ServiceNow user account** (i.e. the account configured in the CI Sync (EE) SaaS application) has **sufficient permissions to read the various CMDB CI tables**. In this case you can proceed to the next check/test in Task 7c below

Click here to return to steps index

3. If you have **red lights**, then there is likely to be a **permission issue** with the **ServiceNow user account** being used by CI Sync (EE) (i.e. the account is unable to read the various CMDB CI tables). In this case:
   - **Recheck the roles/permissions** assigned to the **user account within ServiceNow.**
   - If you can't resolve the issue, then **contact Syncfish Support**.
   - Also, you can proceed to the next check/test in **Task 7c below** though depending on the issue some of those tests may also fail (also see Information Note below for exceptions).

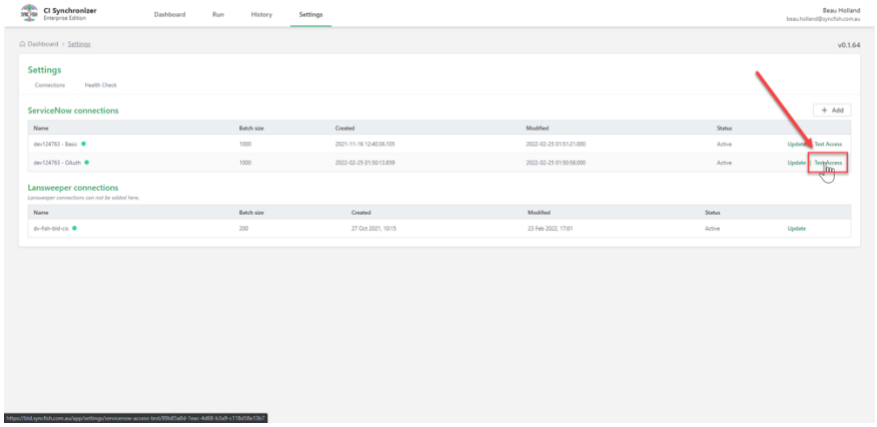
Informational Note
- As at 28th March 2024 - You can safely ignore a red light (failed test) indicator for "Application Services" (if this appears in the list of Search Key tests)

Click here to return to steps index

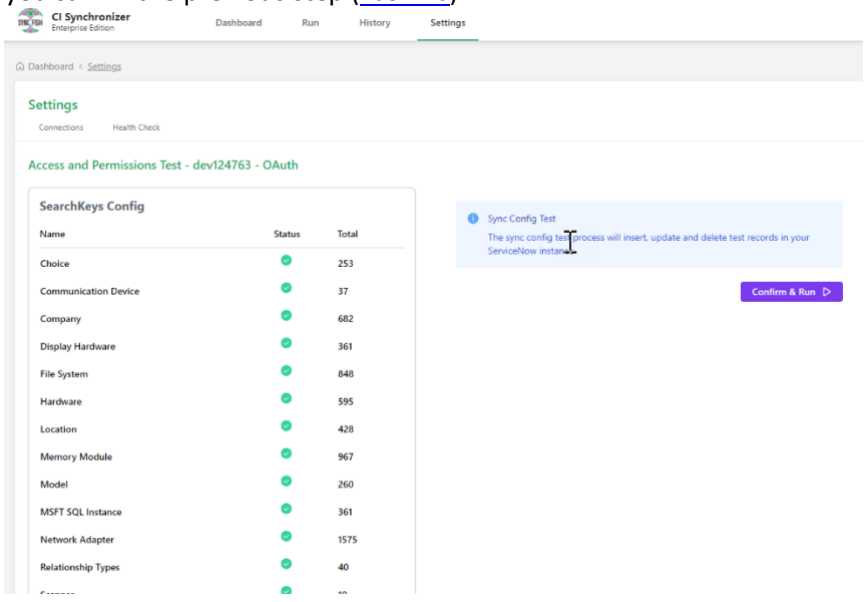
Task 7c: Execute a Sync Config Test against the ServiceNow instance.

Context Note: In this step you will trigger a "dummy" sync of CIs into the ServiceNow CMDB. CI Sync (EE) will attempt to insert, update and delete a small number of CI records into each of the target CMDB CI tables.

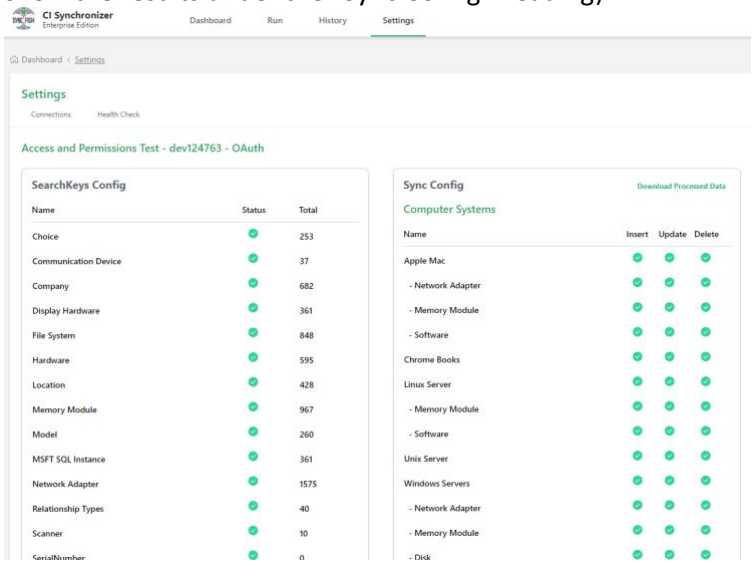
1. Login to your CI Sync (EE) SaaS application user interface, then:
   - Go to the **Settings** page
   - Locate your **ServiceNow** connection under the **Connections** heading
   - Click the **Test Access** link



4. On the **Test Access** page, you will see a screen similar to the one below. This is the same screen you saw in the previous step (Task 7b).



5. Under the "**Sync Config Test**" heading, click the "**Confirm & Run**" button. The test will execute and a results screen similar to one below will be returned (i.e. the right hand side of the screen will show the results under the "Sync Config" heading).



6. If you have **all green lights**, you are ready to go with an initial synchronization of Lansweeper assets into your ServiceNow CMDB. In this case you can proceed to the next section.
7. If you have **red lights**, then there is a **permission or other issue**. In this case:
   - **Recheck the roles/permissions** assigned to the **user account within ServiceNow.**
   - **Check for custom ACLs** applied to the CMDB CI tables in ServiceNow (a custom ACL may be preventing access to CMDB CI tables for the CI Sync integration user account you created).
   - If you can't resolve the issue, then **contact Syncfish Support**. Also, **it's best not to proceed with an initial synchronization** of Lansweeper assets into your ServiceNow CMDB until the failed tests are resolved (see also Information Note below for exceptions).

Informational Note

- You can safely ignore a red light (failed test) indicator for any tests related to "Mapped Network Drives" (this is due to a defect in the access test logic which has been fixed in later versions of CI Sync). Even though the test shows a fail, the actual sync'ing of Mapped Network Drives within a sync job will work just fine.
- You may see red light (failed tests) against one or more "Asset Group Link" related tests. This relates to additional permissions required in ServiceNow for these record types. In Syncfish experience Asset Groups Links are not widely used by Lansweeper/CI Sync customers. If you have a requirement to use/sync Lansweeper Asset Group Links (which are treated as a relationship type in ServiceNow) please contact Syncfish support and we will explain the extra permissions needed in ServiceNow.

Click here to return to steps index

Step 8 – Run your first synchronization and then progressively run more syncs

In this step, the individual/s with login access to the CI Sync (EE) SaaS application user interface, working with your Source System (e.g. Lansweeper, SCCM, InTune, etc) and CMDB colleagues will perform the following tasks.

- Task 8a: Review the relevant CI Sync (EE) Default Configuration Overview document..
- Task 8b: Perform a small initial synchronization.
- Task 8c: Progressively perform more (and more) synchronizations.

Context Note: The CI Sync (EE) User Interface was designed with simplicity and ease of use in mind. As such, getting up and running with initial and ongoing synchronizations should be quick and easy. If not, please let us know!

The tasks in this section are not intended to be a screen-by-screen guide to using the CI Sync (EE) UI. Instead, this section provides general pointers for performing an initial sync (using a small number of records) so you can assess the results in your CMDB before moving on to further/larger synchronization jobs.

Validating the source data (in the relevant source system), and the resulting data in ServiceNow is an important part of thoroughly testing CI Sync (EE) in your environment as you progressively increase the scale, complexity and frequency of your synchronization jobs in CI Sync (EE).

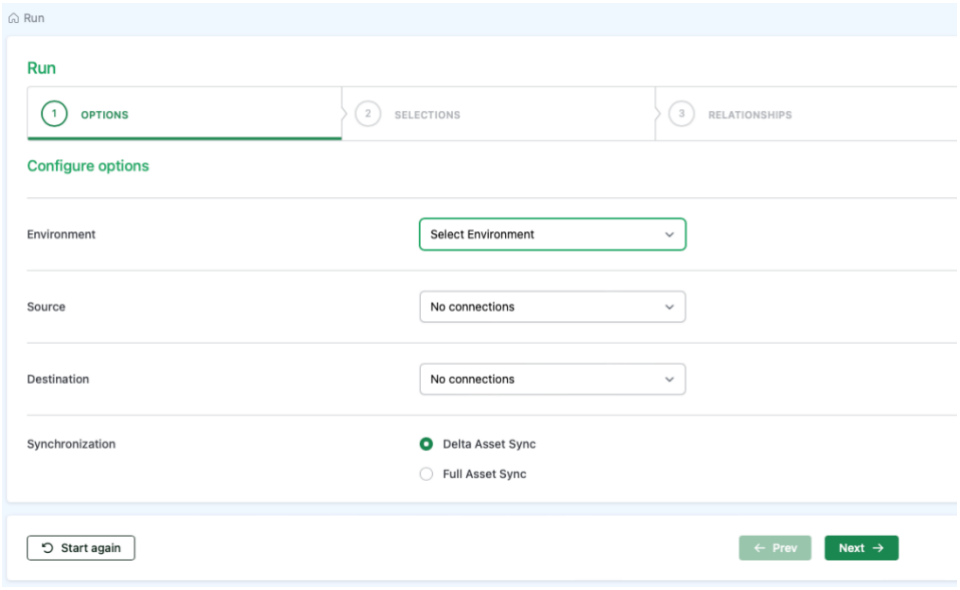
Task 8a: Review the relevant CI Sync (EE) Default Configuration Overview document.

Syncfish publish a “Default Configuration Overview” document specific to a given Source and Destination System. These guides cover various topics about the CI Sync (EE) Default Configuration rules, and in particular how the configuration rules will read, map, transform and interact with data and objects between a given Source System and Destination System. If you haven’t already been provided with the relevant Default Configuration Overview document, please contact Syncfish.

Click here to return to steps index

Task 8b: Perform a small initial synchronization

1. Login to the **CI Sync (EE) UI** and navigate to the **Run** page



2. Select the **Environment** first. You can select either Test or Production. The selection you make here will filter the list of available Source and Destination connections you can select in the subsequent drop-down boxes.
3. Select the **source** and **destination** from the drop-down lists. Your source is any one of your CI Sync (EE) Agents/Connections (which represent the given source asset repository) and the destination is your ServiceNow CMDB.

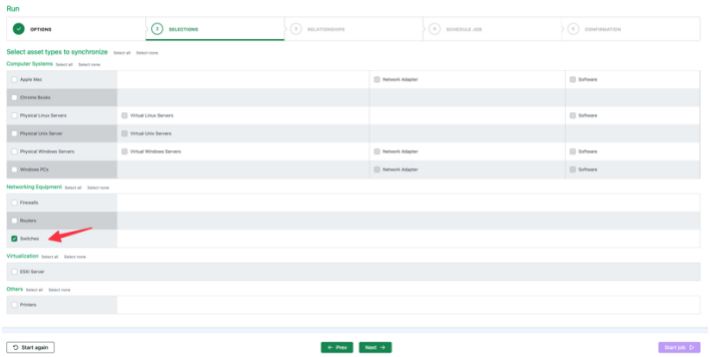
   The drop-down lists are populated from the connections defined in the **Settings** page. Also, the drop-down lists are filtered based on the **Environment** you set against each source and destination connection using the Settings page.
   Make sure you are aware of the destination you are synchronizing to, in particular if the destination is production!

4. Select either **Delta Asset Sync** or **Full Asset Sync** as the job type, then click **Next**

> **Guidance Note:** For **initial/first time** synchronizations a **Delta sync** (i.e. what's changed since the previous sync) will be the **same as a Full sync** (because there has never been a "previous" sync). As such you can select either Delta or Full sync and the result will be the same. For ongoing sync jobs, you should use Delta syncs in the vast majority of cases.

Click here to return to steps index

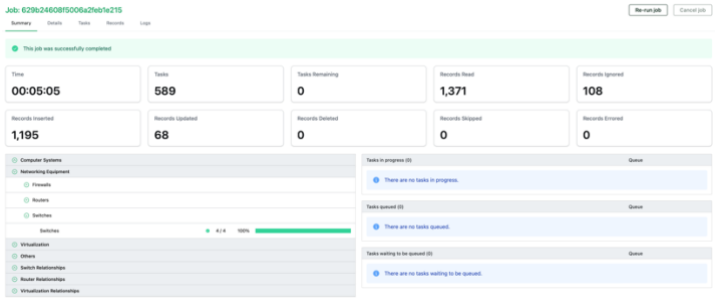
5. On the **(2) Selections** page tick one (maybe two) asset types that you know **contain a small number of records** (e.g. Switches). The idea is to create a small sync job as a basic end-to-end test. When ready, click **Next**



6. Depending on the asset types you ticked on the (2) Selections page you will see all possible relationships presented on the (3) Relationships page. **Use the defaults** which have been selected and then click **Next**

7. Ensure the "**Run now**" option is selected on the **(4) Schedule Job** page, cand then click **Next**

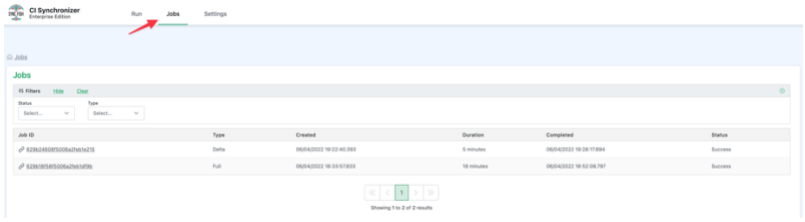
8. Review the details on the **(5) Confirmation** page and when ready click "**Start job**". The job will take a few moments to begin, and you will see various metrics and visual indicators of the job progress. Upon completion you see if the job completed successfully or not.



9. Depending on the results presented by CI Sync (EE), you can/should also **review the CIs in your ServiceNow CMDB** to confirm the expected data has been sent across.

Click here to return to steps index

10. You can also use the **Jobs** page in the CI Sync (EE) User Interface to **drill down into the Job Logs** at varying levels of detail.

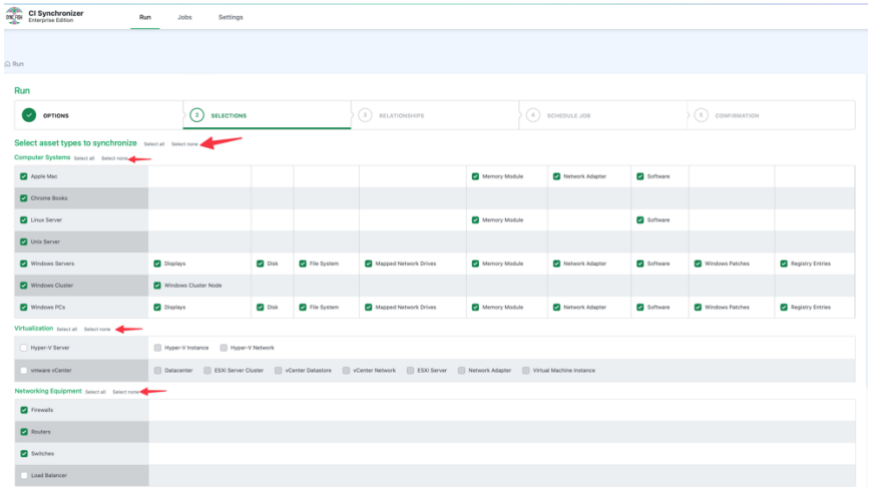


> **Guidance Note:** The Job Logs can be a bit overwhelming on first review as they contain a great deal of information to assist Syncfish diagnosing issues. If your first job has errors which aren't immediately obvious, please contact the Syncfish team for assistance.

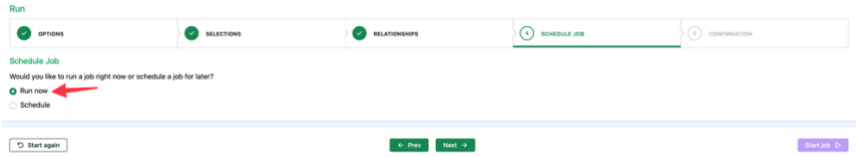
11. Once you are comfortable with the CI Sync (EE) and you have **successfully sync'd a small number of records without error** (or with explainable error/s), **proceed to the next task** to **synchronize more records** and to **create** one or more **Scheduled Jobs**.

Task 8c: Progressively perform more (and more) synchronizations.

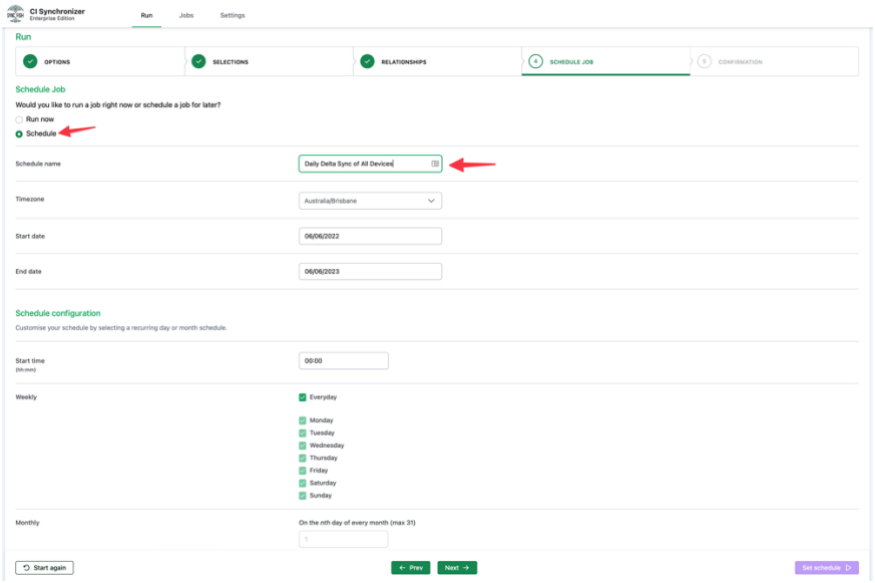
1. Use the **(2) Selections** page in the **CI Sync (EE) UI** to increase the number of asset types you are selecting for synchronization. Use the "**Select all**" and "**Select none**" links at the very top of the page or in each asset type section to make selections. Also tick/untick any of the related records such as displays, disks, etc.



2. Syncfish recommend you continue to use the "**Run now**" option as you **build up/select to more and more asset types** (and relationships) being synchronized. Ensure you are **frequently checking the resulting CIs in your ServiceNow CMDB** and comparing to the source data in Lansweeper.



3. Once you've executed and verified enough synchronizations to be confident about the results, use the **(4) Schedule Job** page to create one or more scheduled jobs based on the selections you've made on the previous pages. Scroll the entire page to see all of the options and the schedule detail which aim to demystify time zone and time of day topics from your choices. When ready click the "**Set schedule**" button in the bottom right of the page.



Guidance Note: You can modify and delete scheduled jobs after they have been created.

Click here to return to steps index

Appendix A – Understanding how the CI Sync (EE) Agent authenticates to SQL Server during the normal operational of the Agent

This appendix shows several example diagrams to explain how the CI Sync (EE) Agent authenticates to SQL Server during the normal/ongoing operation of the agent.

The CI Sync (EE) Agent uses a SQL Server (and therefore needs to authenticate to it) for the following two purposes:

1. To read and write to the RecVer database. The RecVer database is the CI Sync (EE) Agent's delta record tracking SQL DB.
2. To read only from data sources (i.e. synchronization data sources) from products such as Lansweeper and SCCM (i.e. SQL based data sources).

There are two topologies described below:

1. Where the CI Sync (EE) Agent has been deployed on the same server as your Source System SQL Server.
2. Or, where the CI Sync (EE) Agent has been deployed on a separate server to the Source System SQL Server**.**

The topology you use (e.g. same server or separate servers) affects the options you can use for authentication between the CI Sync (EE) Agent and the SQL Server. This table explains the options:

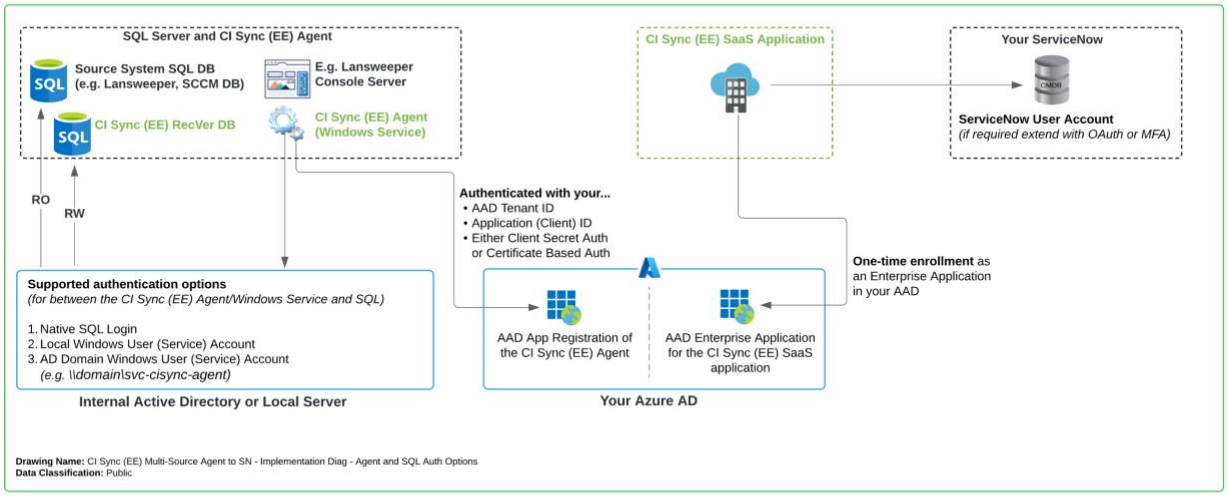
Deployment Scenario	**SQL Native Authentication**	**Local Windows User Account**	**AD Domain Windows User Account**
CI Sync (EE) Agent and SQL are on the **same server**.	Supported	Supported	Supported
CI Sync (EE) Agent and SQL are on **separate servers**.	Supported	Not supported	Supported

The following page shows a diagrammatic view of these options.

Click here to return to steps index

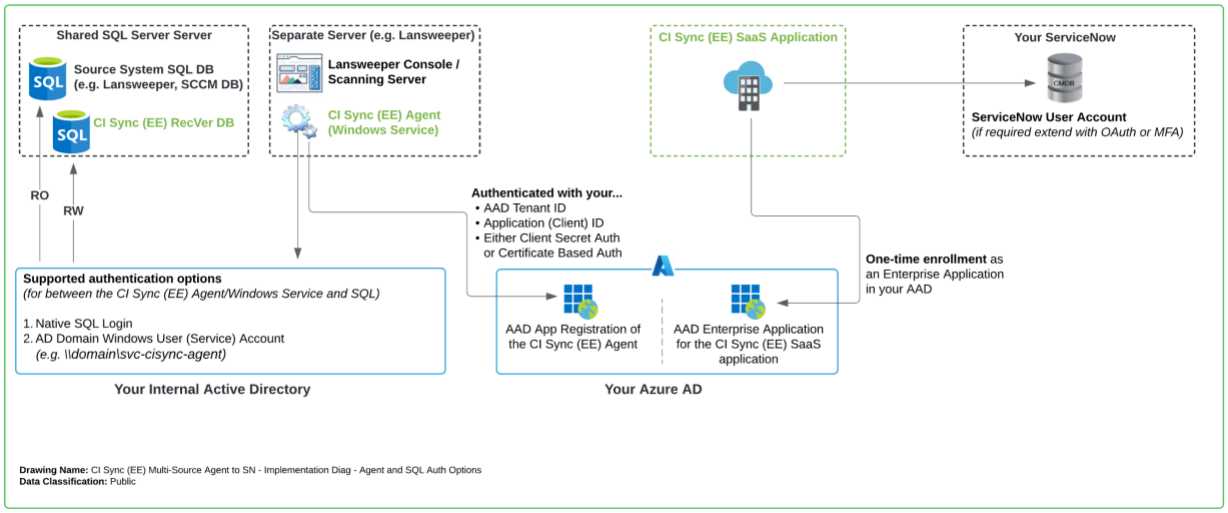
CI Sync (EE) Agent and SQL on the same server

CI Sync (EE) Agent and SQL are on the same server



CI Sync (EE) Agent and SQL on separate servers

CI Sync (EE) Agent and SQL are on separate servers



Appendix B – Configure SQL Maintenance Plans on SQL database(s) for SQL based Source Systems

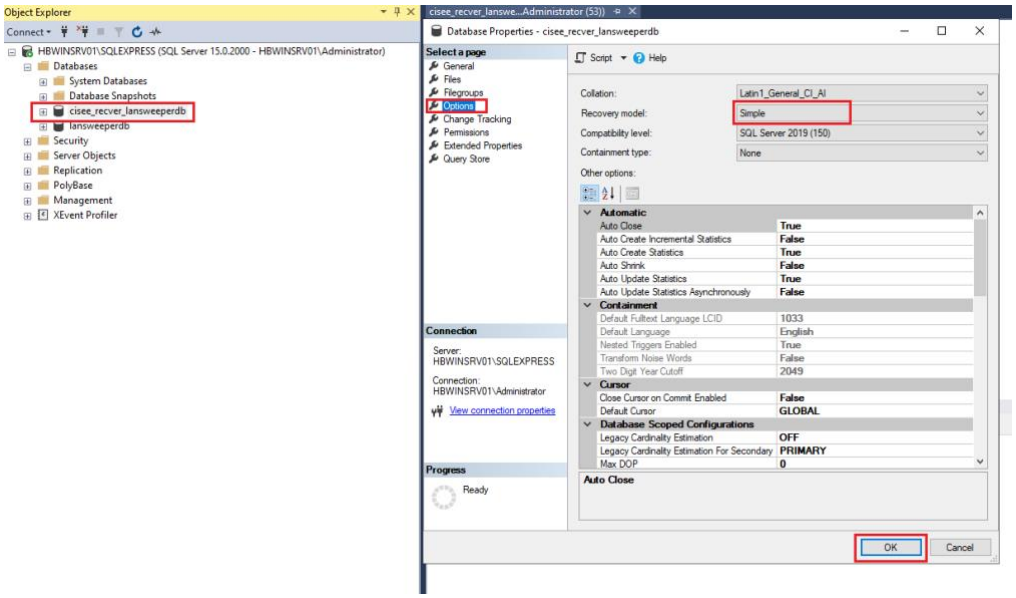
In this step your SQL SME will perform the following tasks

- Task B1: Validate the CI Sync (EE) RecVer database is in Simple Recovery Mode.
- Task B2: Setup a Rebuild Index Maintenance Plan (on the RecVer DB & potentially the Source System DB).
- Task B3: Setup an Update Statistics Maintenance Plan (on the RecVer DB & potentially the Source System DB).
- Task B4: Assess other suggestions for SQL database health.

Task B1: Validate the CI Sync (EE) RecVer database is in Simple Recovery Mode.

> **Informational Note:** Simple recovery mode prevents the creation of transaction log files which could impact disk space over time.

1. Login to SQL Server and open **SQL Server Management Studio** and expand **Databases**, then
   - **Right-click on the CI Sync (EE) RecVer database** (named accordingly when it was created as part of setting up a Source System Connection during Step 5) and select **Properties**.
   - Click the **Options** page
   - Check the **Recover Mode** of the RecVer database and make sure it's set to **Simple** as the **Recovery Mode** (if it isn't then set to "Simple")
   - Click **OK**

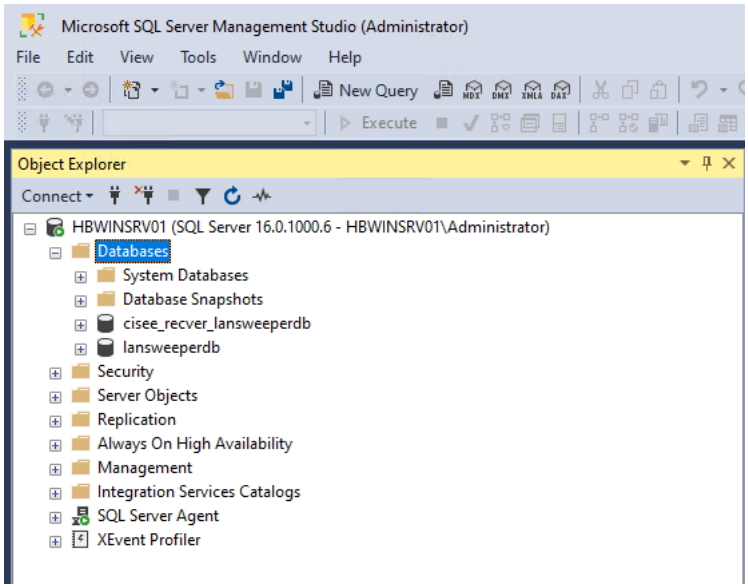


Click here to return to steps index

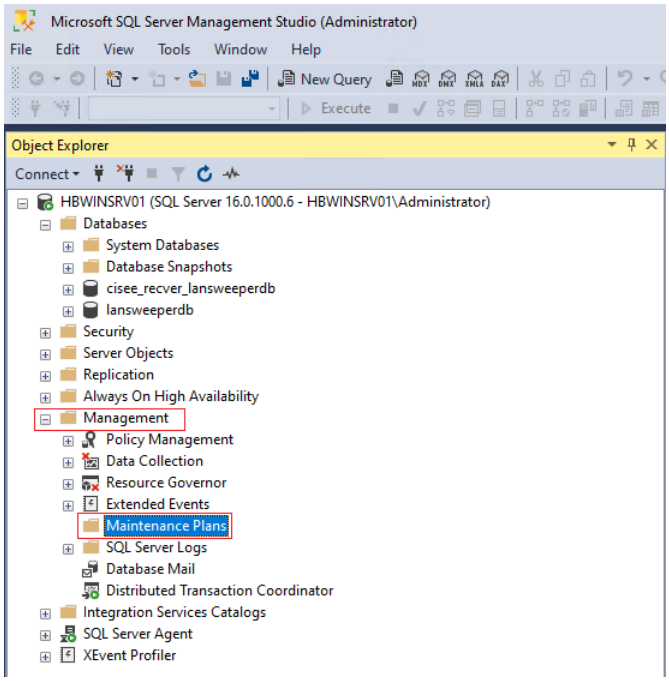
Task B2: Setup a Rebuild Index Maintenance Plan (on the RecVer DB & potentially the Source System DB).

Informational Note: This procedure assumes SQL Server Management Studio is installed setup and running on the same server as the SQL Databases.

1. Login to SQL Server and open **SQL Server Management Studio** and expand **Databases**

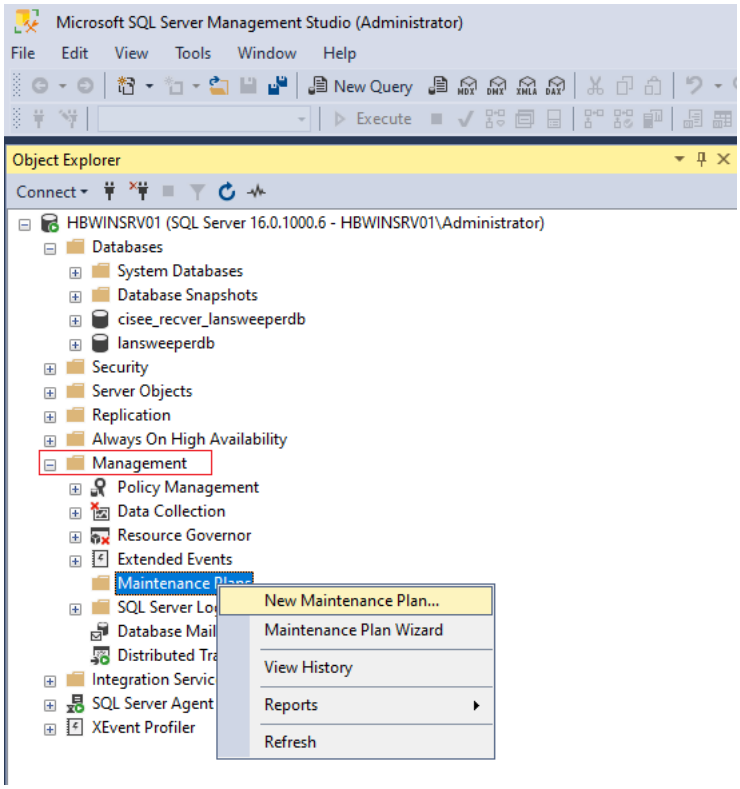


2. Expand **Management** then click on **Maintenance Plans**

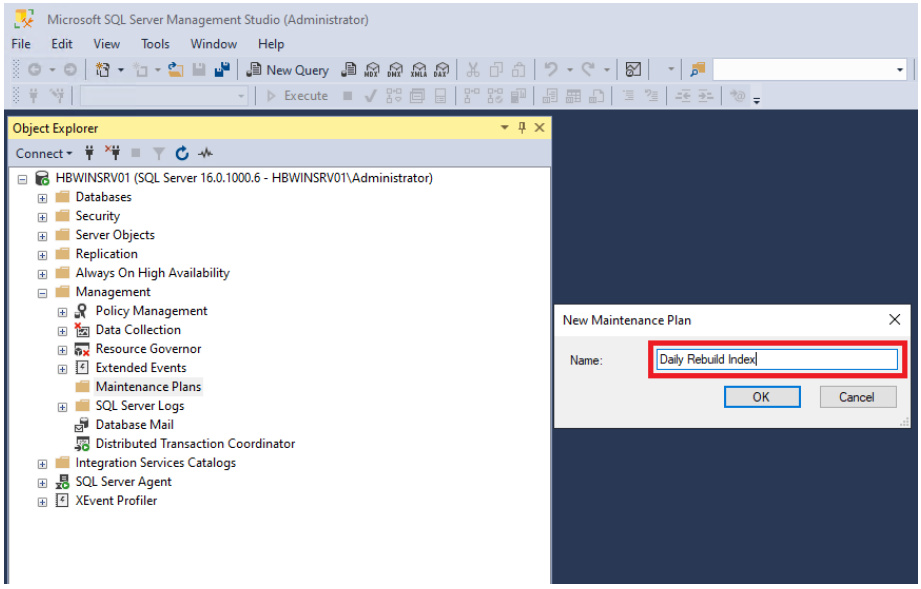


Click here to return to steps index

3. Right click on **Maintenance Plans** and click **New Maintenance Plan**



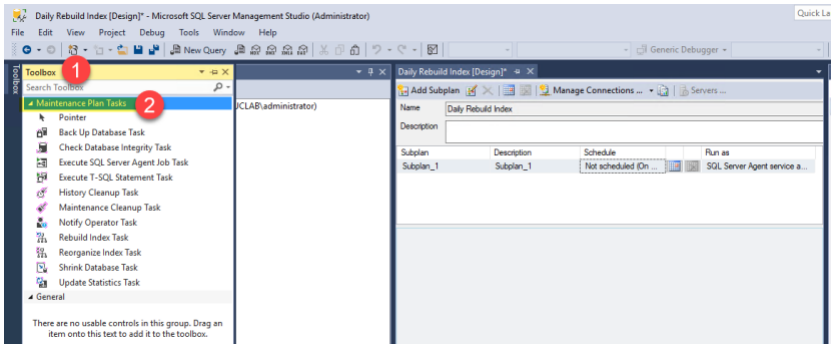
4. Name the plan **Daily Rebuild Index** and click **OK**



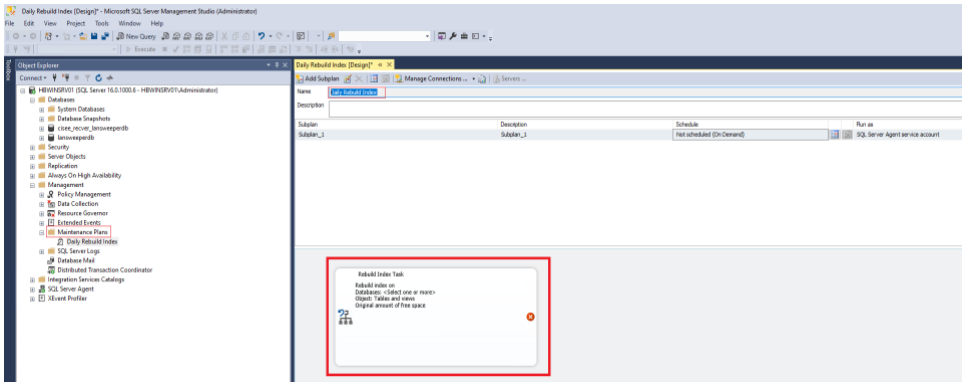
Click here to return to steps index

5. On the left of SQL click on **Toolbox** and expand **Maintenance Plan Tasks**

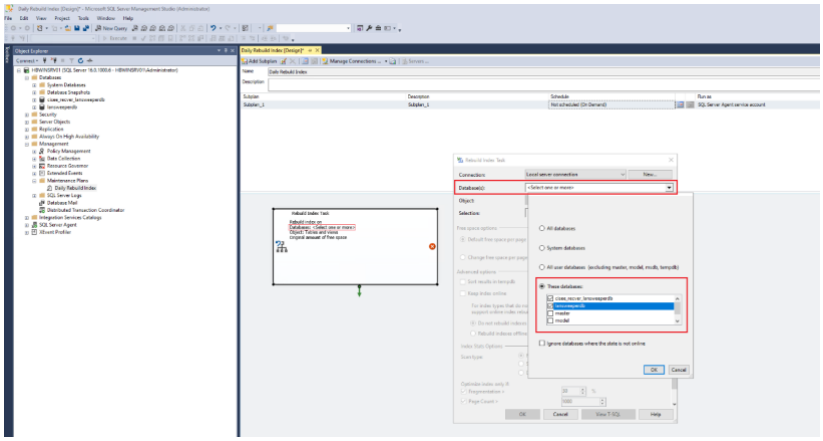
Guidance Note: If you cannot see Toolbox then on the SQL menu click View then Toolbox



6. Click on **Rebuild Index Task** and drag it to the new plan

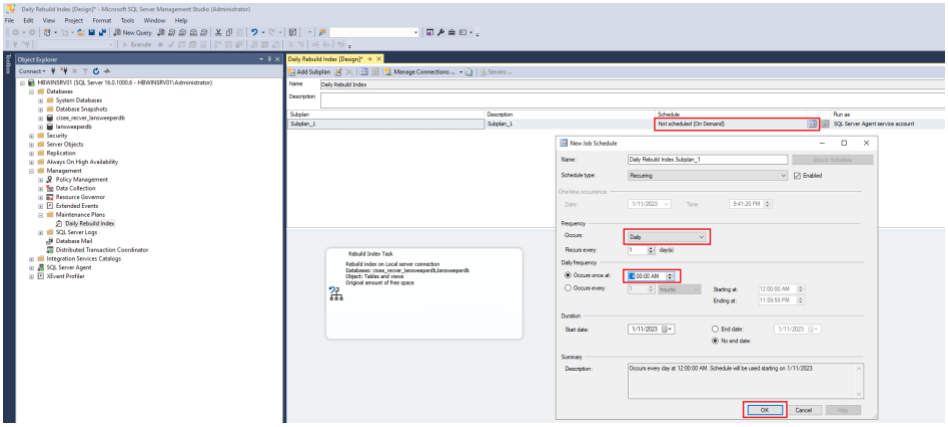


7. Double Click on the **Rebuild Index Task** then:
   - Drop down **Databases**
   - Select the RecVer database (and if using Lansweeper Syncfish recommend you also select the **lansweeperdb** database)
   - Click **OK**
   - Then again click **OK**

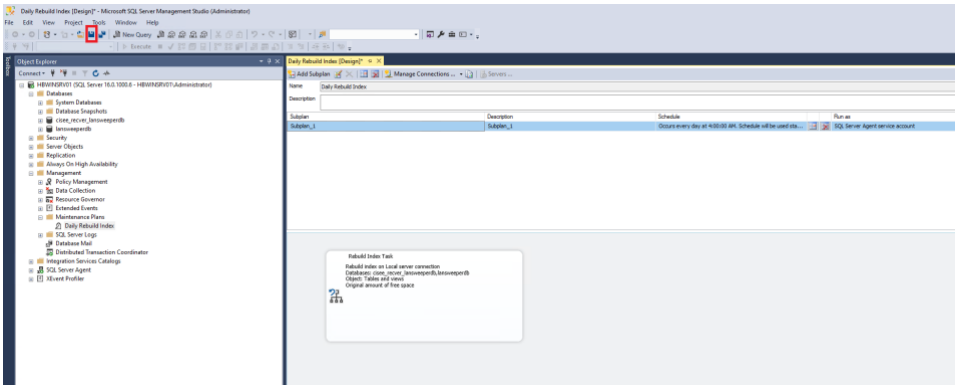


8. Click on the **Schedule** and set to **Daily** and chose a **Time of Day** to run the job. Then click **OK**

Guidance Note: Choose a time that does not clash with Backups and other tasks being run on the SQL Server.



8. Click on **Save**

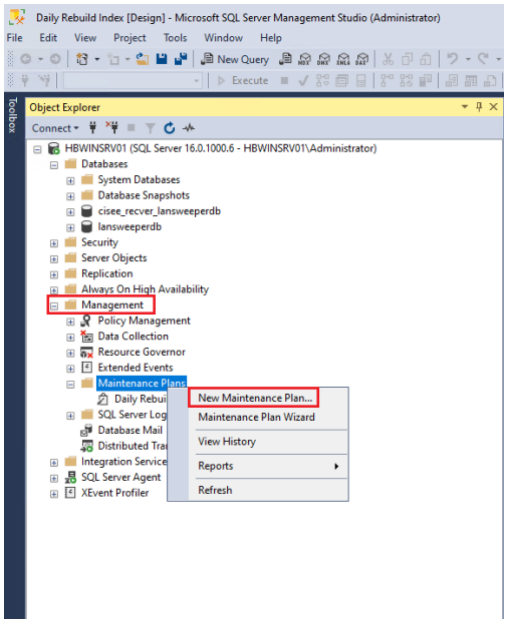


Click here to return to steps index

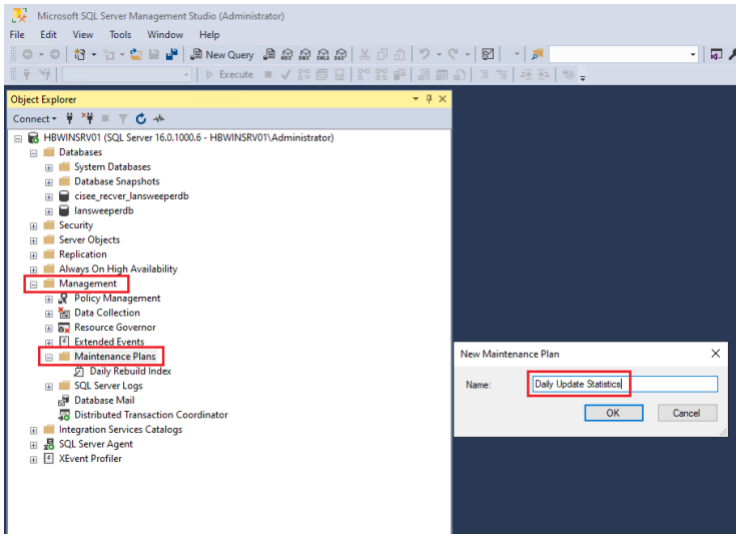
Task B3: Setup an Update Statistics Maintenance Plan (on the RecVer DB & potentially the Source System DB).

Informational Note: This procedure assumes SQL Server Management Studio is installed setup and running on the same server as the SQL Databases.

1. Login to SQL Server and open SQL Server Management Studio**, then:**
   - **Expand** Management
   - **Right click on** Maintenance Plans
   - **Select** New Maintenance Plan

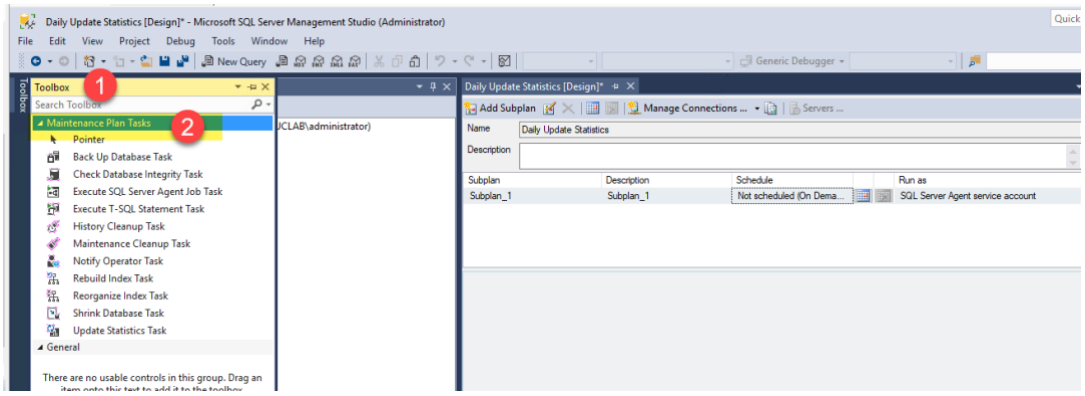


2. Name the plan **Daily Update Statistics** and click **OK**

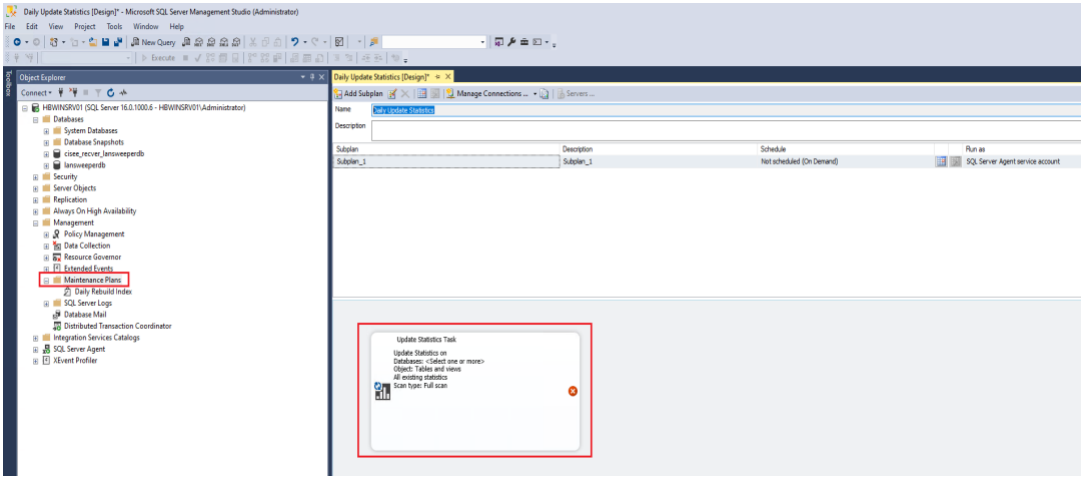


Click here to return to steps index

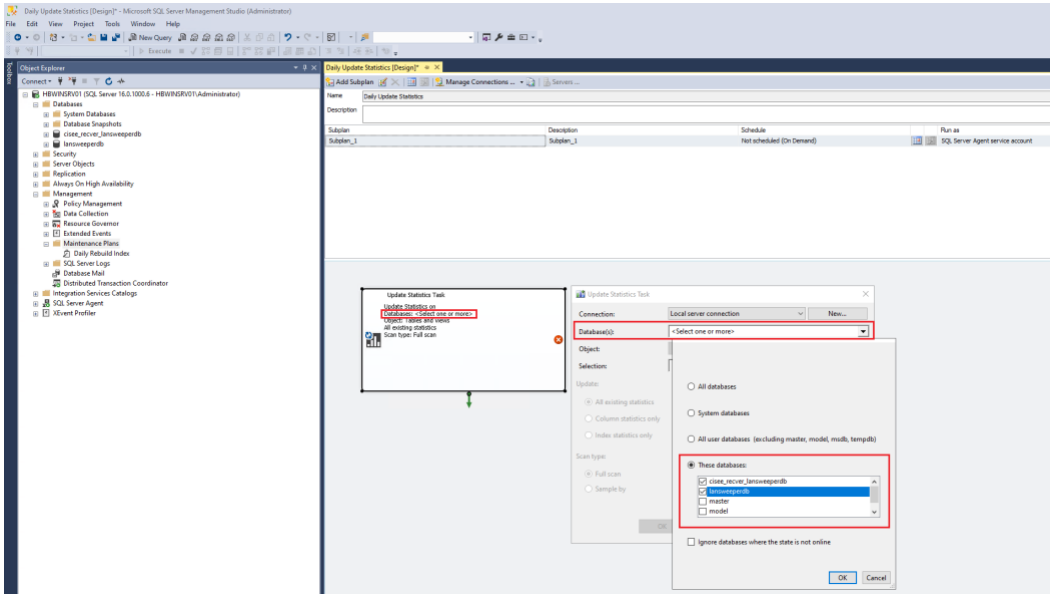
3. On the left of SQL click on **Toolbox** and expand **Maintenance Plan Tasks**



4. Click on **Update Statistics Task** and drag it to the new plan

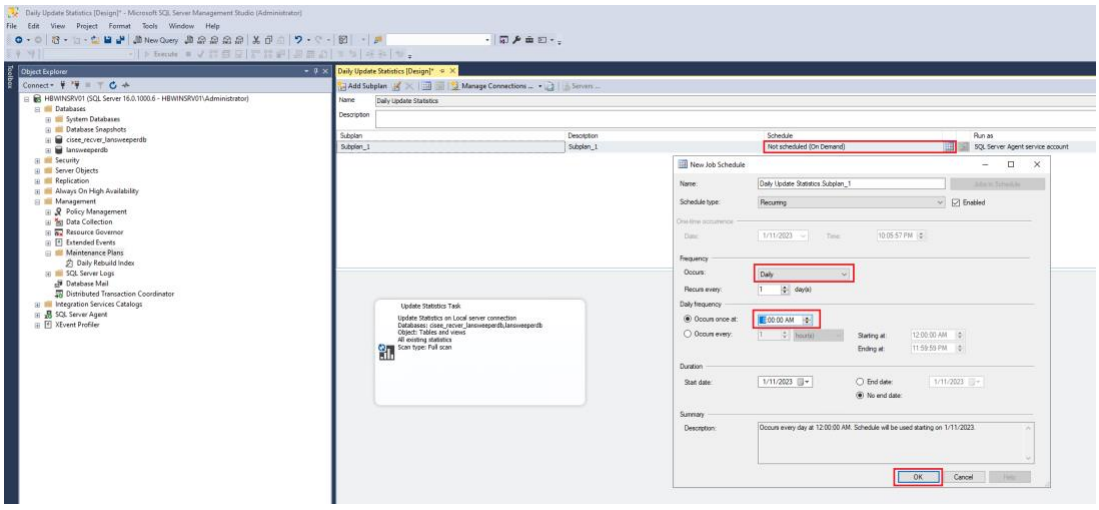


5. Double click on the **Update Statistics Task** then
   - Drop down **Databases**
   - Select the RecVer database (and if using Lansweeper Syncfish recommend you also select the **lansweeperdb** database)
   - Click **OK**
   - Then again click **OK**

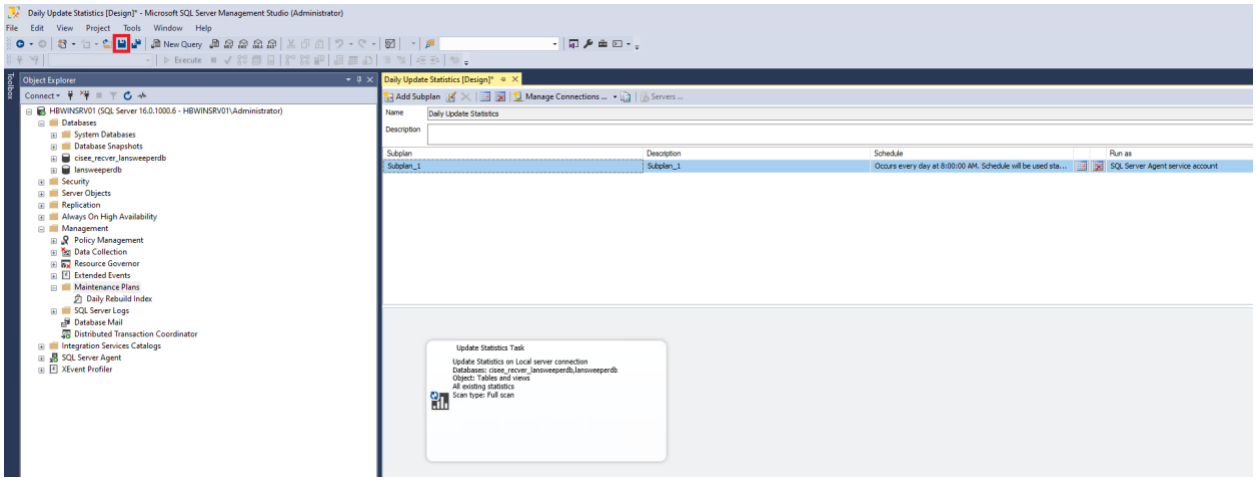


6. Click on the **Schedule** and set to **Daily** and chose a **Time of Day** to run the job. Then click **OK**

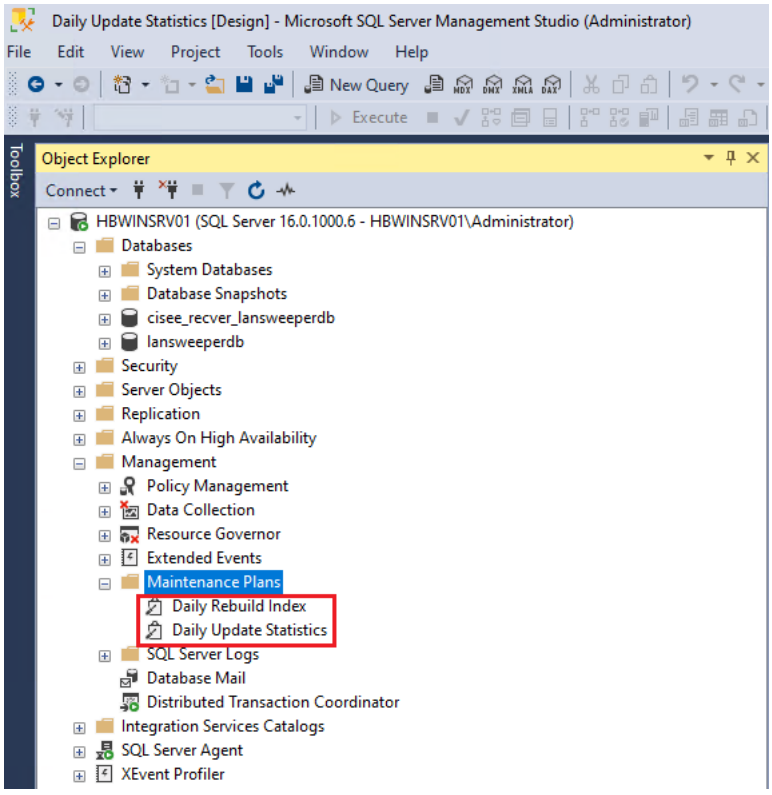
> **Guidance Note:** Choose a time that does not clash with Backups and other tasks being run on the SQL Server.



7. Click on **Save**



8. Refresh the **Maintenance Plans** and expand to **confirm the new job appears** in the list (i.e. the new job has been created)



Task B4: Assess other suggestions for SQL database health.

Informational Note: Your SQL DBA may want to consider the following additional jobs.

1. MSDB history clean-up
2. Database integrity checks
3. Job report file clean-up (i.e. a purge of report files created by the job)

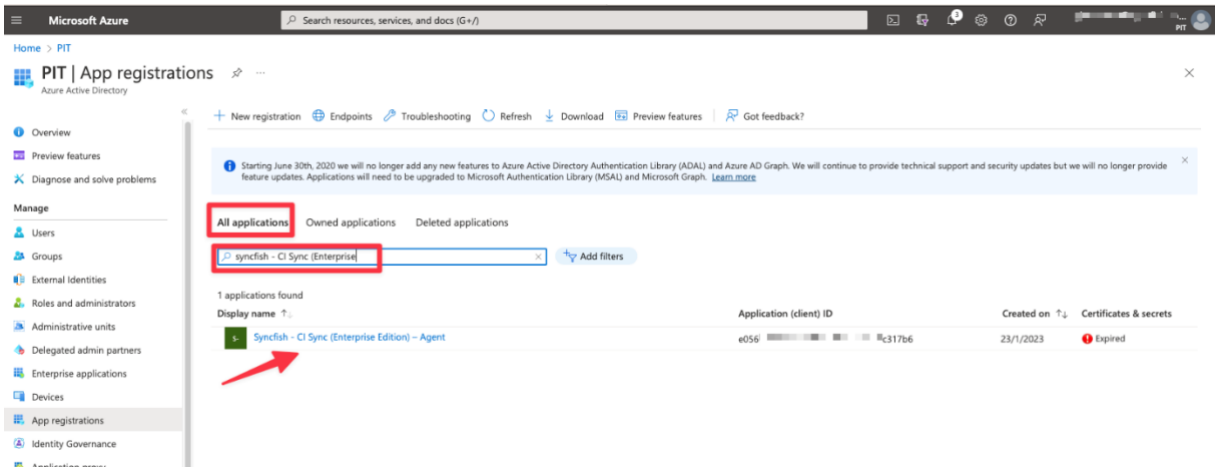
Appendix C – How to update the AAD Client Secret for the App Registration that represents the CI Sync Agent itself

Follow this section to regenerate the Client Secret for the AAD App Registration that represents the CI Sync (EE) Agent itself (as distinct from any client secrets/app registrations for individual source connections such as InTune or Azure). Once the Client Secret has been regenerated (via the Azure Portal), the instructions explain how to update the secret using the CI Sync (EE) Agent Config Utility User Interface.

- Task C1: Regenerate a new Client Secret for the CI Sync (EE) Agent App Registration using the Azure Portal.
- Task C2: Use the CI Sync (EE) Agent Config Utility to store the updated Client Secret.

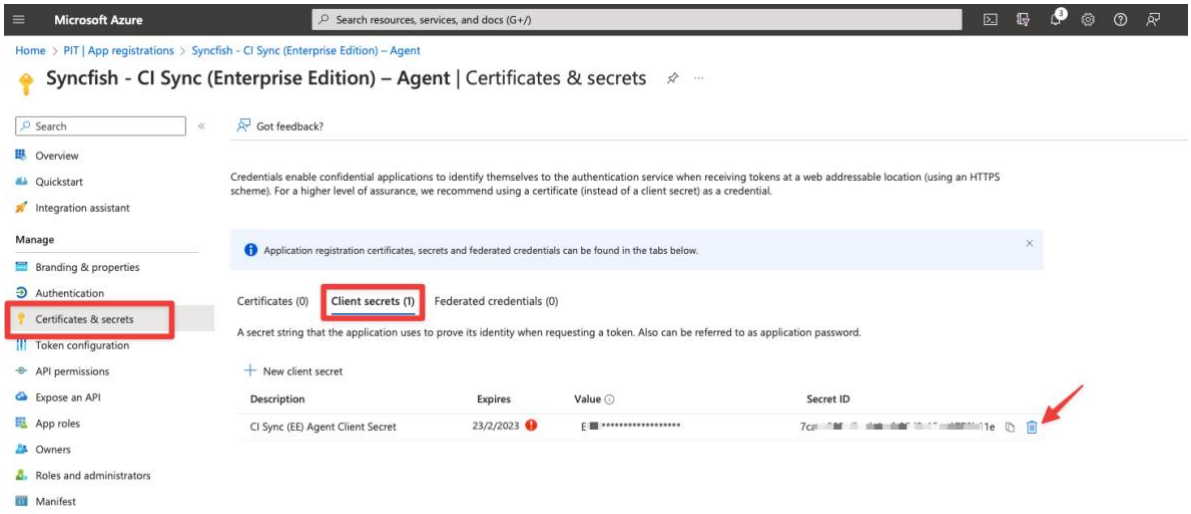
Task C1: Regenerate a new Client Secret for the CI Sync (EE) Agent App Registration using the Azure Portal.

1. In the **Azure Portal**, navigate to **Azure Active Directory** -> **App Registrations** and click **All Applications**
2. Locate the App Registration that you originally created for the CI Sync (EE) Agent (e.g. “Syncfish – CI Sync (Enterprise Edition) – Agent”)
3. Select the application name.

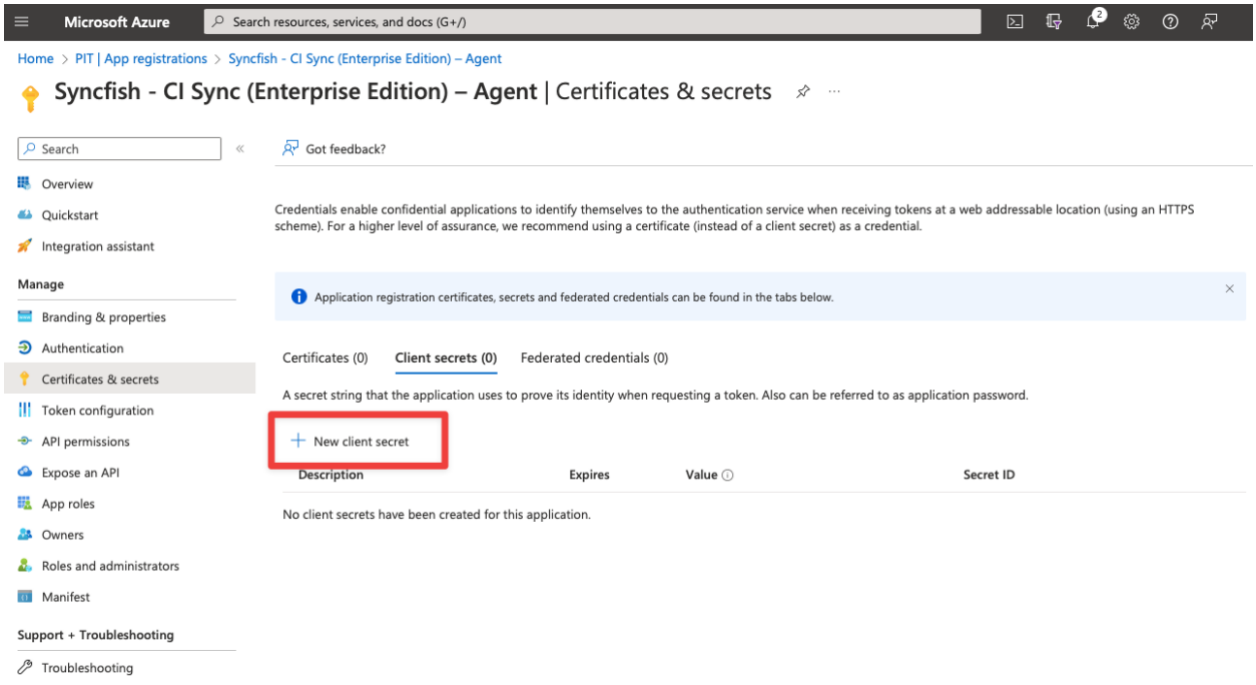


Click here to return to steps index

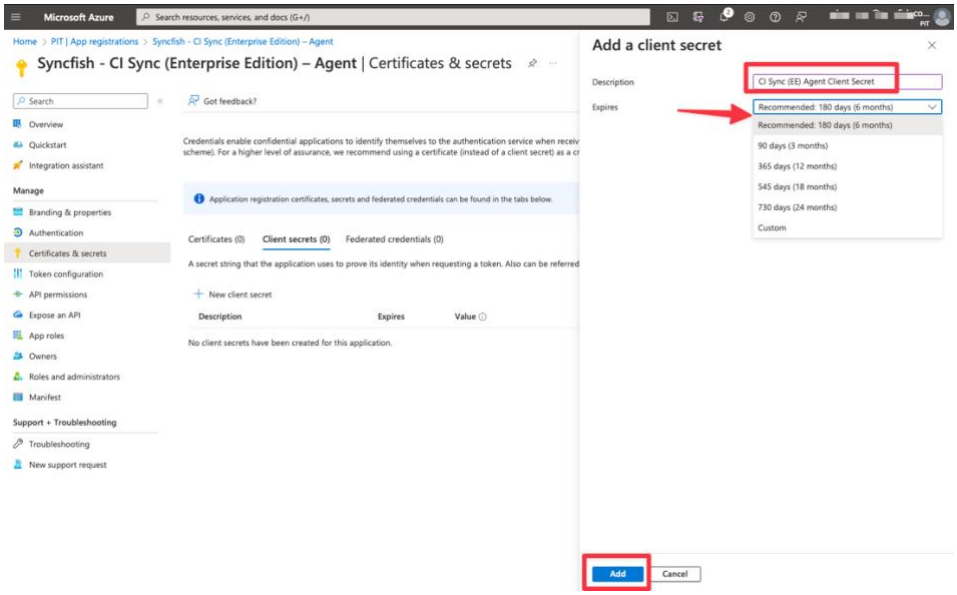
4. Select **Certificates & Secrets** and use the Trash icon to delete the existing/expired Secret.



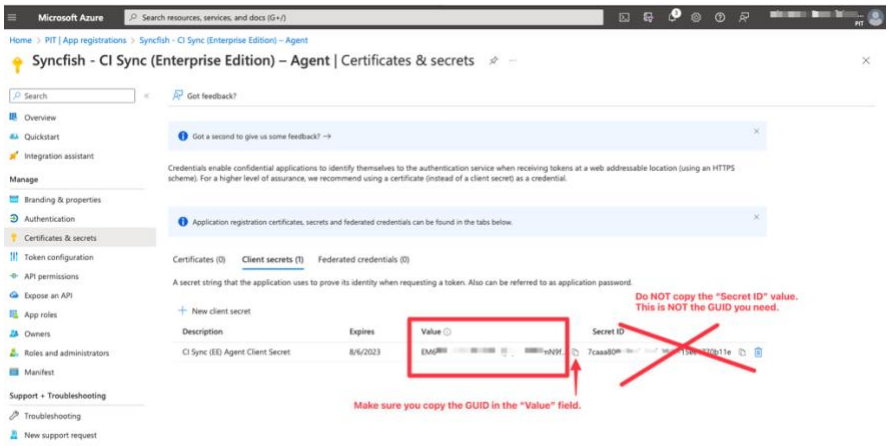
5. Click the "**New client secret**" button.



6. Enter a unique **Description** for the secret associated with the CI Sync (EE) Agent App Registration.
7. Then, select a suitable **Expires** duration based on your organisational policy. Finally click the **Add** button.



8. The form now displays the generated **secret value** (shown in the **Value** field). The **Value** is only available while you remain on this screen. You must make a copy of the **Value** GUID before leaving this form. We also recommend you securely store a copy of the secret internally in case you need to use it later after the CI Sync (EE) is installed.
   - Using the form, make a copy of the GUID in the **Value field.**



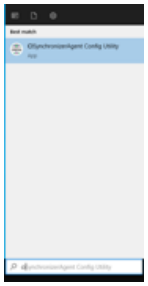
Data Capture Note: You will need the "**Value**" GUID when performing the subsequent steps to update the **CI Sync (EE) Agent (the Windows Service)** running on the relevant server.

Click here to return to steps index

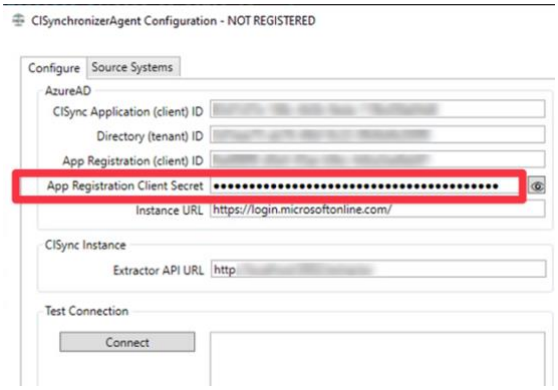
Task C2: Use the CI Sync (EE) Agent Config Utility to store the updated Client Secret.

Perform the following steps on the server where the CI Sync (EE) Agent was installed (referred to as the "CI Sync (EE) Agent Server").

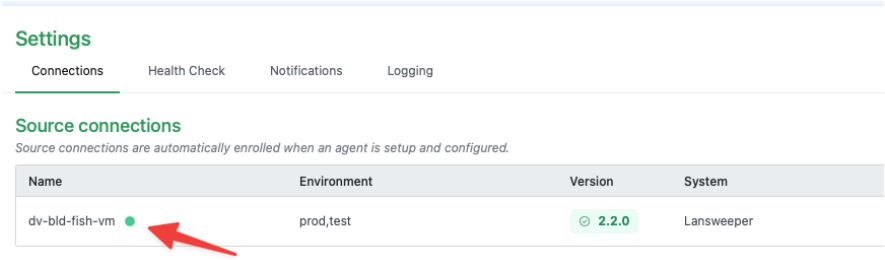
1. Locate and run the "**CISynchronizerAgent Config Utility**" program from the Windows Start Menu



2. Click the **Configure** tab and paste the new **Client Secret** (generated in the previous task) into the "App Registration Client Secret" field on the form.



3. In the Test Connection section, click "Connect".
4. Follow any remaining prompts to save and exit.
5. As a further verification you may also want to do the following:
   - Login to your CI Sync (EE) SaaS UI
   - Navigate the **Settings** Page and under **Connections**
   - Ensure you have a green light against the CI Sync (EE) Agent connection.



Click here to return to steps index

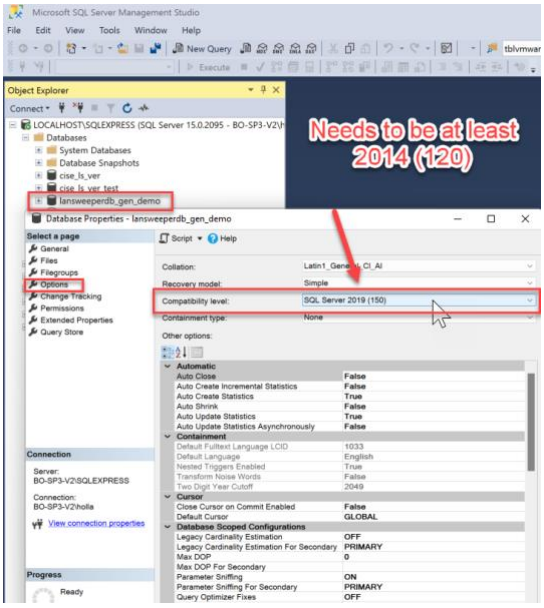
Appendix D – How to add Lansweeper as a Source Connection to the CI Sync (EE) Agent

- Task D1: Review the SQL Server Edition and Version being used to host the Lansweeper SQL DB.
- Task D2: Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection to Lansweeper.
- Task D3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EE) Agent status and set any remaining connection parameters.

Task D1: Review the SQL Server Edition and Version being used to host the Lansweeper SQL DB.

You need to review the SQL Server Version and SQL Server Edition to ensure it is supported by (and optimal) for CI Sync (EE). Check the following:

- CI Sync (EE) **does not support the LocalDB version of SQL** (i.e. the free SQL which is sometimes used as the default during the installation of certain products (e.g. Lansweeper). If your Lansweeper is using the LocalDB version of SQL you will need to upgrade to SQL Express Edition or preferably SQL Standard Edition (see next point).
- Syncfish **highly recommend using Microsoft SQL Standard Edition** (or SQL Enterprise Edition though that edition is typically overkill/not required). SQL Standard (or above) supports SQL Maintenance Plans and other housekeeping jobs.
- Check you are using a recent **version** of SQL Server (**SQL Version 2014 or later**).
- Validate the **database compatibility level** of the Lansweeperdb is set to **SQL Server 2014 (120)** or later.



Click here to return to steps index

Task D2: Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection to Lansweeper.

Informational Note

This section includes references to a database referred to as "RecVer". The RecVer database is a small three column checksum style of database used by CI Sync (EE) to deliver the delta record sync functionality (i.e. where CI Sync (EE) only sync's those records which have changed since the previous sync job. Below are some key points about the RecVer database:

1. The RecVer database is currently only supported via Microsoft SQL Server (i.e. RecVer must be a SQL database). In the new future RecVer will also be supported on MongoDB.
2. The RecVer database **must** be hosted on the same SQL Server as your Lansweeper source system database. You cannot host the RecVer database on a separate SQL server.
3. You will need to create multiple RecVer databases if you are setting up multiple source systems via the CI Sync (EE) Agent (i.e. each Source Connection that supports the delta sync functionality requires its own unique RecVer database.

The follow pages explain how to use the CI Sync (EE) Agent Config Utility to setup a Lansweeper source system connection. You will see references in the screen shots and text which explains how to facilitate the creation and use of your RecVer database.

Guidance Notes

When performing the tasks in this section (to setup a Source System Connection for an Lansweeper using the **CI Sync (EE) Agent Config Utility**) you will need to know the following:

1. The path to the SQL server that will host the "RecVer" database.

2. Which authentication type the CI Sync (EE) Agent will use for readonly access to the Lansweeper database and read/write access to the RecVer database. The **Config Utility** supports the following authentication types for accessing the SQL server:
   - **Integrated Security** (which uses the Windows User Account you created for the CI Sync (EE) Agent Windows Service).
   - **SQL Native Login** (which uses a SQL username and password you enter into the **Config Utility** when creating the Source System Connection).
   - *Note: In either case (Integrated Security of SQL Native Login), the account/login requires minimum (least privileged) access to the databases. Permissions are set either by your DBA (through a manual process) or the Config Utility (through the automated process) as explained below.*

Continue to the next page for further Guidance Notes.

Guidance Notes (continued from prior page)

When performing the tasks in this section (to setup a Source System Connection for an Lansweeper using the **CI Sync (EE) Agent Config Utility**) you will also need to know the following:

3. Which approach you will use to create the CI Sync "RecVer" database. Your options are as follows:
   - Option 1 - You will use the **Config Utility** to automatically create the RecVer database and grant the CI Sync (EE) Agent access to the Lansweeper database while you are running the **Config Utility** to setup the Source System Connection to the Lansweeper SQL database.
   - Option 2 – You will engage your SQL Database Administrator (DBA) to manually create the RecVer database (and grant the CI Sync (EE) Agent access to the Lansweeper database) on your behalf (i.e. in advance) if you don't have permissions to the SQL server and therefore you cannot use the **Config Utility** to automatically create the RecVer database.

If you are clear on the above items, it will improve your use of the **Config Utility** to select the correct options and provide appropriate credentials when setting up Lansweeper as a Source System Connection for the CI Sync (EE) Agent. See also Appendix A which contains useful diagrams about SQL authentication for the CI Sync (EE) Agent.

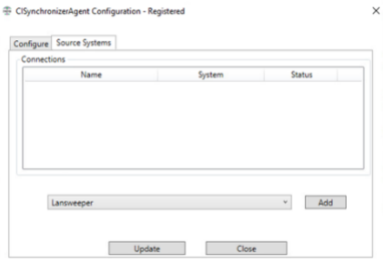
As mentioned in the Guidance Notes on the previous page, there are two options to facilitate the creation of the RecVer database and assigning permissions to it for the CI Sync (Agent).

Option	Title	Additional Details
1	Automatically using the CI Sync (EE) Agent Config Utility	• This option requires you to have the sysadmin (sa) role in the SQL Server that will host the RecVer database. • If you have sa credentials to the SQL Server that will host the RecVer database you can use the **CI Sync (EE) Agent Config Utility** to create the CI Sync (EE) RecVer database**.** • For this option proceed to the very next page under the heading Task D2 (Option 1): Steps to add Lansweeper as a Source Connection to the CI Sync (EE) Agent using the Config Utility to automatically create the RecVer database..
2	Manually via a SQL Database Administrator (DBA)	• If the person executing the CI Sync (EE) Agent Config utility **does not** have the sysadmin (sa) role in the SQL Server, you can have your SQL DBA perform a manual setup of the RecVer database. • For this option you should contact your DBA now and allocate them the manual setup tasks described in Task D2 (Option 2): Steps to add Lansweeper as a Source Connection to the CI Sync (EE) Agent using your SQL Database Administrator (DBA) to manually create the RecVer database in advance. • Once the DBA has completed their allocated tasks you will continue to use the **CI Sync (EE) Agent Config Utility** to connect to the RecVer database the DBA has created for you.

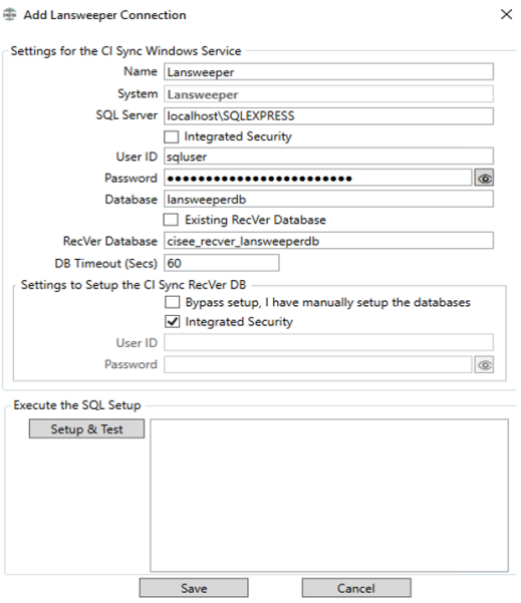
See the instructions in Task D2 (Option 1) and Task D2 (Option 2) on the following pages for detailed steps for each of these two options.

Task D2 (Option 1): Steps to add Lansweeper as a Source Connection to the CI Sync (EE) Agent using the Config Utility to automatically create the RecVer database.

1. Within the CI Sync Agent Configuration Utility, navigate to the **Source Systems** tab, select **Lansweeper** and click the **Add** button.



2. Enter details on the "**Add Lansweeper Connection**" form.

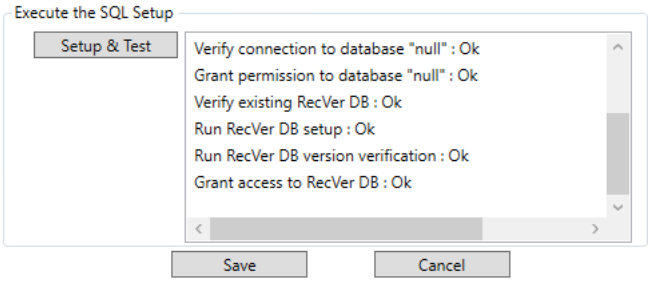


   (a) In the section titled "**Settings for the CI Sync Windows Service**"
   This section defines the settings for the CI Sync (EE) Agent to use as it runs in the background (i.e. after you have exited the Config Utility).
   - **Name**: Enter a name that describes the Lansweeper instance this Source System Connection represents.
     If you have only one Lansweeper instance "Lansweeper" will suffice, if you have multiple enter a name that identifies the specific Lansweeper instance.
   - **SQL Server**: Enter the SQL Server instance path in the format: server\instance.
   - **Integrated Security**: Check this checkbox if you want the CI Sync (EE) Agent to authenticate to the SQL Server using the Windows User Account you setup for the CI Sync (EE) Agent Windows Service. Otherwise, uncheck this checkbox and enter the **User ID** and **Password** for a **Native SQL Login** provided to you by the DBA
     (Note: Syncfish do not recommend the SQL Login have sysadmin rights).
   - **Database**: The name of the Lansweeper database. This will default to "lansweeperdb", only change this value if your Lansweeper database has a non-standard name.
   - **Existing RecVer Database**: Do NOT check this checkbox (because the **Config Utility** will be creating the RecVer database for you).
   - **DB Timeout (Secs)**: Sets both the connection and statement execution timeouts. Syncfish recommend 60 (seconds).

   (b) In the section titled "**Settings to Setup the CI Sync RecVer DB**"
   *This section defines how the **Config Utility** will connect to SQL Server to perform the automated SQL setup and configuration steps. The settings do not impact the ongoing running of the CI Sync (EE) Agent*.
   - **Bypass setup, I have manually setup the databases:** Do NOT check this checkbox (because the **Config Utility** will be creating the RecVer database for you).
   - **Integrated Security:** By default, the **Config Utility** will connect to SQL Server to perform the automated SQL setup and configuration steps using Integrated Security (i.e. using your own credentials as the person running the **Config Utility**). Uncheck this checkbox if you want the automated SQL setup and configuration steps to be performed with a native SQL login. Then enter the **User ID** and **Password** of the native SQL login. *Note: the automated process requires the native login to have the SQL sysadmin role.*

Click here to return to steps index

3. In the section titled "**Execute the SQL Setup**", click the **Setup & Test** button. The tests should complete without error.



Informational Note

The **Config Utility** performs the following SQL setup and configuration steps:

1. It creates a SQL Login for the Windows User Account you setup for the CI Sync (EE) Agent Windows Service if you selected "Integrated Security" in the section "**Settings for the CI Sync Windows Service**", the **Config Utility**. Or, if instead you are using Native SQL Login (instead of "Integrated Security") then **Config Utility** uses the Native SQL Login details you have supplied and performs the following two steps to grant the Login access to the relevant databases with the permissions noted below.
2. It **creates a SQL user in the Lansweeper database** for the Login used by the CI Sync (EE) Agent **and grants the user the db_datareader role** to the Lansweeper database.
   *Note: Depending on whether you selected "Integrated Security" or not, the **SQL user** is either (a) the Windows User/Service Account (created as a SQL Login in point 1 above) or (b) the native SQL login account.*
3. Creates the RecVer database, creates a **SQL user** in the RecVer database for the Login used by the CI Sync (EE) Agent and grants the **SQL user** the agent_role role.
   *Note: Depending on whether you selected "Integrated Security" or not, the **SQL user** is either (a) the Windows User/Service Account (created as a SQL Login in point 1 above) or (b) the native SQL login account.*

Note: If you are using a Native SQL Login (not Integrated Security) with higher privileges than noted above (e.g. if the account had SQL sysadmin) then the Config Utility will not be able to set the SQL roles noted above. Instead, the Config Utility will not touch the permissions and instead will leave the existing/higher privileges as is. This is not required and not recommended by Syncfish.

Important Note: The above process grants the SQL User credential (i.e. the Windows Service Account or the SQL Login) db_datareader at the database level (i.e. grants the credential access to all tables in the Lansweeper database schema).

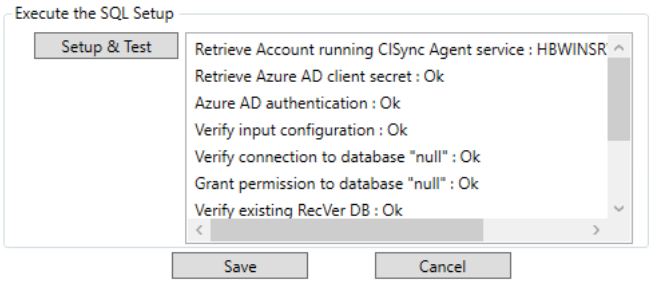
This approach grants the CI Sync (EE) Agent access to Lansweeper tables that are not needed or in-scope for CI Sync (EE). A good example is the tblADusers table in Lansweeper. The default CI Sync (EE) configuration rules do require the CI Sync (EE) Agent to access tblADusers which may contain firstname, surname, email address etc values about your users (subject to whether you have configured Lansweeper to read from your Active Directory or not).

Syncfish recommends you review the Lansweeper system documentation (and system itself) to understand the permissions these roles provide to your CI Sync (EE) Agent.

Syncfish provide further details on this topic in the document titled "**CI Sync (EE) - Overview of Source and Destination Fine Grain Permission Option for Personal Data**". The document also includes non-authoritative guidance on how to assess and potentially apply fine-grain permissions to further restrict CI Sync (EE)'s access within the Lansweeper database if your organization has concerns about Personal Data.

Click here to return to steps index

4. After a successful test has been completed, click the **Save** button.



5. Finally, click **Yes** to register the CI Sync (EE) Agent with your customer specific CI Sync (EE) SaaS instance.

6. You can now skip the next couple of pages (which describe a SQL DBA doing manual setup of the RecVer database) and proceed to Task D3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EEE) Agent status and set any remaining connection parameters.

Task D2 (Option 2): Steps to add Lansweeper as a Source Connection to the CI Sync (EE) Agent using your SQL Database Administrator (DBA) to manually create the RecVer database in advance.

1. Ask you SQL DBA to perform the tasks listed in the table below.

#	DBA Tasks (in SQL Server)	Additional Notes
1	Either **register a Login** (which represents the Windows User Account used by the CI Sync (EE) Agent Windows Service) Or **create a SQL Login** (for the CI Sync (EE) Agent to use).	If you are using **Windows "Integrated Security"** between the CI Sync (EE) Agent and the SQL Server, **then the CI Sync (EE) Agent Windows Service account needs to be registered within SQL Server as a Login**. *Note: The CI Sync (EE) Agent Windows Service account is the one you created during Task 4a: Create a Windows Service Account for the CI Sync (EE) Agent to use.* **Alternatively**, if you are using **Native SQL authentication** between the CI Sync (EE) Agent and the SQL Server, **then a Login will need to be created**. In this case your DBA will need to provide you with the SQL User ID and password details so you can enter them into the Config Utility UI as explained further below.
2	**Map the Login as a user** in the **Lansweeper** database and **grant the user** the **db_datareader** role.	Map the user and grant the role to the "lansweeperdb" (this is the default name for the Lansweeper database).

Important Note: The above process grants the SQL User credential (i.e. the Windows Service Account or the SQL Login) db_datareader at the database level (i.e. grants the credential access to all tables in the Lansweeper database schema).

This approach grants the CI Sync (EE) Agent access to Lansweeper tables that are not needed or in-scope for CI Sync (EE). A good example is the tblADusers table in Lansweeper. The default CI Sync (EE) configuration rules do require the CI Sync (EE) Agent to access tblADusers which may contain firstname, surname, email address etc values about your users (subject to whether you have configured Lansweeper to read from your Active Directory or not).

Syncfish recommend you review the Lansweeper system documentation (and system itself) to understand the permissions these roles provide to your CI Sync (EE) Agent.

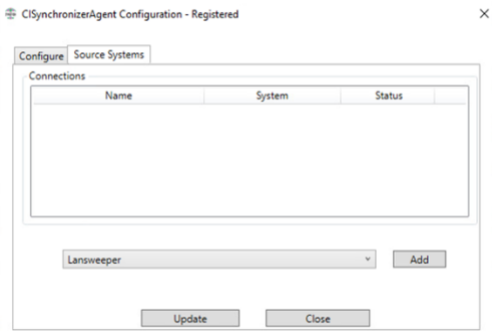
Syncfish provide further details on this topic in the document titled "**CI Sync (EE) - Overview of Source and Destination Fine Grain Permission Option for Personal Data**". The document also includes non-authoritative guidance on how to assess and potentially apply fine-grain permissions to further restrict CI Sync (EE)'s access within the Lansweeper database, in particular if your organization has concerns about Personal Data.

Proceed to the next page to complete the remaining steps.

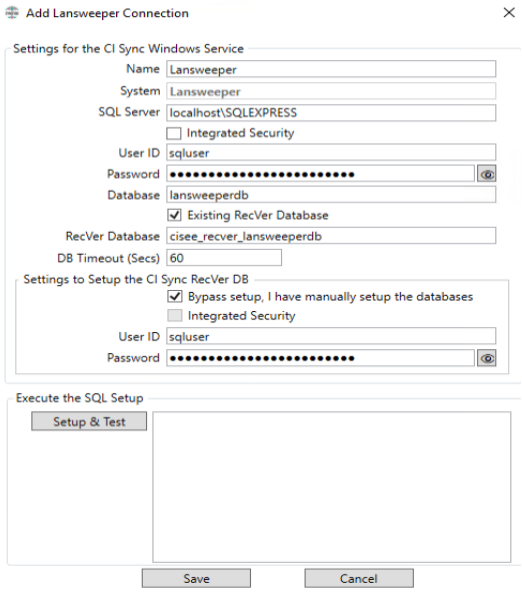
#	DBA Tasks (in SQL Server)	Additional Notes
3	**Execute the SQL Script** provided by Syncfish located here. The script creates the Syncfish RecVer database. The script is: Cisee-recver-create-script.sql	Notes for the DBA when running the script: • The script has a placeholder called $(database_name) as the replacement for the RecVer database name. • For a Lansweeper Source System Syncfish recommend "**cisee_recver_lansweeperdb**" as the RecVer database name. • To run the setup script using the sqlcmd utility, the database_name parameter needs to be passed in using the -v switch: *> sqlcmd -I -S localhost -v database_name=" **cisee_recver_lansweeperdb** " -i "cisee-recver-create-script.sql"* • If running the script in SSMS, the value $(database_name) (including the $ and brackets) needs to be replaced with the name for the recver database.
4	**Map the Login as a user** in the **RecVer** database and **grant the user** the **role_agent** role.	Map the user and grant the role to the RecVer database created by the SQL script above.

Click here to return to steps index

2. Within the CI Sync Agent Configuration Utility, navigate to the **Source Systems** tab, select **Lansweeper** and click the **Add** button.



3. Enter details on the "**Add Lansweeper Connection**" form.



   (a) In the section titled "**Settings for the CI Sync Windows Service**"
   This section defines the settings for the CI Sync (EE) Agent to use as it runs in the background (i.e. after you have exited the Config Utility).

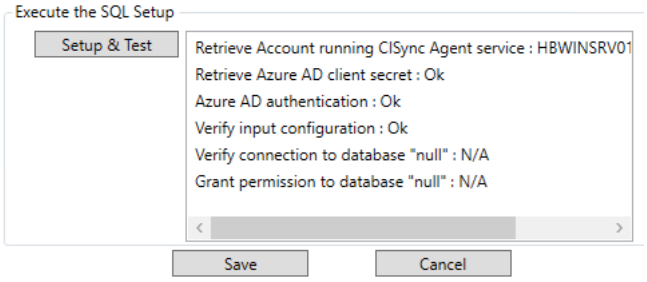
   - **Name**: Enter a name that describes the Lansweeper instance this Source System Connection represents.
     If you have only one Lansweeper instance "Lansweeper" will suffice, if you have multiple enter a name that identifies the particular Lansweeper instance.
   - **SQL Server**: Enter the SQL Server instance path in the format: server\instance.
   - **Integrated Security**: Check this checkbox if you want the CI Sync (EE) Agent to authenticate to the SQL Server using the Windows User Account you setup for the CI Sync (EE) Agent Windows Service. Otherwise, uncheck this checkbox and enter the **User ID** and **Password** for a **Native SQL Login** provided to you by the DBA.
   - **Database**: The name of the Lansweeper database. This will default to "lansweeperdb", only change this value if your Lansweeper database has a non-standard name.
   - **Existing RecVer Database**: Check this checkbox (because your SQL DBA has already manually created the RecVer database for you).
   - **DB Timeout (Secs)**: Sets both the connection and statement execution timeouts. Syncfish recommend 60 (seconds).

   (b) In the section titled "**Settings to Setup the CI Sync RecVer DB**"
   *This section defines how the **Config Utility** will connect to SQL Server to perform the automated SQL setup and configuration steps. Because your SQL DBA has already manually created the RecVer database you use this section to confirm this is the case.*

   - **Bypass setup, I have manually setup the databases:** Check this checkbox (because your SQL DBA has already manually created the RecVer database and set the relevant permissions for you).

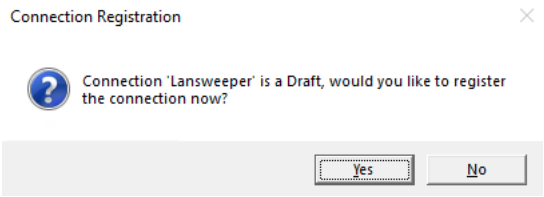
Click here to return to steps index

4. In the section titled "**Execute the SQL Setup**", click the **Setup & Test** button. The tests should complete without error.

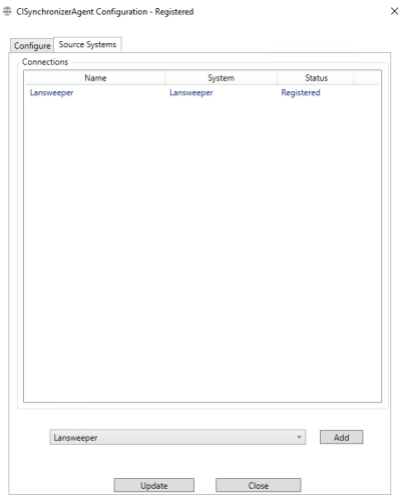


5. After a successful test has been completed, click the **Save** button.

6. Finally, click **Yes** to register the CI Sync (EE) Agent with your customer specific CI Sync (EE) SaaS instance.



7. The text in the Config UI will now show "Registered" as part the heading.

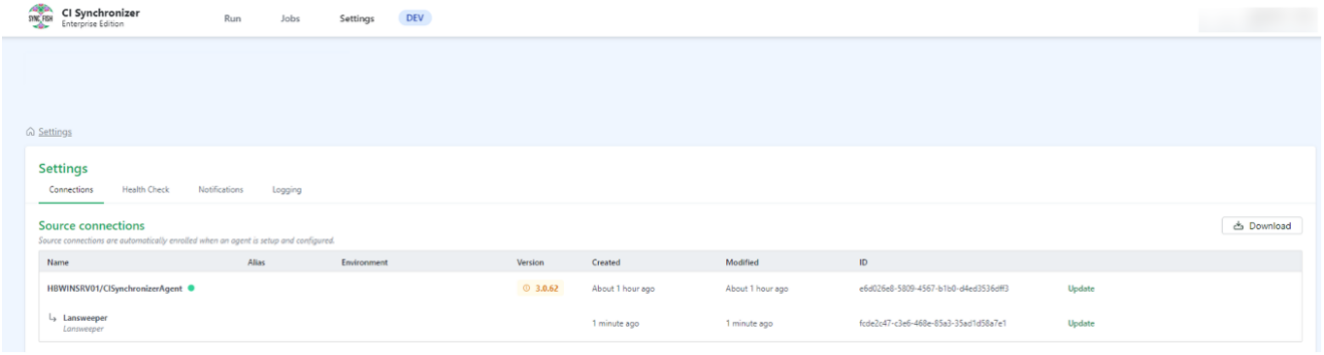


8. You can now proceed to [Task D3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EEE) Agent status and set any remaining connection parameters.]

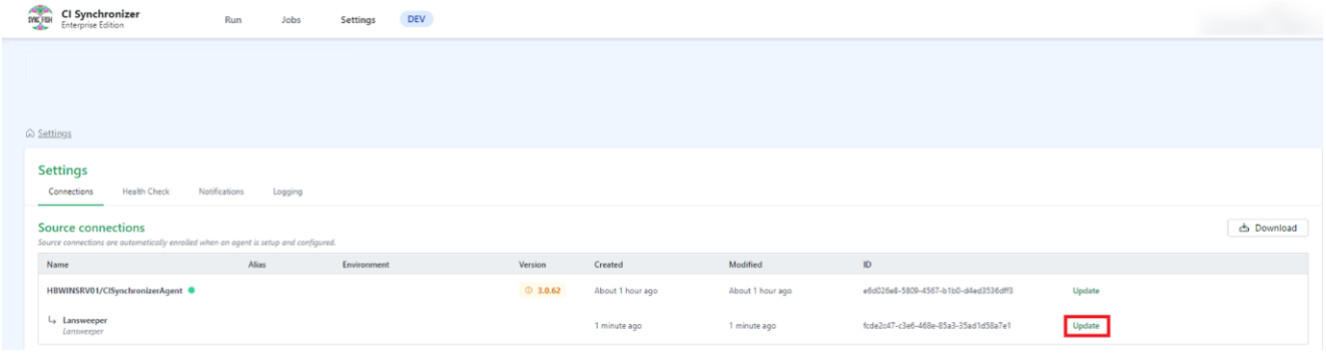
Click here to return to steps index

Task D3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EE) Agent status and set any remaining connection parameters.

1. Login to your CI Sync (EE) SaaS instance at https://***YourCo***.syncfish.app
2. In the CI Sync UI, navigate to **Settings > Connections**. The new Source System Connection should appear in the Source connections list

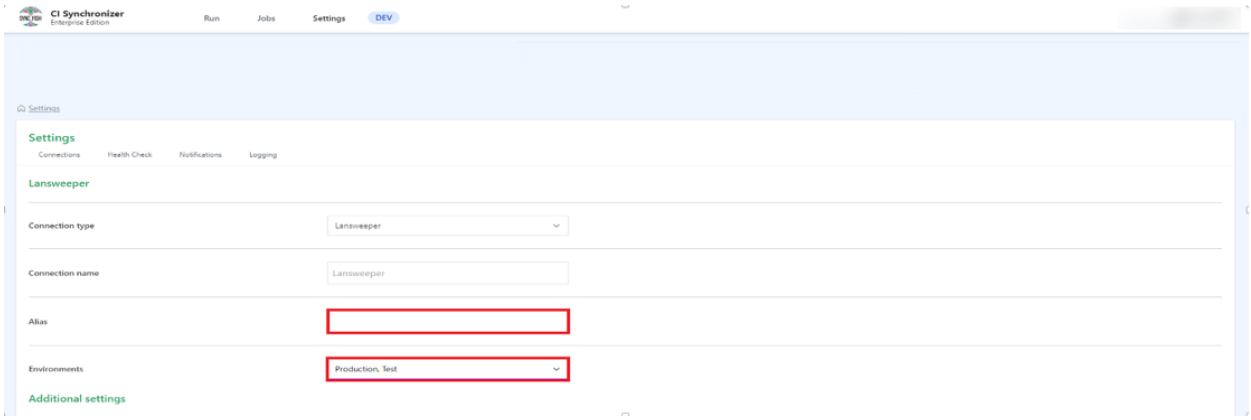


3. Click the **Update** hyperlink



Click here to return to steps index

4. Use the settings form to as follows:
   - Enter an **Alias** (optional)
   - Select the **Environment**/s the new source connection can be used for. In most cases a Source System Connection is used for both Test and Production sync jobs (as distinct from the Destination Connections which can only be either Test or Production).
   - Check the **consent** checkbox and Click the **Save connection** button.



You have now completed all tasks to add Lansweeper as a Source Connection in the CI Sync (EE) Agent. You should now proceed back to Step 6 - Configure your ServiceNow to be ready for CI Sync (EE) (and subsequent steps).

Appendix E – How to add Intune as a Source Connection to the CI Sync (EE) Agent

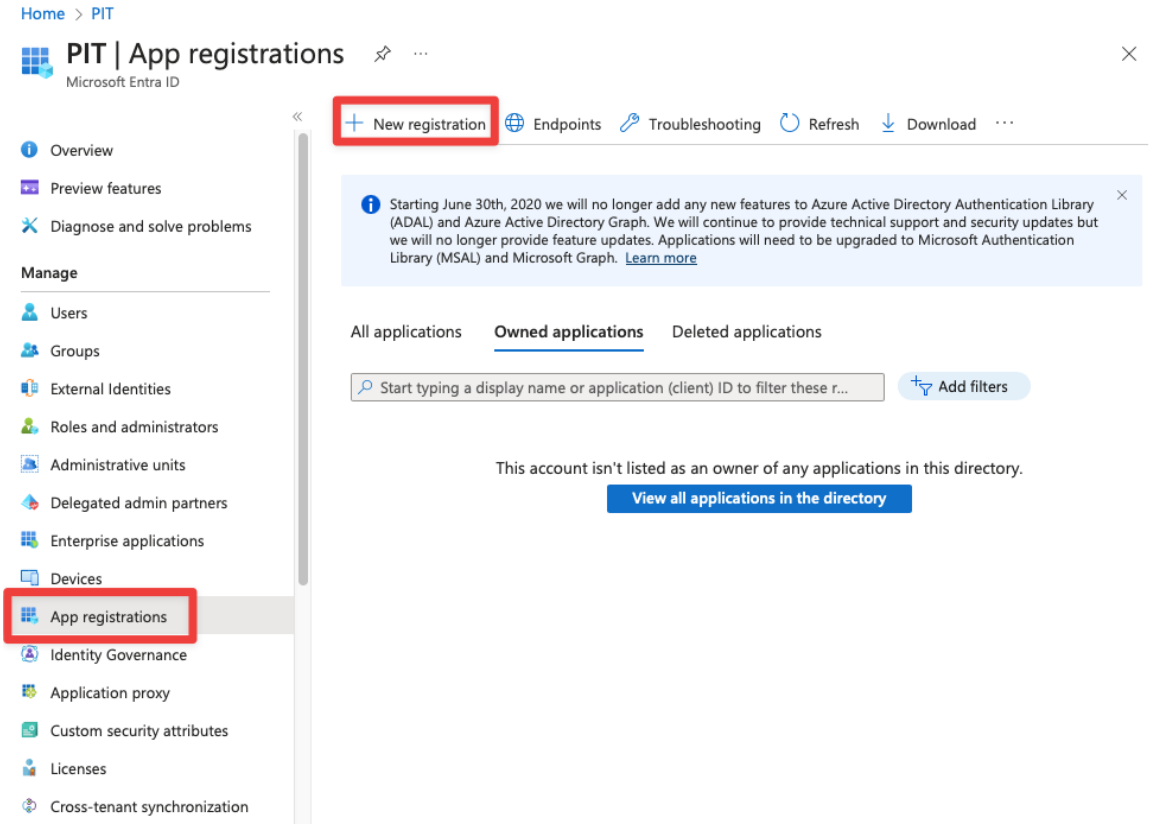
- Task E1: Create a new App Registration to grant permissions for the CI Sync (EE) Agent to access InTune.
- Task E2: Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection to InTune.
- Task E3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EEE) Agent status and set any remaining connection parameters.

Task E1: Create a new App Registration to grant permissions for the CI Sync (EE) Agent to access InTune.

Context Notes:

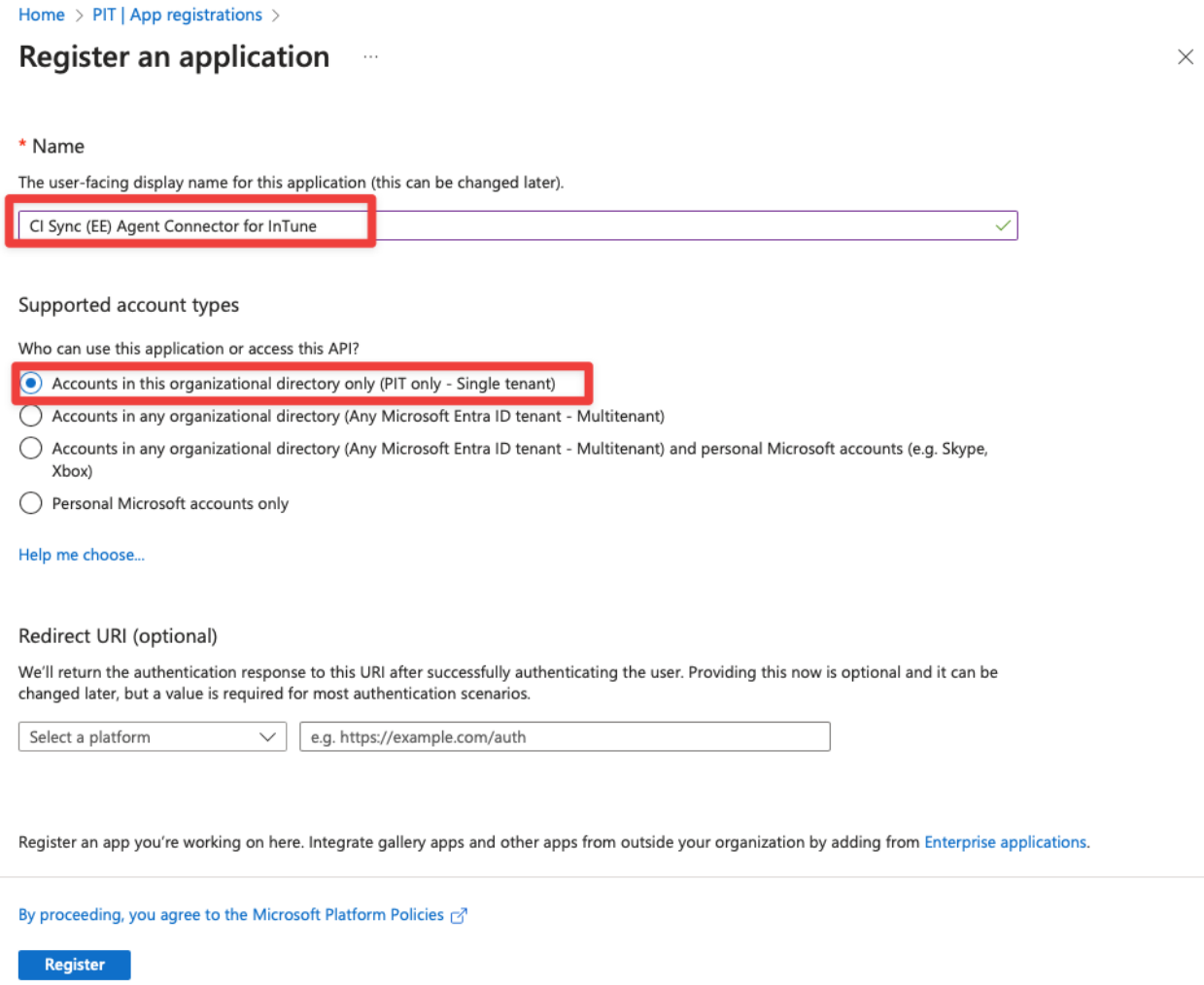
- In Azure ADD (AAD) an App Registration is used to define a Service Principal for the purpose of authenticating a source application to a destination system/application.
- The **App Registration created in this section relates to the InTune Source Connection** created within the CI Sync (EE) Agent. This InTune specific App Registration is different to the App Registration for the CI Sync (EE) Agent itself that was create in Step 3.

1. In the **Azure Portal**, navigate to **Azure Active Directory** -> **App Registrations** and click **New Registration**



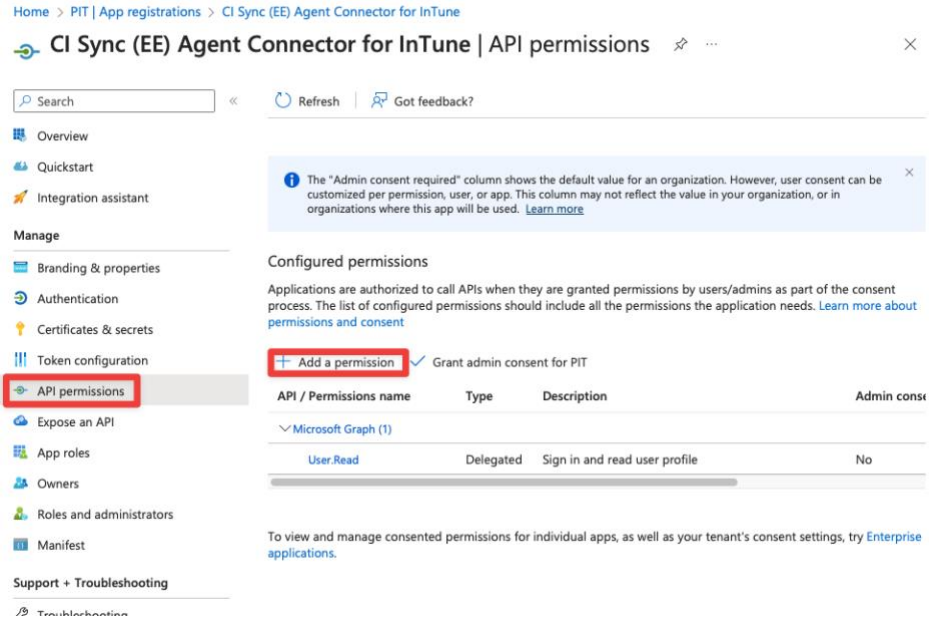
Click here to return to steps index

2. On the **Register an application** form complete as follows:
   - Enter the **Name** (Note: Syncfish recommend using "CI Sync (EE) Agent Connector for InTune")
   - Under **Supported account types** select "**Accounts in this organizational directory only (*{Your Domain/Tenant Name}* only - Single tenant)**"
   - Click **Register**

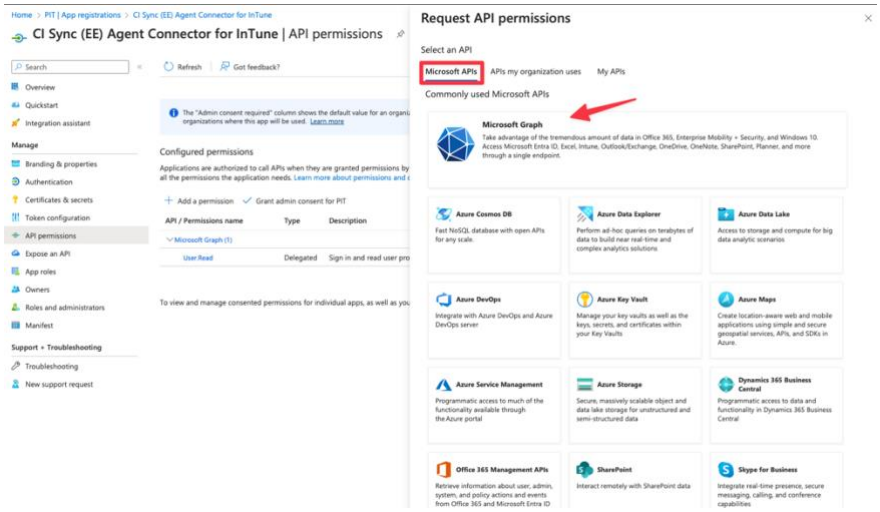


Click here to return to steps index

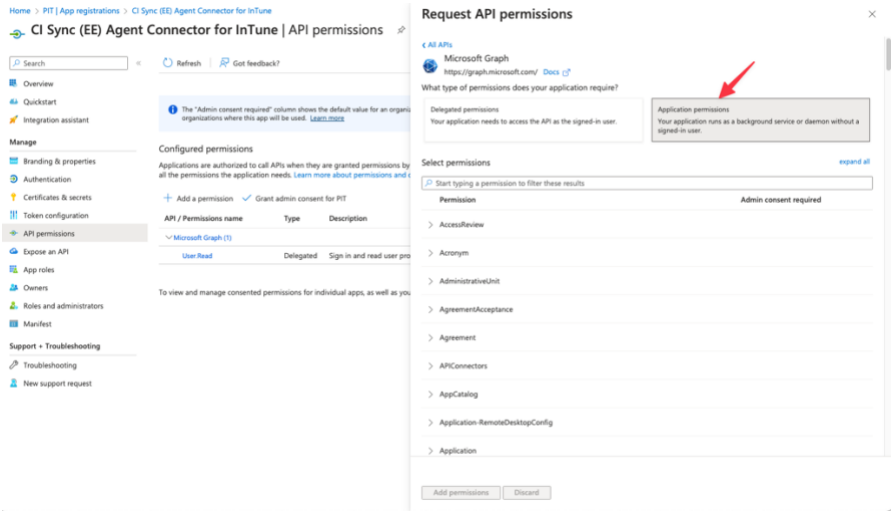
3. Navigate to **API permissions** and click **Add Permission**



4. Click on **Microsoft Graph**



5. Click on **Application permissions**

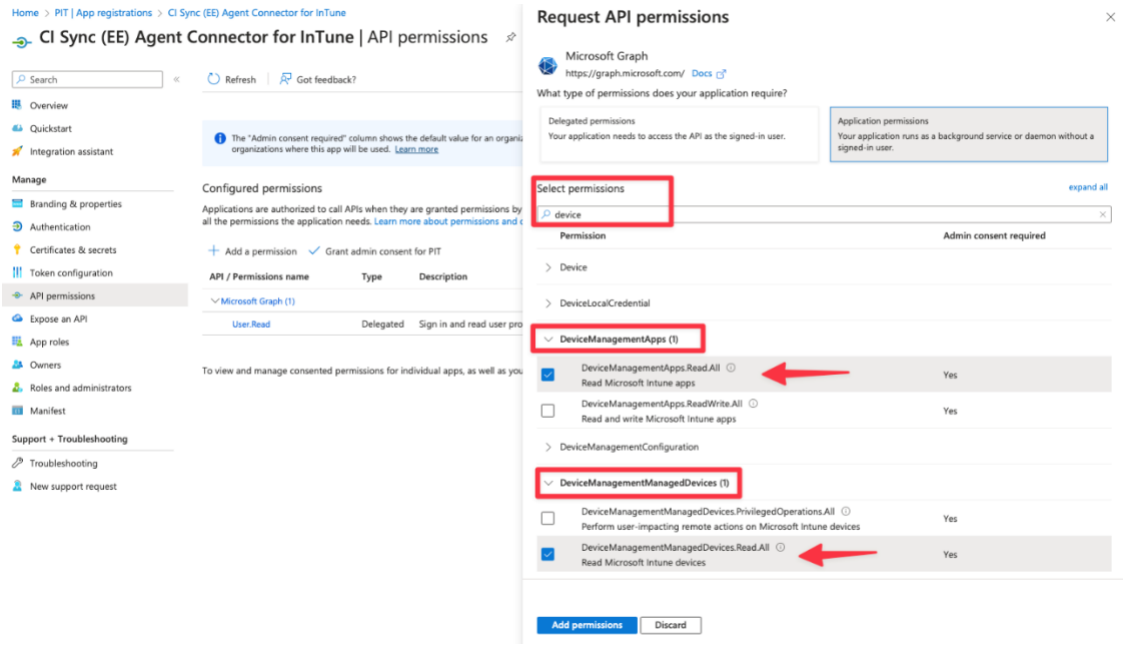


6. In the **Select permissions** filter, enter **"**device" and then do the following:

- First, find and expand **DeviceManagementApps**, then **tick/select DeviceManagementApps.ReadAll Read Microsoft InTune apps**

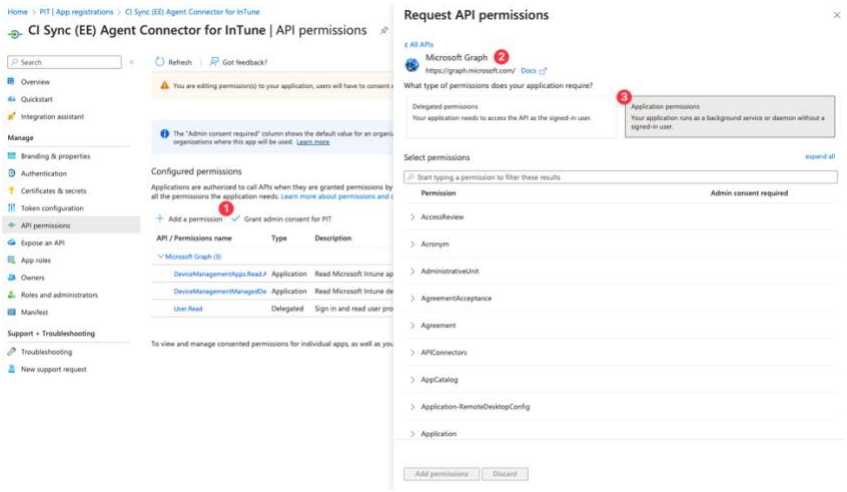
- Next, find and expand **DeviceManagementManagedDevices**, then **tick/select DeviceManagementManagedDevices.ReadAll Read Microsoft InTune devices**

- Then, click the **Add permission** button



Click here to return to steps index

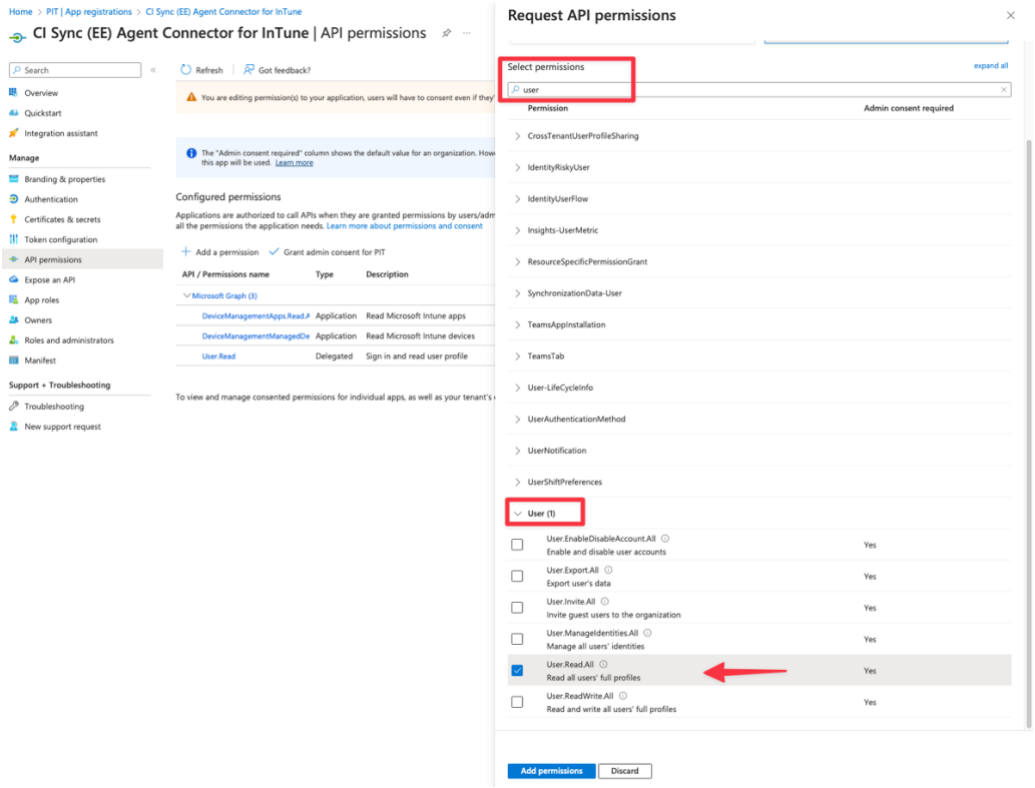
7. For a second time, click on **Application permissions,** select **Microsoft Graph** and then click Application Permissions



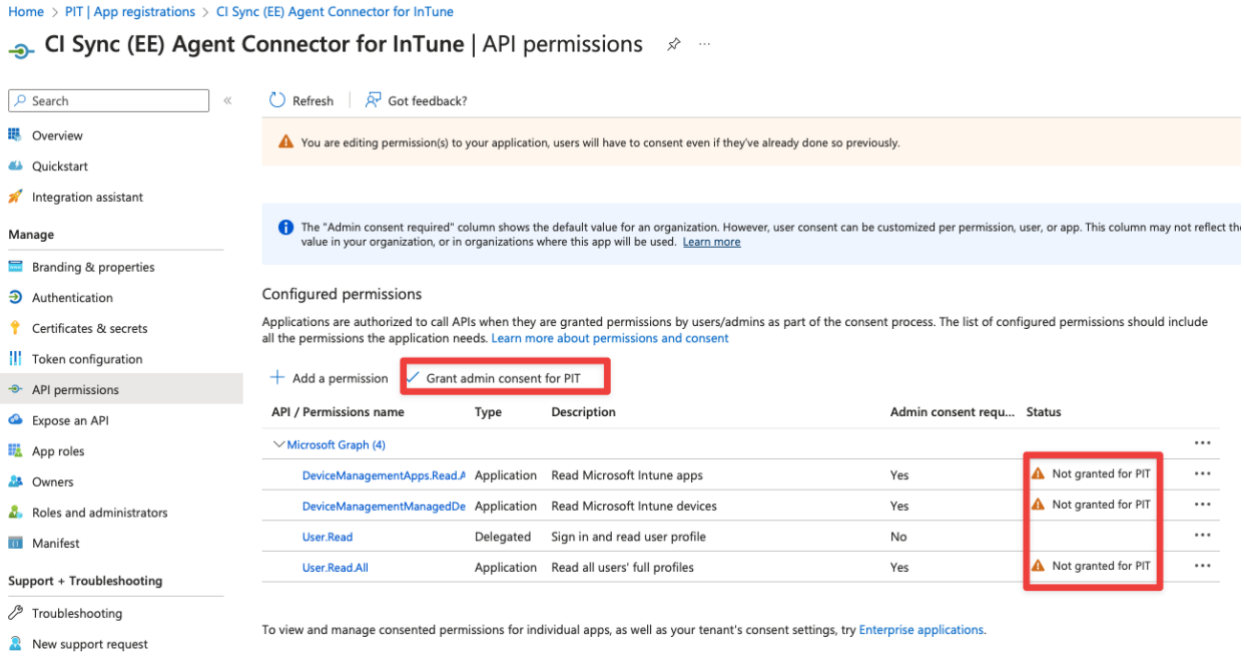
8. In the **Select permissions** filter, this time enter **"**user" and then do the following:

   - First, expand **User**, then **tick/select**
     User.ReadAll
     Read all users's full profiles

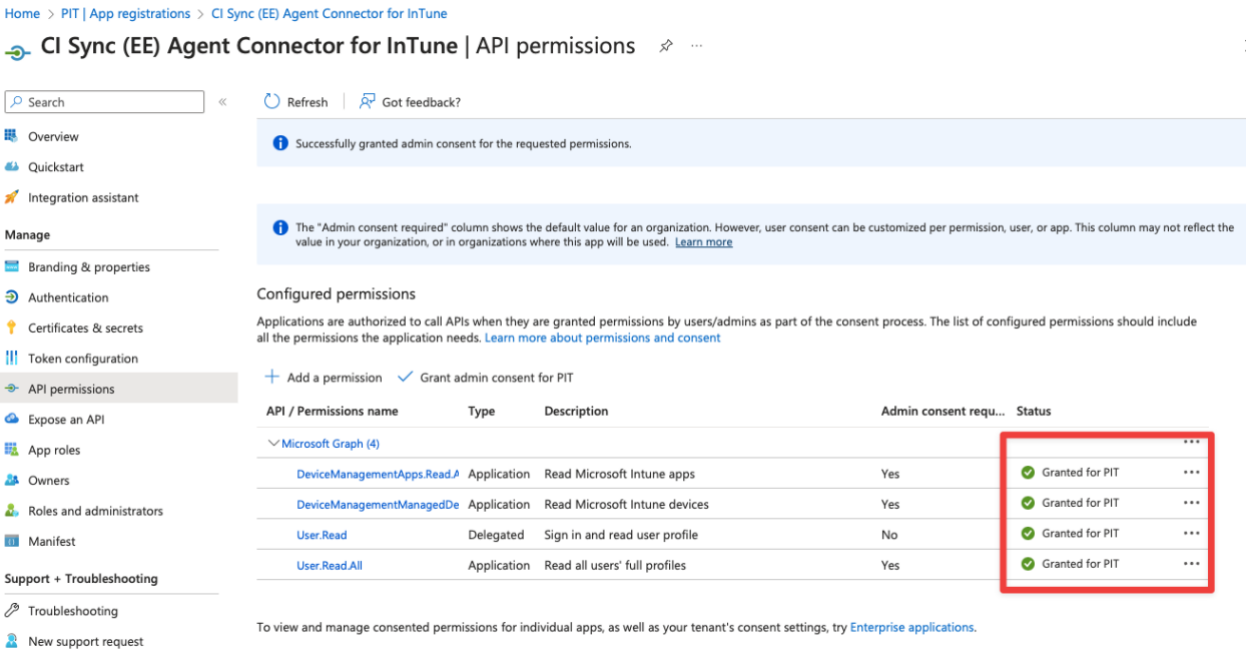
   - Then, click the **Add permission** button



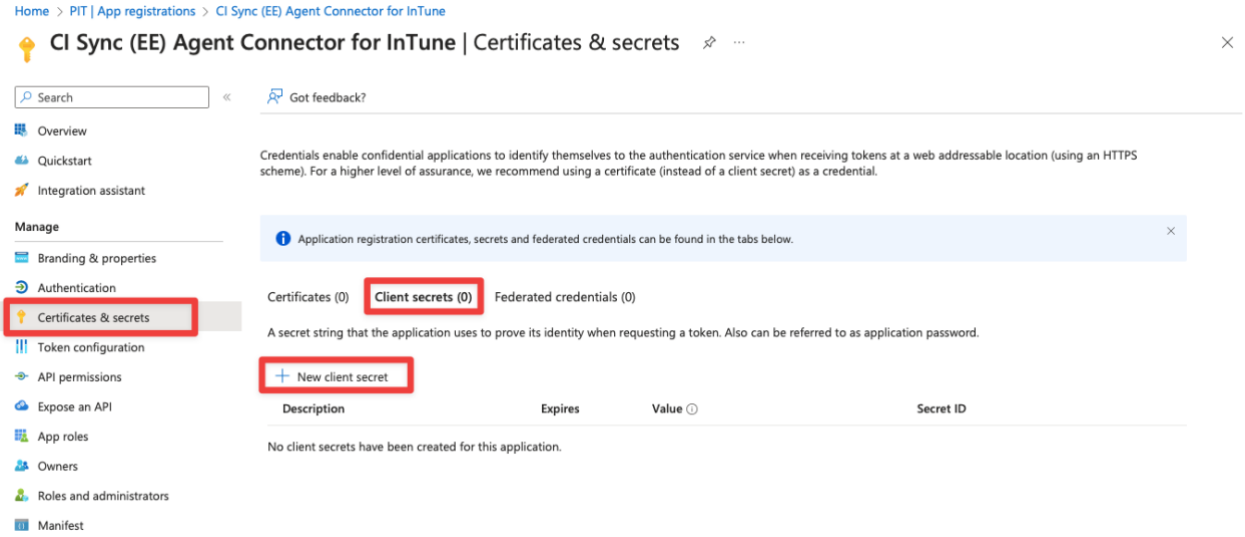
9. Click Grant **admin consent for {Domain}**



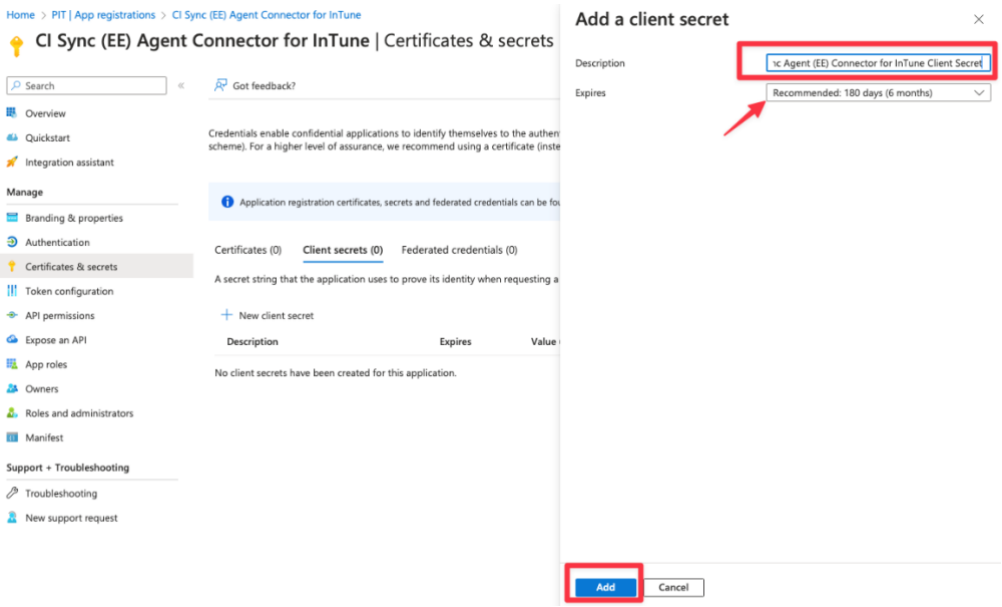
10. Click **Yes** to confirm granting Admin Consent. The resulting screen should look as shown below.



11. Using the left-hand menu, navigate and select **Certificates & secrets**. Select "**Client secrets (0)**" in the middle of the form and then click the "**New client secret**" button.



12. Enter a unique **Description** for the secret associated with this CI Sync (EE) Agent Connector for InTune App Registration (e.g. “CI Sync Agent (EE) Connector for InTune Client Secret”).
13. Then, select a suitable **Expires** duration based on your organisational policy. Finally click the **Add** button.



Guidance Note: It is recommended you set a reminder prior to the expiry date of the Secret (i.e. a reminder to regenerate and update the Secret in the CI Sync (EE) Agent configuration.

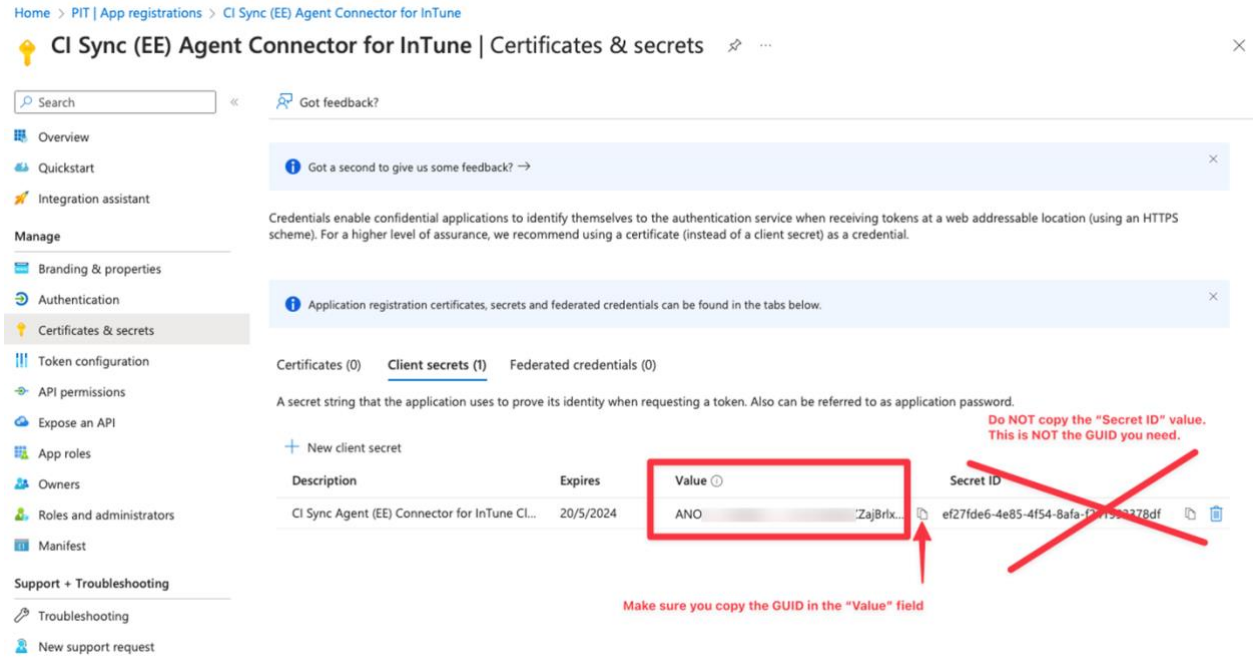
Click here to return to steps index

14. The form now displays the generated **secret value** (shown in the **Value** field).

Guidance Note: The **Value** is only available while you remain on this screen. You must make a copy of the Value GUID before leaving this form.

Make sure you copy the "Value" and NOT the "Secret ID".

- Use the copy option to make a copy of the GUID in the Value field.



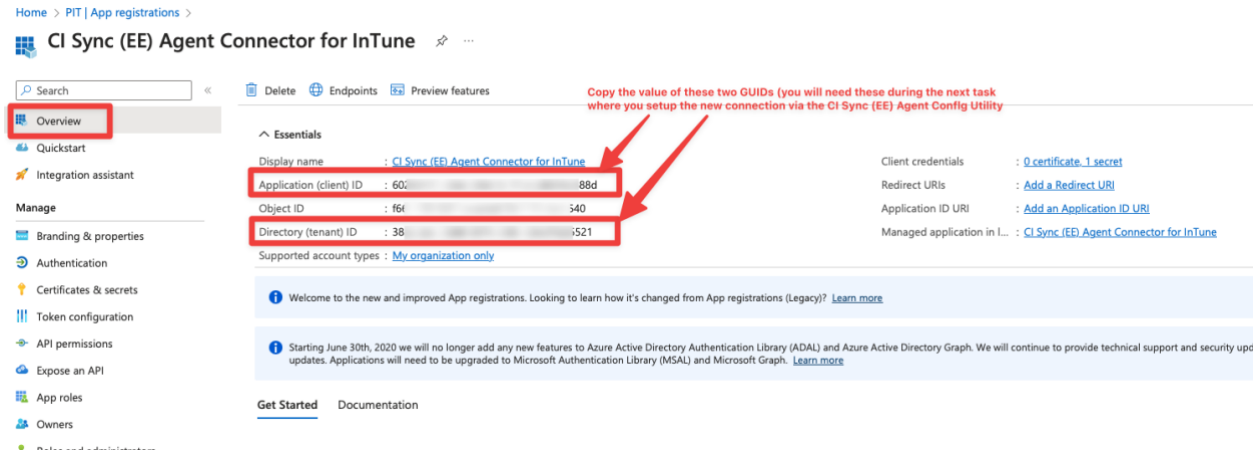
Bookmark: InTune_AppReg_ClientScrtSecretScrn

Data Capture Note: The secret "**Value**" will used by the person performing the next task to setup the source connecting using the CI Sync (EE) Agent Config Utility (i.e. used in Task E2: Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection to InTune. Within Task E2 this value is referred to as the "Client Secret".

It is recommended you securely store a copy of the secret so it can be shared internally with the SME using the CI Sync (EE) Agent Config Utility to setup the InTune Connector.

15. Return to the **Overview** page for the App Registration.

    - Use the copy option to make a copy of the "**Application (client) ID**" GUID value and the "**Directory (tenant) ID" GUID** value..



Bookmark: InTune_AppReg_ClientScrtAppIDScrn

> **Data Capture Note:** The "**Application (client) ID**" and "**Directory (tenant) ID**" GUID values will used by the person performing the next task to setup the source connecting using the CI Sync (EE) Agent Config Utility (i.e. is used in Task E2: Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection to InTune.

16. **For clarity purposes only:** Upon completion of the above tasks, you should have recorded a copy of the values shown below. These values will be needed by the SME performing the next task.

    - The **Secret Value** (i.e. the Client Secret)
    - The **Application (client) ID**
    - The **Directory (tenant) ID**

Task E2: Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection to InTune.

Informational Note

This section includes references to a database referred to as "RecVer". The RecVer database is a small three column checksum style of database used by CI Sync (EE) to deliver the delta record sync functionality (i.e. where CI Sync (EE) only sync's those records which have changed since the previous sync job. Below are some key points about the RecVer database:

1. The RecVer database is currently only supported via Microsoft SQL Server (i.e. RecVer must be a SQL database). In the new future RecVer will also be supported on MongoDB.
2. You will need to create multiple RecVer databases if you are setting up multiple source systems via the CI Sync (EE) Agent (i.e. each Source Connection that supports the delta sync functionality requires its own unique RecVer database.

The follow pages explain how to use the CI Sync (EE) Agent Config Utility to setup an Intune source system connection. You will see references in the screen shots and text which explains how to facilitate the creation and use of your RecVer database.

Guidance Notes

When performing the tasks in this section (to setup a Source System Connection for InTune using the **CI Sync (EE) Agent Config Utility**) you will need to know the following:

1. The path to the SQL server that will host the "RecVer" database.

2. Which authentication type the CI Sync (EE) Agent will use for read/write access to the RecVer database. The **Config Utility** supports the following authentication types for accessing the SQL server:
   - **Integrated Security** (which uses the Windows User Account you created for the CI Sync (EE) Agent Windows Service).
   - **SQL Native Login** (which uses a SQL username and password you enter into the **Config Utility** when creating the Source System Connection).
   - *Note: In either case (Integrated Security of SQL Native Login), the account/login requires minimum (least privileged) access to the databases. Permissions are set either by your DBA (through a manual process) or the Config Utility (through the automated process) as explained below.*

Continue to the next page for further Guidance Notes.

Guidance Notes (continued from prior page)

When performing the tasks in this section (to setup a Source System Connection for InTune using the **CI Sync (EE) Agent Config Utility**) you will also need to know the following:

3. Which of the following approaches you are using to create the CI Sync "RecVer" database on your SQL Server. Your options are as follows:
   - Option 1 – You will use the **Config Utility** to automatically create the RecVer database (and grant the CI Sync (EE) Agent access to the RecVer database) while you are using the **Config Utility** to setup the Source System Connection to InTune.
   - Option 2 – You will engage your SQL Database Administrator (DBA) to manually create the RecVer database (and grant the CI Sync (EE) Agent access to the RecVer database) on your behalf (i.e. in advance) if you don't have permissions to the SQL server and therefore you cannot use the **Config Utility** to automatically create the RecVer database.

If you are clear on the above items, it will improve your use of the **Config Utility** to select the correct options and provide the appropriate credentials when setting up InTune as a Source System Connection for the CI Sync (EE) Agent. See also Appendix A which contains useful diagrams about SQL authentication for the CI Sync (EE) Agent.

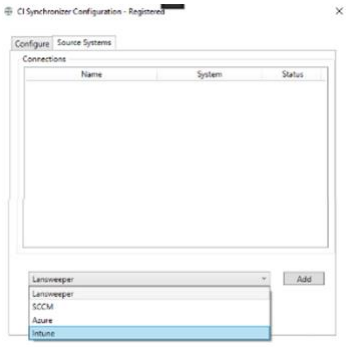
As mentioned in the Guidance Notes on the previous page, there are two options to facilitate the creation of the RecVer database and assigning permissions to it for the CI Sync (Agent).

Option	Title	Additional Details
1	Automatically using the CI Sync (EE) Agent Config Utility	• This option requires you to have the sysadmin (sa) role in the SQL Server that will host the RecVer database. • If you have sa credentials to the SQL Server that will host the RecVer database you can use the **CI Sync (EE) Agent Config Utility** to create the CI Sync (EE) RecVer database**.** • For this option proceed to the very next page under the heading Task E2 (Option 1): Steps to add InTune as a Source Connection to the CI Sync (EE) Agent using the Config Utility to automatically create the RecVer database.
2	Manually via a SQL Database Administrator (DBA)	• If the person executing the CI Sync (EE) Agent Config utility **does not** have the sysadmin (sa) role in the SQL Server, you can have your SQL DBA perform a manual setup of the RecVer database. • For this option you should contact your DBA now and allocate them the manual SQL setup tasks described in Task E2 (Option 2): Steps to add InTune as a Source Connection to the CI Sync (EE) Agent using your SQL Database Administrator (DBA) to manually create the RecVer database in advance. • Once the DBA has performed their allocated tasks you will continue to use **CI Sync (EE) Agent Config Utility** to connect to the RecVer database the DBA has created for you.

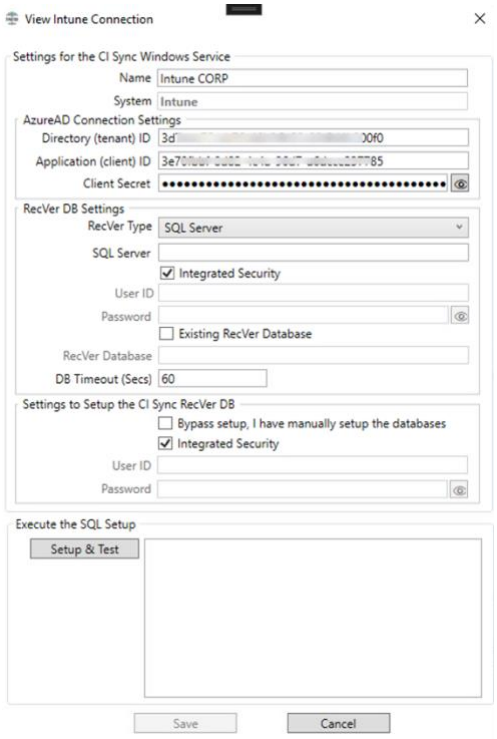
See the instructions in Task E2 (Option 1) and Task E2 (Option 2) on the following pages for detailed steps for each of these two options.

Task E2 (Option 1): Steps to add InTune as a Source Connection to the CI Sync (EE) Agent using the Config Utility to automatically create the RecVer database.

1. Within the CI Sync Agent Configuration Utility, navigate to the **Source Systems** tab, select **InTune** and click the **Add** button.



2. Enter details on the "**Add InTune Connection**" form.

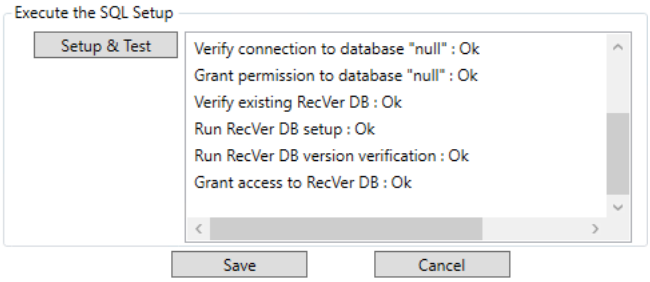


   (a) In the section titled "**Settings for the CI Sync Windows Service**"
   This section defines the settings for the CI Sync (EE) Agent to use as it runs in the background (i.e. after you have exited the Config Utility).
   - **Name**: Enter a name that describes this Source System Connection represents.
   - **Directory (tenant) ID:** Enter the value you captured in **Task E1**.
   - **Application (tenant) ID:** Enter the value you captured in **Task E1**.
   - **Client Secret:** Enter the value you captured in **Task E1**.
   - **RecVer Type:** Set this to SQL Server (*in the future CI Sync will support other database types for the RecVer/delta functionality*).
   - **SQL Server**: Enter the SQL Server instance path in the format: server\instance.
   - **Integrated Security**: Check this checkbox if you want the CI Sync (EE) Agent to authenticate to the SQL Server using the Windows User Account you setup for the CI Sync (EE) Agent Windows Service. Otherwise, uncheck this checkbox and enter the **User ID** and **Password** for a **Native SQL Login** provided to you by the DBA *(Note: Syncfish do not recommend the SQL Login have sysadmin rights).*
   - **Existing RecVer Database**: Do NOT check this checkbox (because the **Config Utility** will be creating the RecVer database for you).
   - **DB Timeout (Secs)**: Sets both the connection and statement execution timeouts. Syncfish recommend 60 (seconds).

   (b) In the section titled "**Settings to Setup the CI Sync RecVer DB**"
   *This section defines how the **Config Utility** will connect to SQL Server to perform the automated SQL setup and configuration steps. The settings do not impact the ongoing running of the CI Sync (EE) Agent*.
   - **Bypass setup, I have manually setup the databases:** Do NOT check this checkbox (because the **Config Utility** will be creating the RecVer database for you).
   - **Integrated Security:** By default, the **Config Utility** will connect to SQL Server to perform the automated SQL setup and configuration steps using Integrated Security (i.e. using your own credentials as the person running the **Config Utility**). Uncheck this checkbox if you want the automated SQL setup and configuration steps to be performed with a native SQL login. Then enter the **User ID** and **Password** of the native SQL login. *Note: the automated process requires the native login to have the SQL sysadmin role.*

Click here to return to steps index

9. In the section titled "**Execute the SQL Setup**", click the **Setup & Test** button. This will create the CI Sync RecVer database (on the provided SQL server), grant the CI Sync (EE) Agent Windows Service Account to the RecVer database and test connectivity to the source system (in this case to InTune) by authentication with the App Registration created earlier.



Informational Note

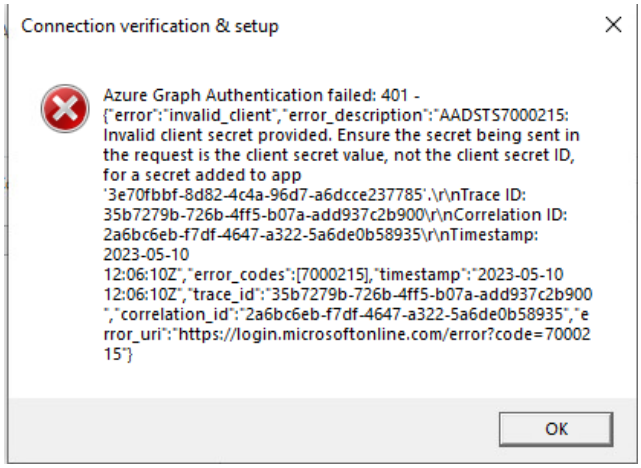
The **Config Utility** performs the following SQL setup and configuration steps:

1. It creates a SQL Login for the Windows User Account you setup for the CI Sync (EE) Agent Windows Service if you selected "Integrated Security" in the section "**Settings for the CI Sync Windows Service**", the **Config Utility**. Or, if instead you are using Native SQL Login (instead of "Integrated Security") then **Config Utility** uses the Native SQL Login details you have supplied and performs the following two steps to grant the Login access to the relevant databases with the permissions noted below.
2. Creates the RecVer database, creates a **SQL user** in the RecVer database for the Login used by the CI Sync (EE) Agent and grants the **SQL user** the agent_role role.
   *Note: Depending on whether you selected "Integrated Security" or not, the **SQL user** is either (a) the Windows User/Service Account (created as a SQL Login in point 1 above) or (b) the native SQL login account.*

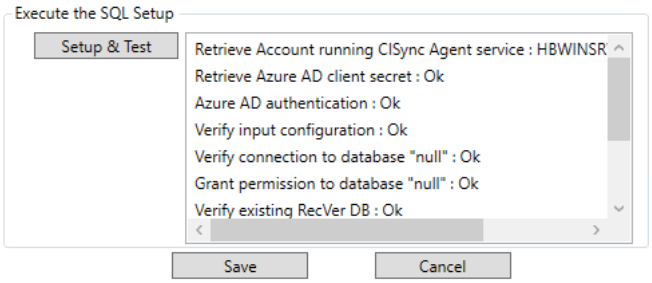
Note: If you are using a Native SQL Login (not Integrated Security) with higher privileges than noted above (e.g. if the account had SQL sysadmin) then the Config Utility will not be able to set the SQL roles noted above. Instead, the Config Utility will not touch the permissions and instead will leave the existing/higher privileges as is. This is not required and not recommended by Syncfish.

Click here to return to steps index

3. If the connection to the source system (in this case via the Microsoft Azure Graph API for InTune) fails, an error will be displayed (see example below). In this case recheck all App Registration details and the various settings entered into the CI Sync (EE) Agent Config Utility on the prior connection setup/settings screen.



4. After a successful test has been completed, click the **Save** button.



5. Finally, click **Yes** to register the CI Sync (EE) Agent with your customer specific CI Sync (EE) SaaS instance.

6. You can now skip the next couple of pages (which describe a SQL DBA doing manual setup of the RecVer database) and proceed to **Task E3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EEE) Agent status and set any remaining connection parameters**.

Click here to return to steps index

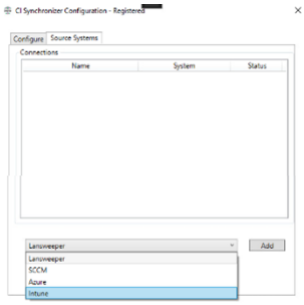
Task E2 (Option 2): Steps to add InTune as a Source Connection to the CI Sync (EE) Agent using your SQL Database Administrator (DBA) to manually create the RecVer database in advance.

1. Ask you SQL DBA to perform the tasks listed in the table below.

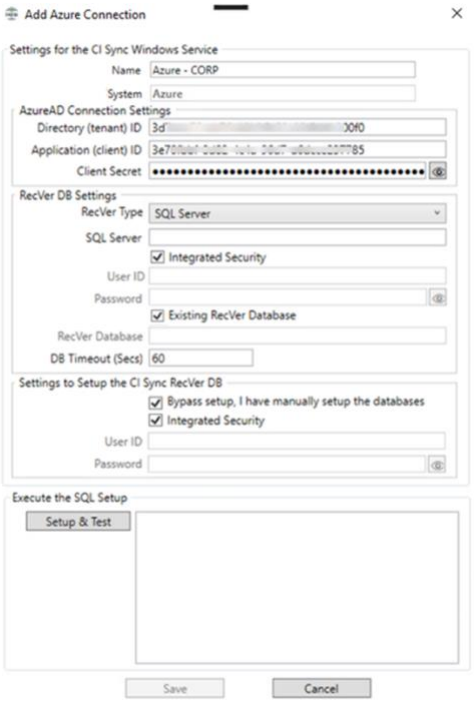
#	DBA Tasks (in SQL Server)	Additional Notes
1	Either **register a Login** (which represents the Windows User Account used by the CI Sync (EE) Agent Windows Service) Or **create a SQL Login** (for the CI Sync (EE) Agent to use).	If you are using **Windows "Integrated Security"** between the CI Sync (EE) Agent and the SQL Server, **then the CI Sync (EE) Agent Windows Service account needs to be registered within SQL Server as a Login**. *Note: The CI Sync (EE) Agent Windows Service account is the one you created during Task 4a: Create a Windows Service Account for the CI Sync (EE) Agent to use.* **Alternatively**, if you are using **Native SQL authentication** between the CI Sync (EE) Agent and the SQL Server, **then a Login will need to be created**. In this case your DBA will need to provide you with the SQL User ID and password details so you can enter them into the Config Utility UI as explained further below.
2	**Execute the SQL Script** provided by Syncfish located here. The script creates the Syncfish RecVer database. The script is: Cisee-recver-create-script.sql	Notes for the DBA when running the script: • The script has a placeholder called $(database_name) as the replacement for the RecVer database name. • For an InTune Source System Syncfish recommend "**cisee_recver_intune**" as the RecVer database name. • To run the setup script using the sqlcmd utility, the database_name parameter needs to be passed in using the -v switch: *> sqlcmd -I -S localhost -v database_name=" **cisee_recver_intune**" -i "cisee-recver-create-script.sql"* • If running the script in SSMS, the value $(database_name) (including the $ and brackets) needs to be replaced with the name for the recver database.
3	**Map the Login as a user** in the **RecVer** database and **grant the user** the **role_agent** role.	Map the user and grant the role to the RecVer database created by the SQL script above.

Click here to return to steps index

2. Within the CI Sync Agent Configuration Utility, navigate to the **Source Systems** tab, select **InTune** and click the **Add** button.



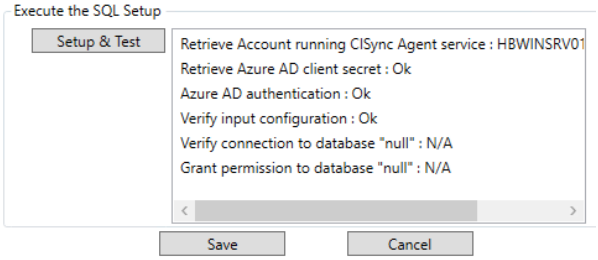
3. Enter details on the "**Add InTune Connection**" form.



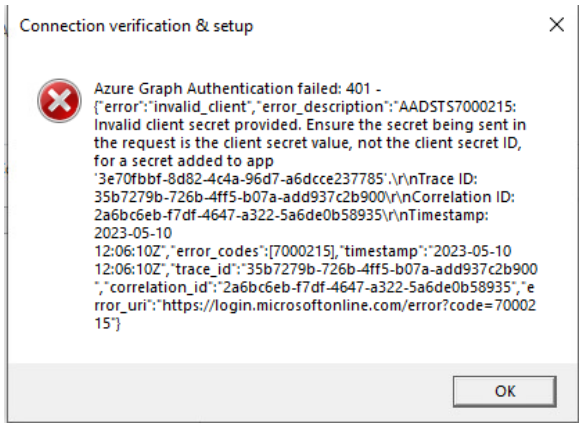
   (a) In the section titled "**Settings for the CI Sync Windows Service**"
   This section defines the settings for the CI Sync (EE) Agent to use as it runs in the background (i.e. after you have exited the Config Utility).
   - **Name**: Enter a name that describes this Source System Connection represents.
   - **Directory (tenant) ID**: Enter the value you captured in **Task E1**.
   - **Application (tenant) ID**: Enter the value you captured in **Task E1**.
   - **Client Secret**: Enter the value you captured in **Task E1**.
   - **RecVer Type**: Set this to SQL Server (*in the future CI Sync will support other database types for the RecVer/delta functionality*).
   - **SQL Server**: Enter the SQL Server instance path in the format: server\instance.
   - **Integrated Security**: Check this checkbox if you want the CI Sync (EE) Agent to authenticate to the SQL Server using the Windows User Account you setup for the CI Sync (EE) Agent Windows Service. Otherwise, uncheck this checkbox and enter the **User ID** and **Password** for a **Native SQL Login** provided to you by the DBA.
   - **Existing RecVer Database**: Check this checkbox (because your SQL DBA has already manually created the RecVer database for you).
   - **DB Timeout (Secs)**: Sets both the connection and statement execution timeouts. Syncfish recommend 60 (seconds).

   (b) In the section titled "**Settings to Setup the CI Sync RecVer DB**"
   *This section defines how the **Config Utility** will connect to SQL Server to perform the automated SQL setup and configuration steps. Because your SQL DBA has already manually created the RecVer database you use this section to confirm this is the case.*
   - **Bypass setup, I have manually setup the databases:** Check this checkbox (because your SQL DBA has already manually created the RecVer database and set the relevant permissions for you).

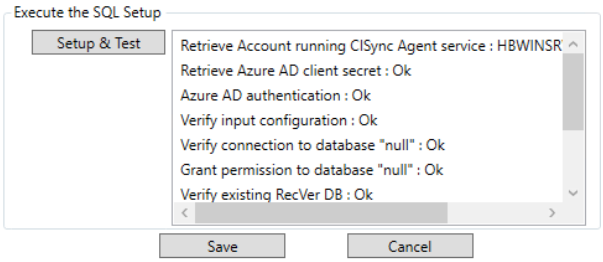
4. In the section titled "**Execute the SQL Setup**", click the **Setup & Test** button. This will create the CI Sync RecVer database (on the provided SQL server), grant the CI Sync (EE) Agent Windows Service Account to the RecVer database and test connectivity to the source system (in this case to InTune) by authentication with the App Registration created earlier.



7. If the connection to the source system (in this case via the Microsoft Azure Graph API for InTune) fails, an error will be displayed (see example below). In this case recheck all App Registration details and the various settings entered into the CI Sync (EE) Agent Config Utility on the prior connection setup/settings screen.

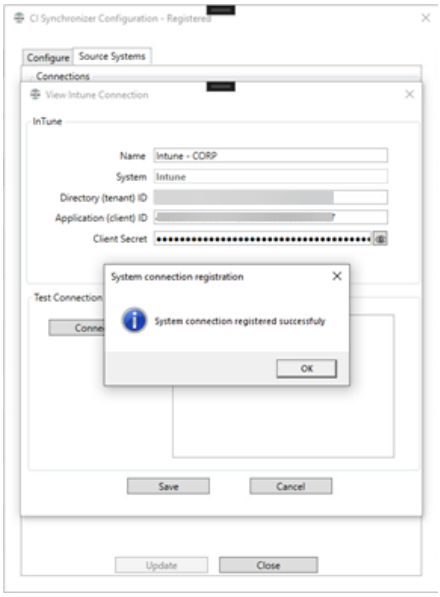


8. After a successful test has been completed, click the **Save** button.



5. Finally, click **Yes** to register this new connection (via your CI Sync (EE) Agent) with your customer specific CI Sync (EE) SaaS instance.

6. Assuming no errors you will see a confirmation as shown below.



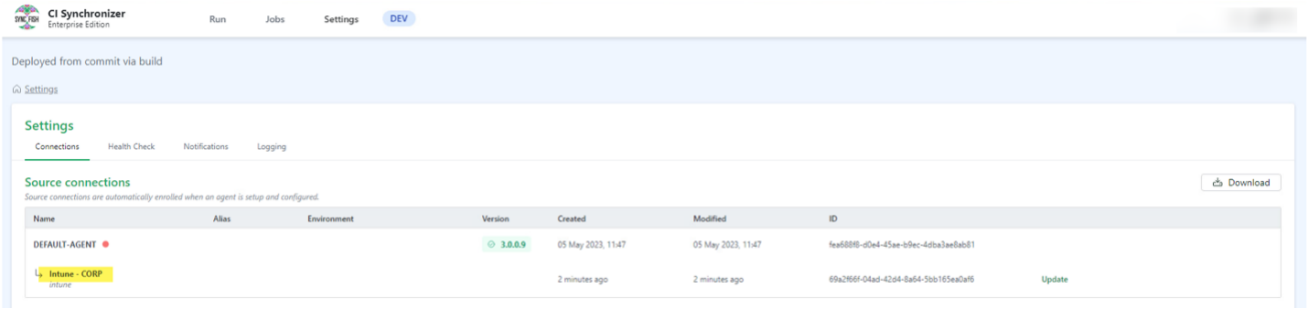
7. Click **OK.**
8. The new connection will appear in the Connections List of the CI Sync (EE) Agent.

9. You can now proceed to Task E3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EE) Agent status and set remaining connection parameters. and to set any remaining connection parameters.

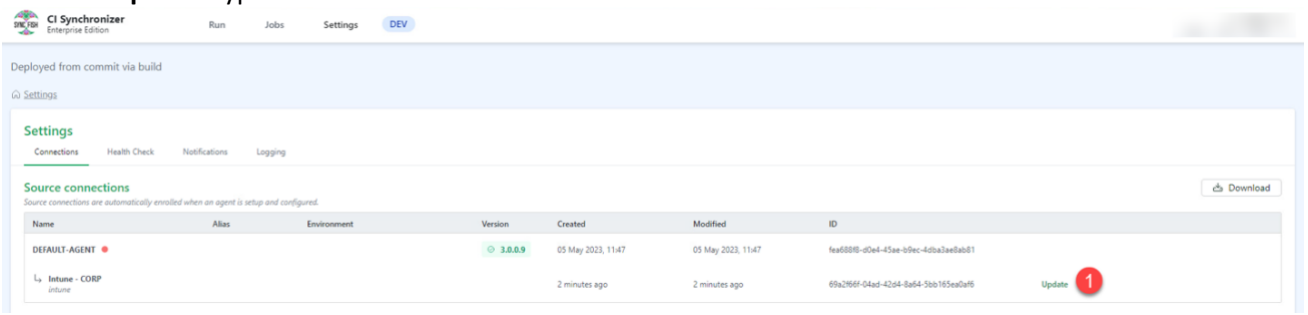
Click here to return to steps index

Task E3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EE) Agent status and set remaining connection parameters.

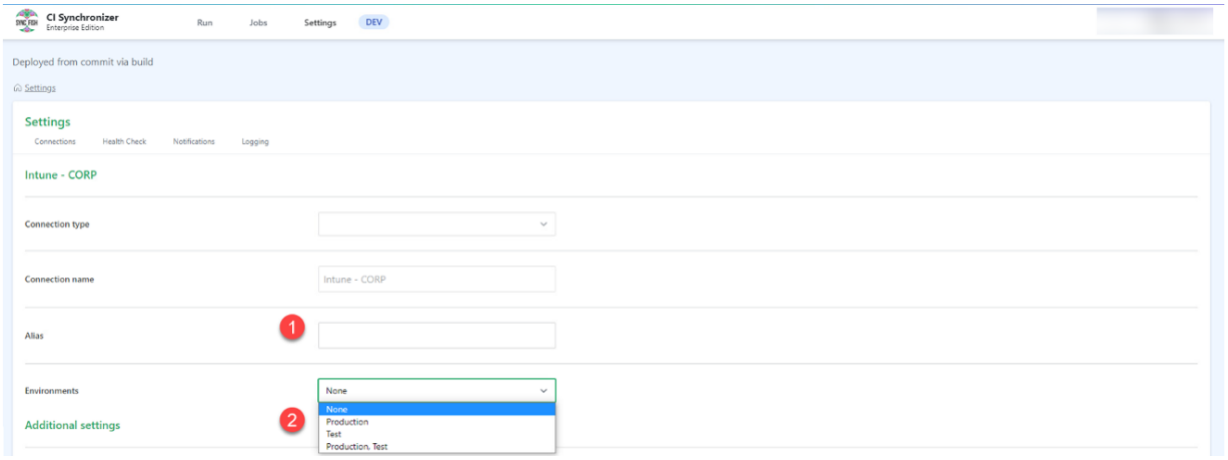
1. Login to your CI Sync (EE) SaaS instance at https://***YourCo***.syncfish.app
2. In the CI Sync UI, navigate to **Settings > Connections**. The new Source System Connection should appear in the Source connections list.



3. Click the **Update** hyperlink



4. Use the settings form to as follows:
   - Enter an **Alias** (optional)
   - Select the **Environment**/s the new source connection can be used for. In most cases a Source System Connection is used for both Test and Production sync jobs (as distinct from the Destination Connections which can only be either Test or Production).
   - Check the **consent** checkbox and Click the **Save connection** button.



You have now completed all tasks to add InTune as a Source Connection in the CI Sync (EE) Agent. You should now proceed back to Step 6 - Configure your ServiceNow to be ready for CI Sync (EE) (and subsequent steps).

Appendix F – How to add Azure as Source Connection to the CI Sync (EE) Agent

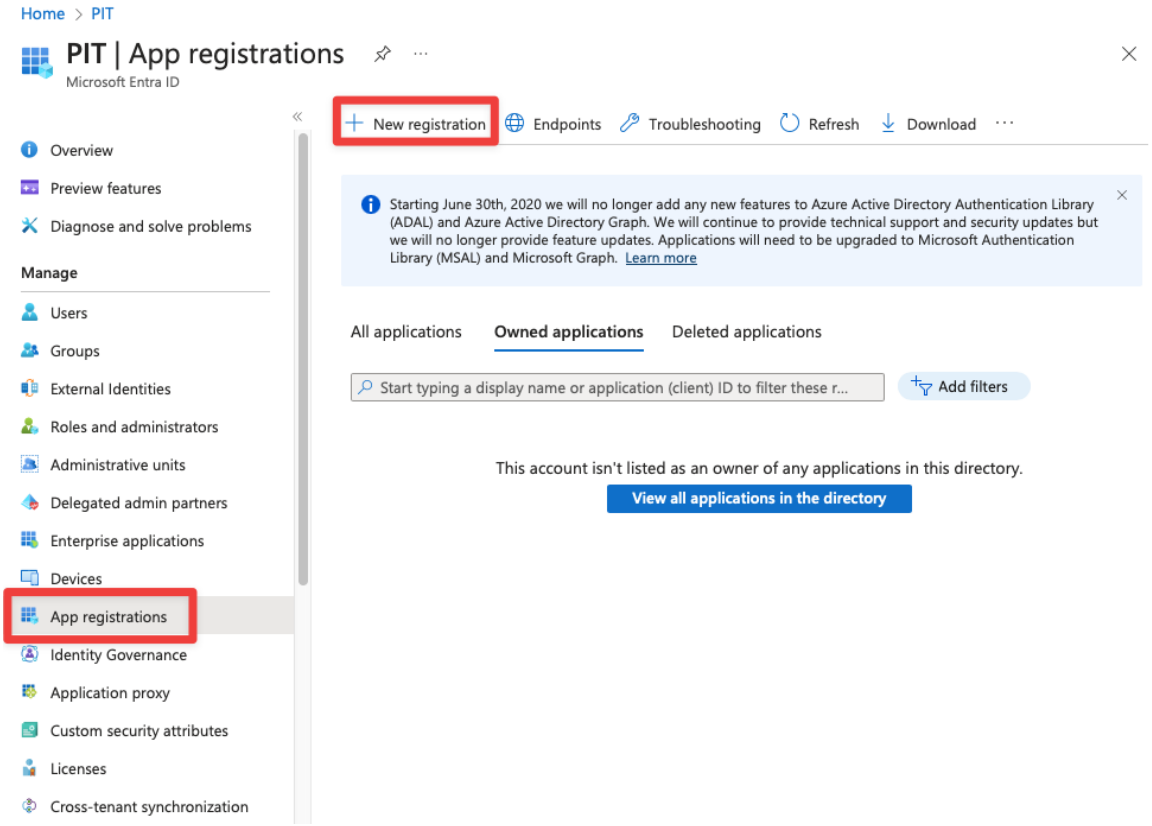
- Task F1: Create a new App Registration to grant permissions for the CI Sync (EE) Agent to access Azure.
- Task F2: Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection to Azure.
- Task F3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EEE) Agent status and set any remaining connection parameters.

Task F1: Create a new App Registration to grant permissions for the CI Sync (EE) Agent to access Azure.

Context Notes:

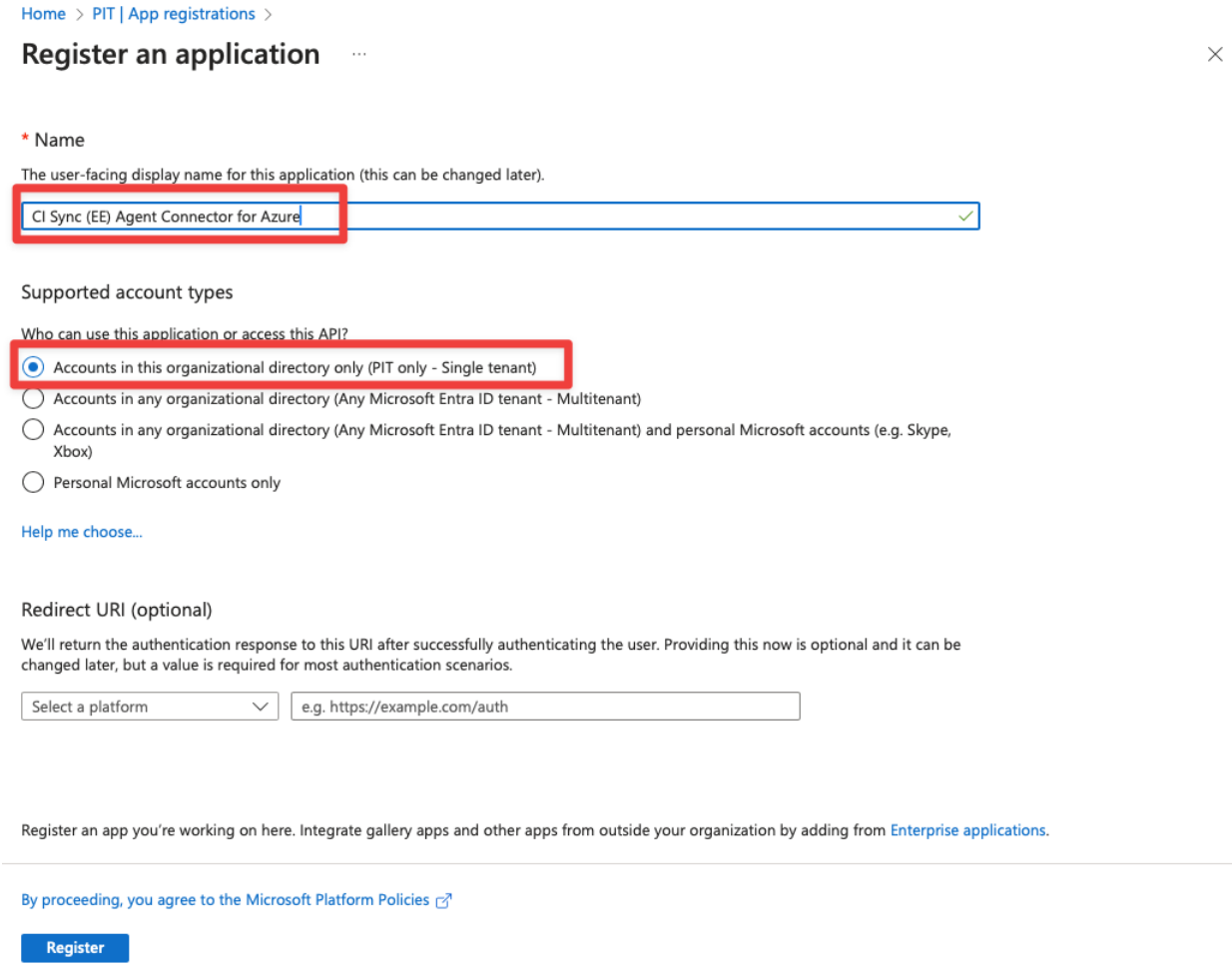
- In Azure ADD (AAD) an App Registration is used to define a Service Principal for the purpose of authenticating a source application to a destination system/application.
- The **App Registration created in this section relates to the Azure Source Connection** created within the CI Sync (EE) Agent. This Azure specific App Registration is different to the App Registration for the CI Sync (EE) Agent itself that was create in Step 3.

1. In the **Azure Portal**, navigate to **Azure Active Directory** -> **App Registrations** and click **New Registration**

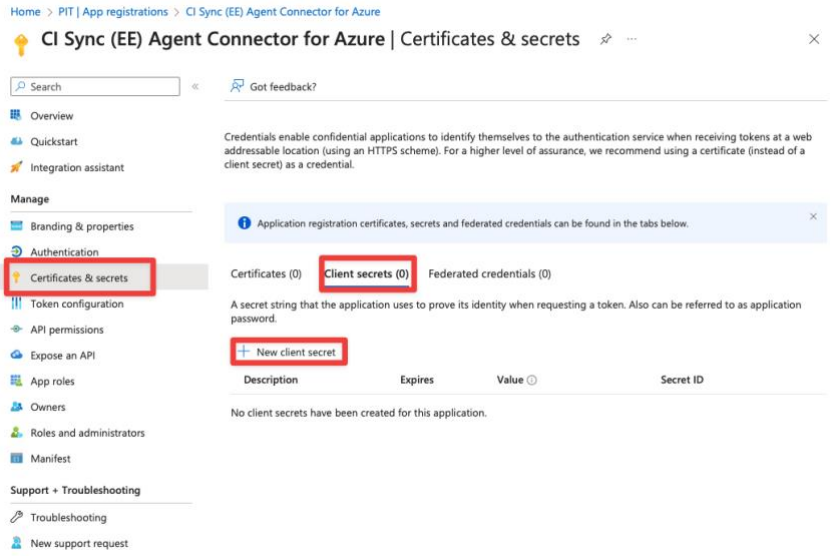


Click here to return to steps index

2. On the **Register an application** form complete as follows:
   - Enter the **Name** (Note: Syncfish recommend using "CI Sync (EE) Agent Connector for Azure")
   - Under **Supported account types** select "**Accounts in this organizational directory only (*{Your Domain/Tenant Name}* only - Single tenant)**"
   - Click **Register**

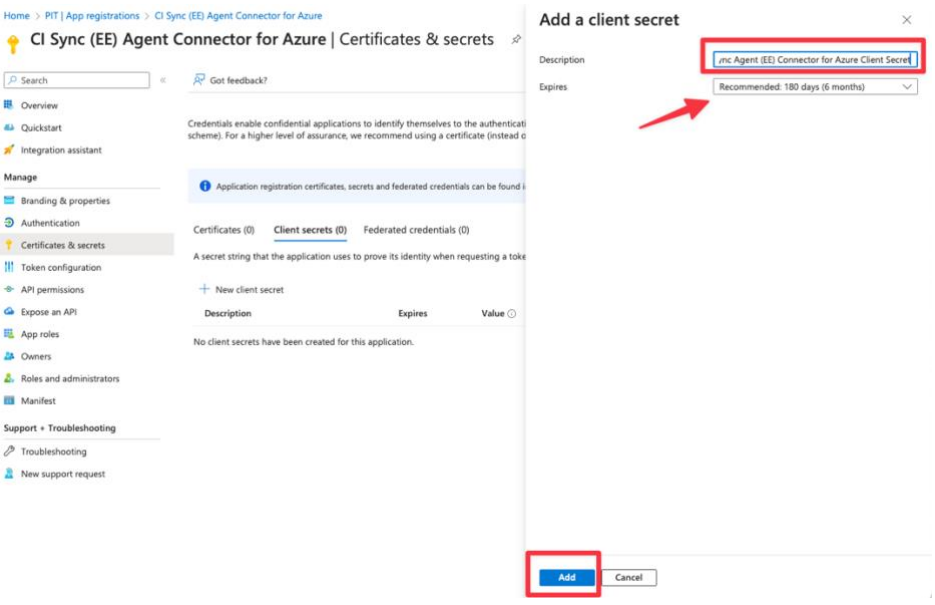


3. Using the left-hand menu, navigate and select **Certificates & secrets**. Select "**Client secrets (0)**" in the middle of the form and then click the "**New client secret**" button.



4. Enter a unique **Description** for the secret associated with this CI Sync (EE) Agent Connector for Azure App Registration (e.g. "CI Sync Agent (EE) Connector for Azure Client Secret").

5. Then, select a suitable **Expires** duration based on your organisational policy. Finally click the **Add** button.



Guidance Note: It is recommended you set a reminder prior to the expiry date of the Secret (i.e. a reminder to regenerate and update the Secret in the CI Sync (EE) Agent configuration.

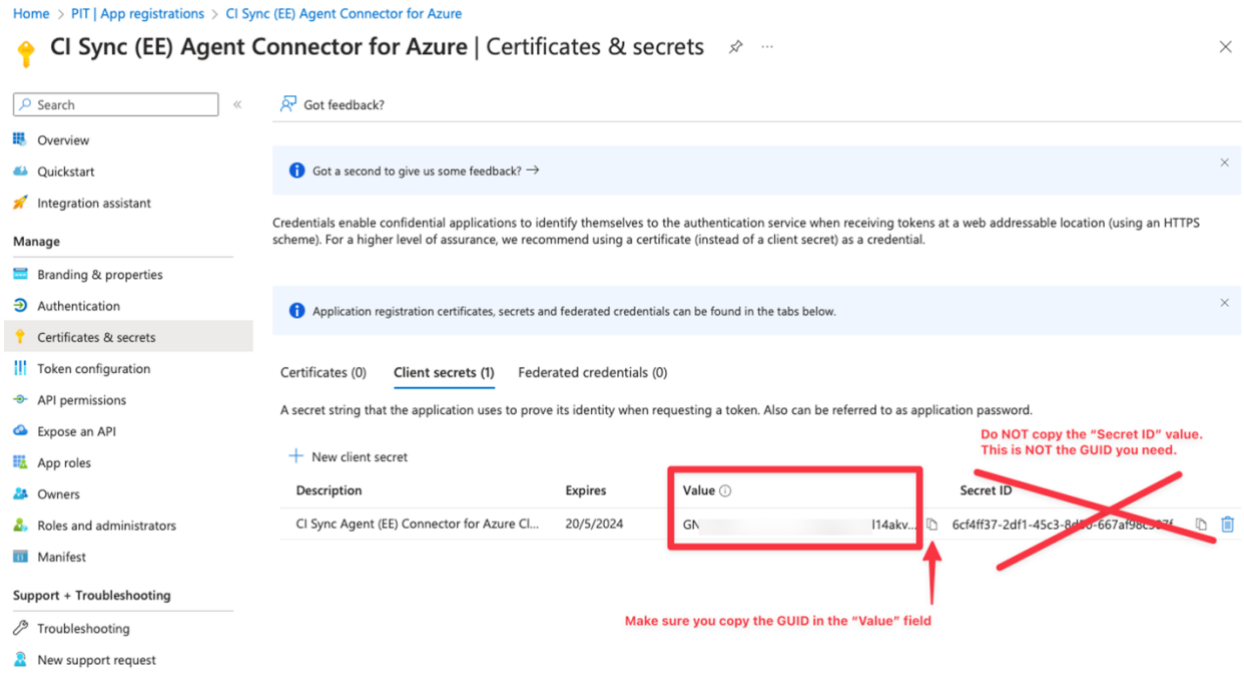
Click here to return to steps index

6. The form now displays the generated **secret value** (shown in the **Value** field).

Guidance Note: The **Value** is only available while you remain on this screen. You must make a copy of the Value GUID before leaving this form.

Make sure you copy the "Value" and NOT the "Secret ID".

- Use the copy option to make a copy of the GUID in the Value field.



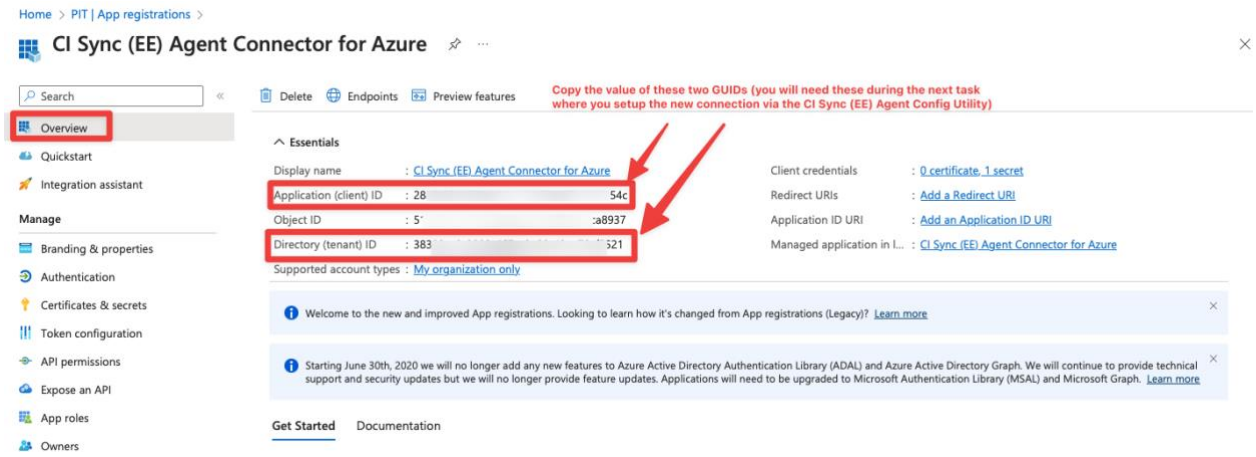
Bookmark: Azure_AppReg_ClientScrtSecretScrn

Data Capture Note: The secret "**Value**" will used by the person performing the next task to setup the source connecting using the CI Sync (EE) Agent Config Utility (i.e. used in Task F2: Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection to Azure.). Within Task F2 this value is referred to as the "Client Secret".

It is recommended you securely store a copy of the secret so it can be shared internally with the SME using the CI Sync (EE) Agent Config Utility to setup the Azure Connector.

7. Return to the **Overview** page for the App Registration.

   - Use the copy option to make a copy of the "**Application (client) ID**" GUID value and the "**Directory (tenant) ID" GUID** value.



Bookmark: Azure_AppReg_ClientScrtAppIDScrn

Data Capture Note: The "**Application (client) ID**" and "**Directory (tenant) ID**" GUID values will used by the person performing the next task to setup the source connecting using the CI Sync (EE) Agent Config Utility (i.e. is used in Task F2: Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection to Azure.).

8. **For clarity purposes only:** Upon completion of the above tasks, you should have recorded a copy of the values shown below. These values will be needed by the SME performing the next task.

   - The **Secret Value** (i.e. the Client Secret)
   - The **Application (client) ID**
   - The **Directory (tenant) ID**

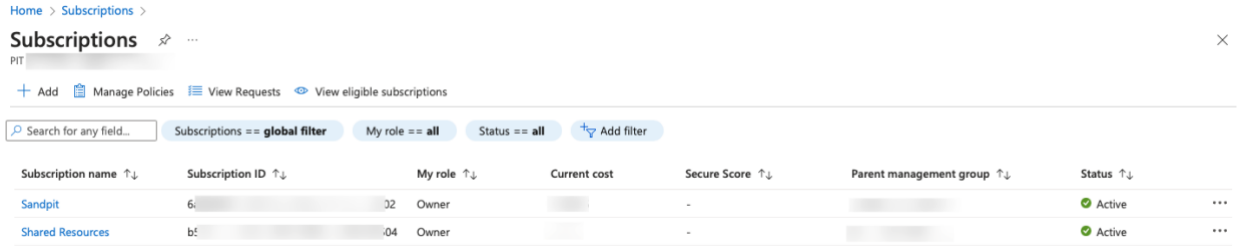
Context Note

Based on the preceding steps a new App Registration has been created (i.e. the new App Registration which provides authentication of the CI Sync (EE) Agent Source Connection to Azure).

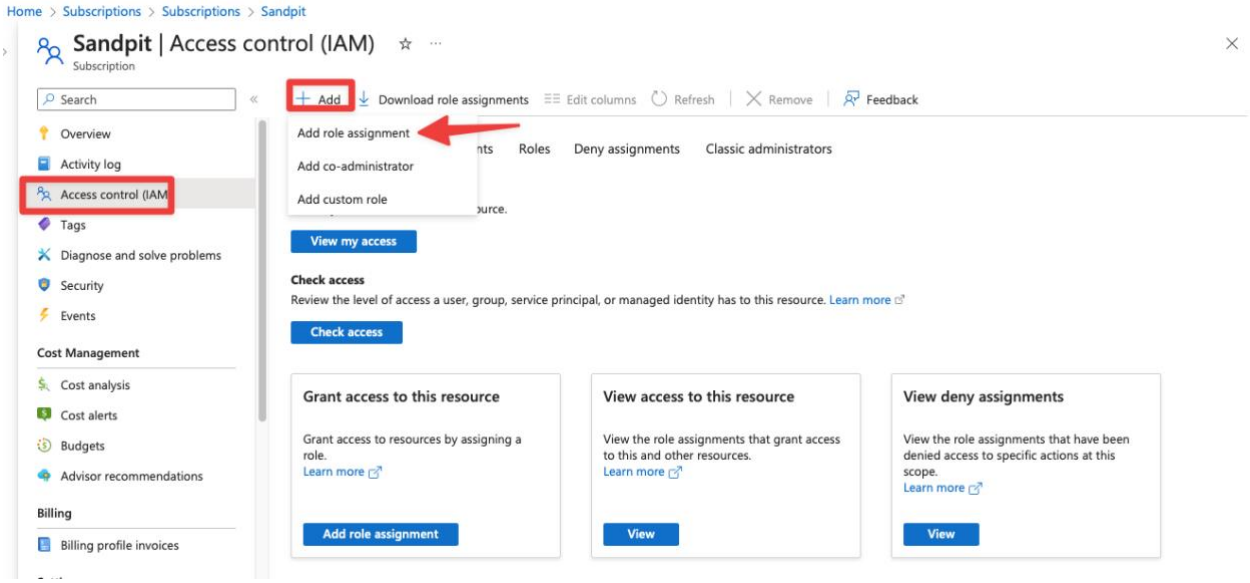
The subsequent steps below are required to grant the App Registration permissions to read Azure Resources (i.e. objects such as subscriptions, resource groups, virtual networks and so on).

The subsequent steps should be repeated for each Azure Subscription you want to grant read access to the CI Sync (EE) Agent Source Connection to Azure. By granting read access to a given Azure Subscription (or multiple Azure Subscriptions) you can synchronize all supported resource types within the Subscription.

9. In the **Azure Portal**, navigate to **Subscriptions** and select/click into the particular Azure Subscription you wish to grant access to the CI Sync (EE) Agent Source Connection to Azure.

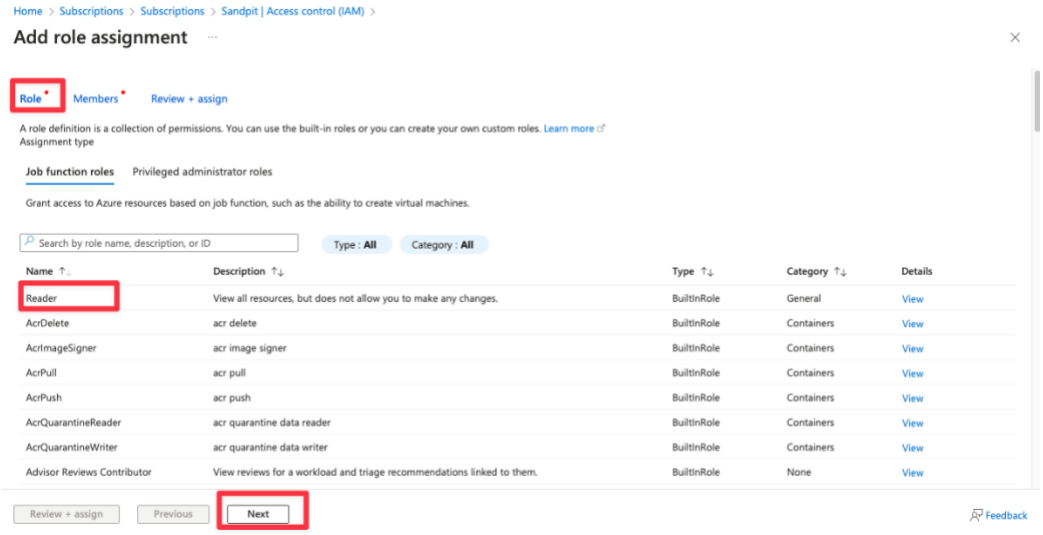


10. Select **Access control (IAM)** from the left hand menu, click the **Add** button and then select **Add role assignment** from the drop down menu.

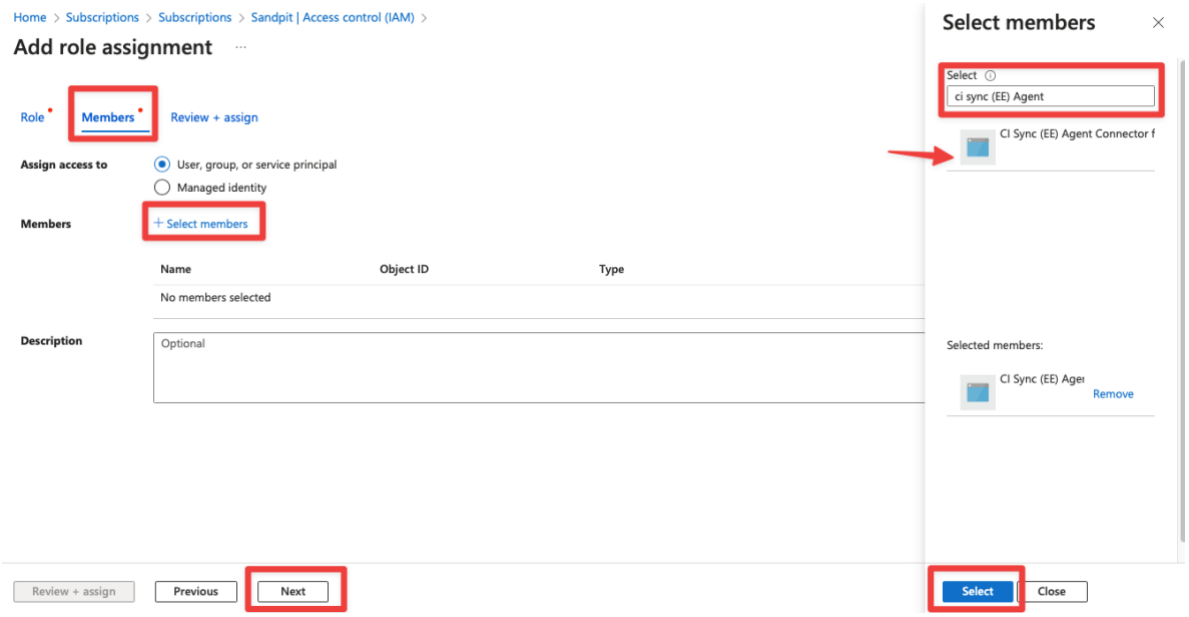


Click here to return to steps index

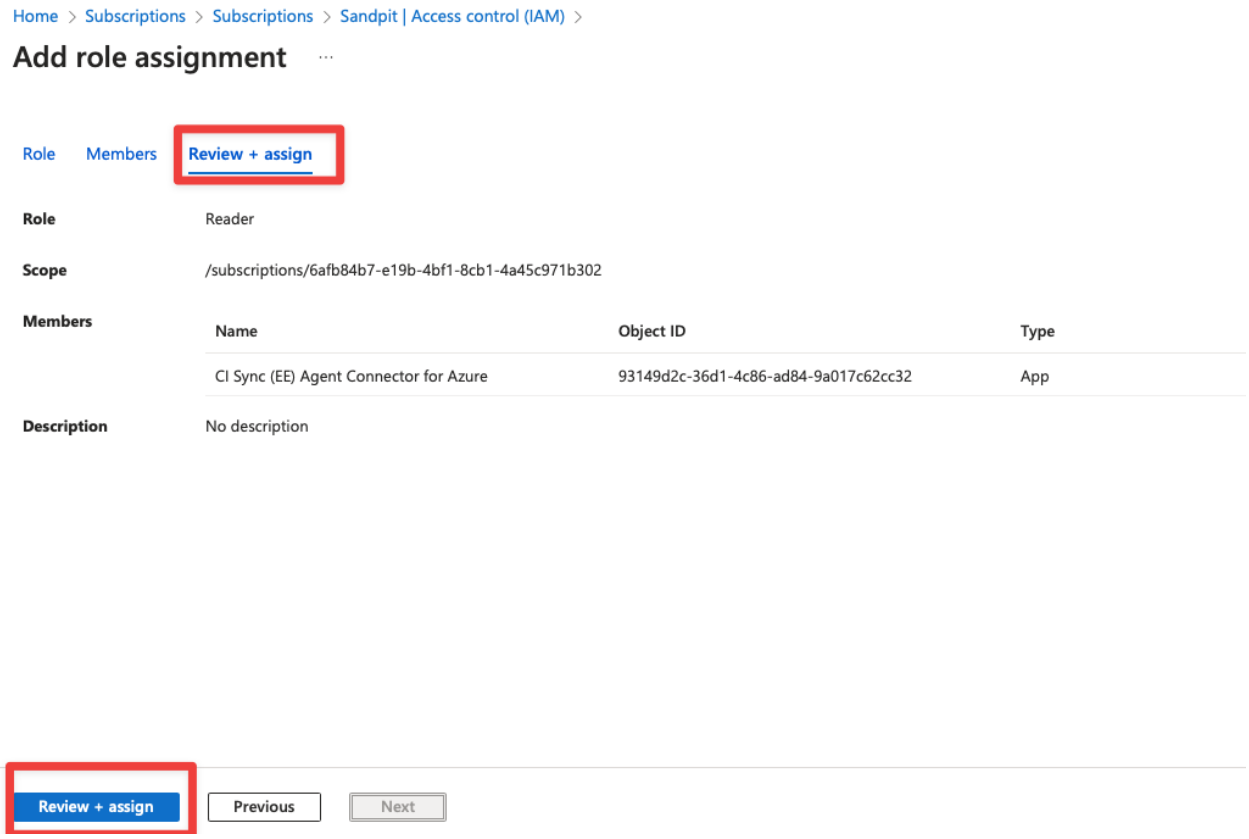
11. On the **Role tab**, select the **Reader** role and click the **Next** button.



12. On the **Members tab**, click the "**+ Select members**" link, then use the **Filter/Select** box enter as sufficient amount of text to locate the App Registration created earlier (the one that represents the CI Sync (EE) Agent Connector for Azure), then the **Select** button (to select the App Registration) and finally the **Next** button.



13. On the **Review + assign tab**, click the **Review + assign** button.



14. You have now granted the App Registration object (i.e. the CI Sync (EE) Agent Connector for Azure) read permissions to an Azure Subscription which will allow you to use the CI Sync User Interface to schedule synchronization jobs using that same Azure Subscription as a synchronization source.

Click here to return to steps index

Task F2: Use the CI Sync (EE) Agent Config Utility to Setup a Source Connection to Azure.

Informational Note

This section includes references to a database referred to as "RecVer". The RecVer database is a small three column checksum style of database used by CI Sync (EE) to deliver the delta record sync functionality (i.e. where CI Sync (EE) only sync's those records which have changed since the previous sync job. Below are some key points about the RecVer database:

1. The RecVer database is currently only supported via Microsoft SQL Server (i.e. RecVer must be a SQL database). In the new future RecVer will also be supported on MongoDB.
2. You will need to create multiple RecVer databases if you are setting up multiple source systems via the CI Sync (EE) Agent (i.e. each Source Connection that supports the delta sync functionality requires its own unique RecVer database.

The follow pages explain how to use the CI Sync (EE) Agent Config Utility to setup an Azure source system connection. You will see references in the screen shots and text which explains how to facilitate the creation and use of your RecVer database.

Guidance Notes

When performing the tasks in this section (to setup a Source System Connection for Azure using the **CI Sync (EE) Agent Config Utility**) you will need to know the following:

1. The path to the SQL server that will host the "RecVer" database.

2. Which authentication type the CI Sync (EE) Agent will use for read/write access to the RecVer database. The **Config Utility** supports the following authentication types for accessing the SQL server:
   - **Integrated Security** (which uses the Windows User Account you created for the CI Sync (EE) Agent Windows Service).
   - **SQL Native Login** (which uses a SQL username and password you enter into the **Config Utility** when creating the Source System Connection).
   - *Note: In either case (Integrated Security of SQL Native Login), the account/login requires minimum (least privileged) access to the databases. Permissions are set either by your DBA (through a manual process) or the Config Utility (through the automated process) as explained below.*

Continue to the next page for further Guidance Notes.

Guidance Notes (continued from prior page)

When performing the tasks in this section (to setup a Source System Connection for Azure using the **CI Sync (EE) Agent Config Utility**) you will need to know the following:

3. Which of the following you will use to create the CI Sync "RecVer" database. Your options are as follows:
   - Option 1 – You will use the **Config Utility** to automatically create the RecVer database (and grant the CI Sync (EE) Agent access to the RecVer database) while you are using the **Config Utility** to setup the Source System Connection to Azure.
   - Option 2 – You will engage your SQL Database Administrator (DBA) to manually create the RecVer database (and grant the CI Sync (EE) Agent access to the RecVer database) on your behalf (i.e. in advance) if you don't have permissions to the SQL server and therefore you cannot use the **Config Utility** to automatically create the RecVer database.

If you are clear on the above items, it will improve your use of the **Config Utility** to select the correct options and provide the appropriate credentials when setting up Azure as a Source System Connection for the CI Sync (EE) Agent. See also Appendix A which contains useful diagrams about SQL authentication for the CI Sync (EE) Agent.

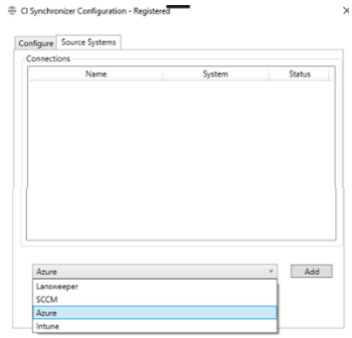
As mentioned in the Guidance Notes above and on the previous page, there are two options to facilitate the creation of the RecVer database and assigning permissions to it for the CI Sync (Agent).

Option	Title	Additional Details
1	Automatically using the CI Sync (EE) Agent Config Utility	• This option requires you to have the sysadmin (sa) role in the SQL Server that will host the RecVer database. • If you have sa credentials to the SQL Server that will host the RecVer database you can use the **CI Sync (EE) Agent Config Utility** to create the CI Sync (EE) RecVer database**.** • For this option proceed to the very next page under the heading Task F2 (Option 1): Steps to add Azure as a Source Connection to the CI Sync (EE) Agent using the Config Utility to automatically create the RecVer database.
2	Manually via a SQL Database Administrator (DBA)	• If the person executing the CI Sync (EE) Agent Config utility **does not** have the sysadmin (sa) role in the SQL Server, you can have your SQL DBA perform a manual setup of the RecVer database. • For this option you should contact your DBA now and allocate them the manual SQL setup tasks described in Task F2 (Option 2): Steps to add Azure as a Source Connection to the CI Sync (EE) Agent using your SQL Database Administrator (DBA) to manually create the RecVer database in advance. • Once the DBA has performed their allocated tasks you will continue to the **CI Sync (EE) Agent Config Utility** to connect to the RecVer database the DBA has created for you.

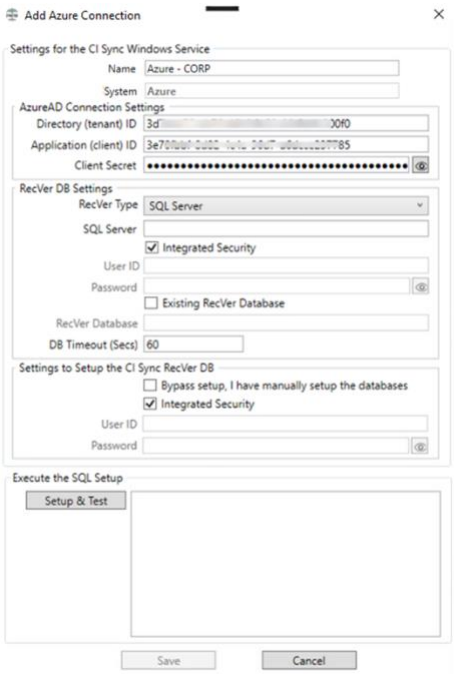
See the instructions in Task F2 (Option 1) and Task F2 (Option 2) on the following pages for detailed steps for each of these two options.

Task F2 (Option 1): Steps to add Azure as a Source Connection to the CI Sync (EE) Agent using the Config Utility to automatically create the RecVer database.

1. Within the CI Sync Agent Configuration Utility, navigate to the **Source Systems** tab, select **Azure** and click the **Add** button.



2. Enter details on the "**Add Azure Connection**" form.

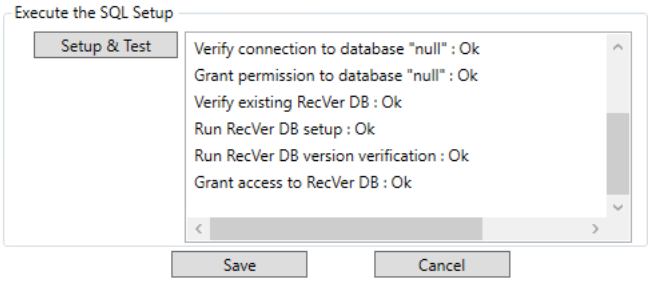


   (a) In the section titled "**Settings for the CI Sync Windows Service**"
   This section defines the settings for the CI Sync (EE) Agent to use as it runs in the background (i.e. after you have exited the Config Utility).
   - **Name**: Enter a name that describes this Source System Connection represents.
   - **Directory (tenant) ID:** Enter the value you captured in **Task F1**.
   - **Application (tenant) ID:** Enter the value you captured in **Task F1**.
   - **Client Secret:** Enter the value you captured in **Task F1**.
   - **RecVer Type:** Set this to SQL Server (*in the future CI Sync will support other database types for the RecVer/delta functionality*).
   - **SQL Server**: Enter the SQL Server instance path in the format: server\instance.
   - **Integrated Security**: Check this checkbox if you want the CI Sync (EE) Agent to authenticate to the SQL Server using the Windows User Account you setup for the CI Sync (EE) Agent Windows Service. Otherwise, uncheck this checkbox and enter the **User ID** and **Password** for a **Native SQL Login** provided to you by the DBA *(Note: Syncfish do not recommend the SQL Login have sysadmin rights).*
   - **Existing RecVer Database**: Do NOT check this checkbox (because the **Config Utility** will be creating the RecVer database for you).
   - **DB Timeout (Secs)**: Sets both the connection and statement execution timeouts. Syncfish recommend 60 (seconds).

   (b) In the section titled "**Settings to Setup the CI Sync RecVer DB**"
   *This section defines how the **Config Utility** will connect to SQL Server to perform the automated SQL setup and configuration steps. The settings do not impact the ongoing running of the CI Sync (EE) Agent*.
   - **Bypass setup, I have manually setup the databases:** Do NOT check this checkbox (because the **Config Utility** will be creating the RecVer database for you).
   - **Integrated Security:** By default, the **Config Utility** will connect to SQL Server to perform the automated SQL setup and configuration steps using Integrated Security (i.e. using your own credentials as the person running the **Config Utility**). Uncheck this checkbox if you want the automated SQL setup and configuration steps to be performed with a native SQL login. Then enter the **User ID** and **Password** of the native SQL login. *Note: the automated process requires the native login to have the SQL sysadmin role.*

Click here to return to steps index

10. In the section titled "**Execute the SQL Setup**", click the **Setup & Test** button. This will create the CI Sync RecVer database (on the provided SQL server), grant the CI Sync (EE) Agent Windows Service Account to the RecVer database and test connectivity to the source system (in this case to Azure) by authentication with the App Registration created earlier.



Informational Note

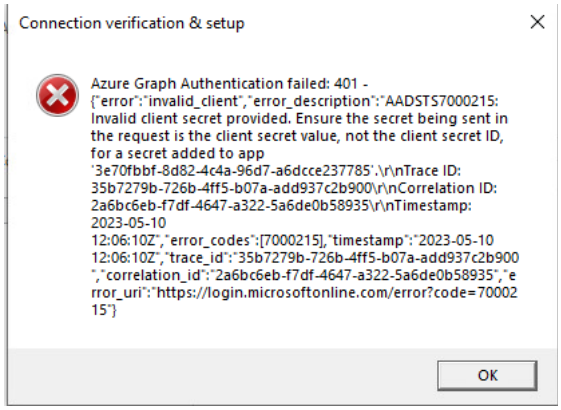
The **Config Utility** performs the following SQL setup and configuration steps:

3. It creates a SQL Login for the Windows User Account you setup for the CI Sync (EE) Agent Windows Service if you selected "Integrated Security" in the section "**Settings for the CI Sync Windows Service**", the **Config Utility**. Or, if instead you are using Native SQL Login (instead of "Integrated Security") then **Config Utility** uses the Native SQL Login details you have supplied and performs the following two steps to grant the Login access to the relevant databases with the permissions noted below.
4. Creates the RecVer database, creates a **SQL user** in the RecVer database for the Login used by the CI Sync (EE) Agent and grants the **SQL user** the agent_role role.
   *Note: Depending on whether you selected "Integrated Security" or not, the **SQL user** is either (a) the Windows User/Service Account (created as a SQL Login in point 1 above) or (b) the native SQL login account.*

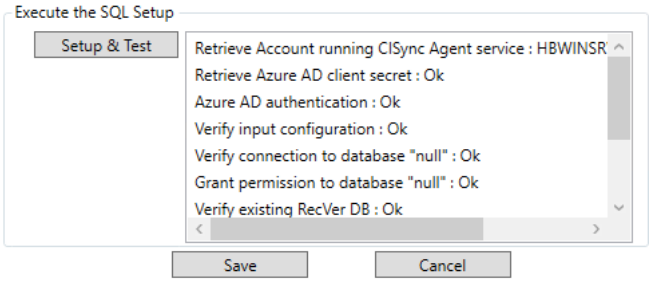
Note: If you are using a Native SQL Login (not Integrated Security) with higher privileges than noted above (e.g. if the account had SQL sysadmin) then the Config Utility will not be able to set the SQL roles noted above. Instead, the Config Utility will not touch the permissions and instead will leave the existing/higher privileges as is. This is not required and not recommended by Syncfish.

Click here to return to steps index

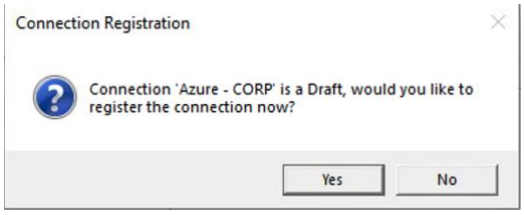
3. If the connection to the source system (in this case via the Microsoft Azure Graph API for Azure) fails, an error will be displayed (see example below). In this case recheck all App Registration details and the various settings entered into the CI Sync (EE) Agent Config Utility on the prior connection setup/settings screen.



4. After a successful test has been completed, click the **Save** button.



5. Finally, click **Yes** to register the CI Sync (EE) Agent with your customer specific CI Sync (EE) SaaS instance.



6. You can now skip the next couple of pages (which describe a SQL DBA doing manual setup of the RecVer database) and proceed to Task F3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EE) Agent status and set remaining connection parameters. and set any remaining connection parameters.

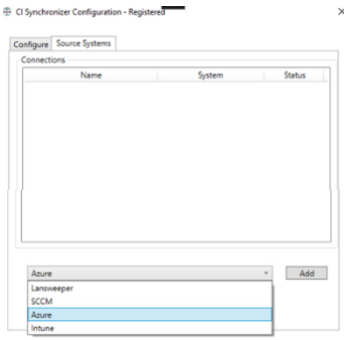
Task F2 (Option 2): Steps to add Azure as a Source Connection to the CI Sync (EE) Agent using your SQL Database Administrator (DBA) to manually create the RecVer database in advance.

1. Ask you SQL DBA to perform the tasks listed in the table below.

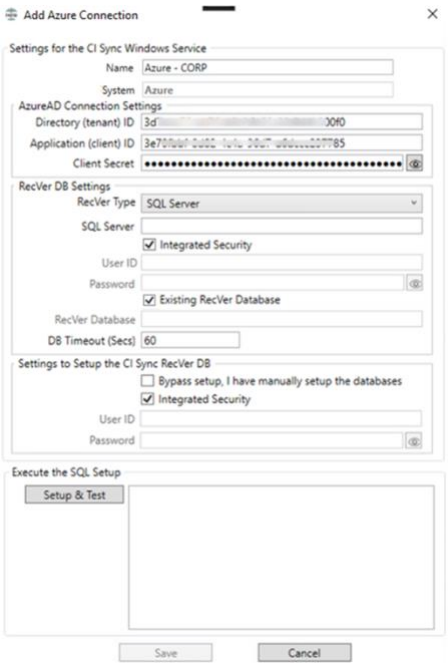
#	DBA Tasks (in SQL Server)	Additional Notes
1	Either **register a Login** (which represents the Windows User Account used by the CI Sync (EE) Agent Windows Service) Or **create a SQL Login** (for the CI Sync (EE) Agent to use).	If you are using **Windows "Integrated Security"** between the CI Sync (EE) Agent and the SQL Server, **then the CI Sync (EE) Agent Windows Service account needs to be registered within SQL Server as a Login**. *Note: The CI Sync (EE) Agent Windows Service account is the one you created during Task 4a: Create a Windows Service Account for the CI Sync (EE) Agent to use.* **Alternatively**, if you are using **Native SQL authentication** between the CI Sync (EE) Agent and the SQL Server, **then a Login will need to be created**. In this case your DBA will need to provide you with the SQL User ID and password details so you can enter them into the Config Utility UI as explained further below.
2	**Execute the SQL Script** provided by Syncfish located here. The script creates the Syncfish RecVer database. The script is: Cisee-recver-create-script.sql	Notes for the DBA when running the script: • The script has a placeholder called $(database_name) as the replacement for the RecVer database name. • For an Azure Source System Syncfish recommend "**cisee_recver_azure**" as the RecVer database name. • To run the setup script using the sqlcmd utility, the database_name parameter needs to be passed in using the -v switch: *> sqlcmd -I -S localhost -v database_name=" **cisee_recver_azure** " -i "cisee-recver-create-script.sql"* • If running the script in SSMS, the value $(database_name) (including the $ and brackets) needs to be replaced with the name for the recver database.
3	**Map the Login as a user** in the **RecVer** database and **grant the user** the **role_agent** role.	Map the user and grant the role to the RecVer database created by the SQL script above.

Click here to return to steps index

2. Within the CI Sync Agent Configuration Utility, navigate to the **Source Systems** tab, select **Azure** and click the **Add** button.



3. Enter details on the "**Add Azure Connection**" form.

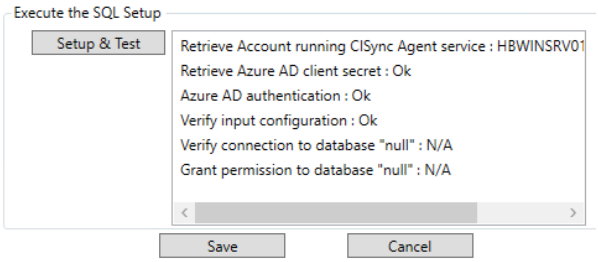


   (a) In the section titled "**Settings for the CI Sync Windows Service**"
   This section defines the settings for the CI Sync (EE) Agent to use as it runs in the background (i.e. after you have exited the Config Utility).
   - **Name**: Enter a name that describes this Source System Connection represents.
   - **Directory (tenant) ID**: Enter the value you captured in **Task F1**.
   - **Application (tenant) ID**: Enter the value you captured in **Task F1**
   - **Client Secret**: Enter the value you captured in **Task F1**.
   - **RecVer Type**: Set this to SQL Server (*in the future CI Sync will support other database types for the RecVer/delta functionality*).
   - **SQL Server**: Enter the SQL Server instance path in the format: server\instance.
   - **Integrated Security**: Check this checkbox if you want the CI Sync (EE) Agent to authenticate to the SQL Server using the Windows User Account you setup for the CI Sync (EE) Agent Windows Service. Otherwise, uncheck this checkbox and enter the **User ID** and **Password** for a **Native SQL Login** provided to you by the DBA.
   - **Existing RecVer Database**: Check this checkbox (because your SQL DBA has already manually created the RecVer database for you).
   - **DB Timeout (Secs)**: Sets both the connection and statement execution timeouts. Syncfish recommend 60 (seconds).

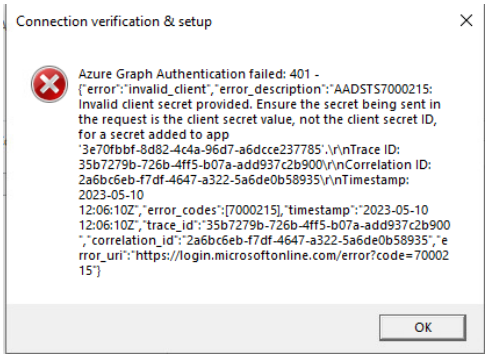
   (b) In the section titled "**Settings to Setup the CI Sync RecVer DB**"
   *This section defines how the **Config Utility** will connect to SQL Server to perform the automated SQL setup and configuration steps. Because your SQL DBA has already manually created the RecVer database you use this section to confirm this is the case.*
   - **Bypass setup, I have manually setup the databases:** Check this checkbox (because your SQL DBA has already manually created the RecVer database and set the relevant permissions for you).

Click here to return to steps index

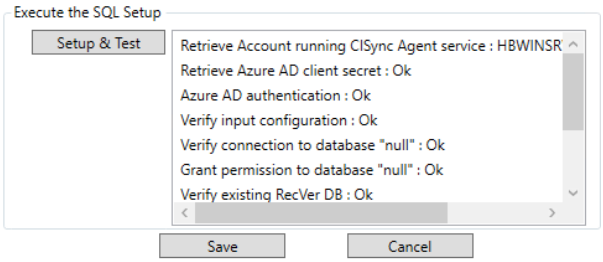
4. In the section titled "**Execute the SQL Setup**", click the **Setup & Test** button. This will create the CI Sync RecVer database (on the provided SQL server), grant the CI Sync (EE) Agent Windows Service Account to the RecVer database and test connectivity to the source system (in this case to Azure) by authentication with the App Registration created earlier.



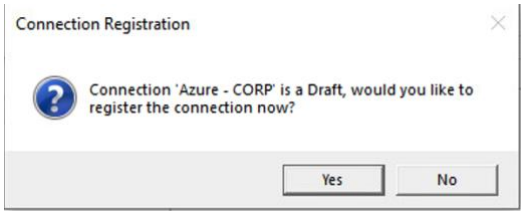
7. If the connection to the source system (in this case via the Microsoft Azure Graph API for Azure) fails, an error will be displayed (see example below). In this case recheck all App Registration details and the various settings entered into the CI Sync (EE) Agent Config Utility on the prior connection setup/settings screen.



8. After a successful test has been completed, click the **Save** button.

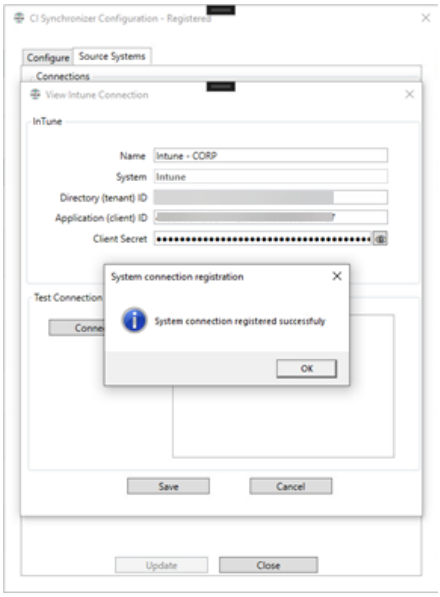


5. Finally, click **Yes** to register this new connection (via your CI Sync (EE) Agent) with your customer specific CI Sync (EE) SaaS instance.

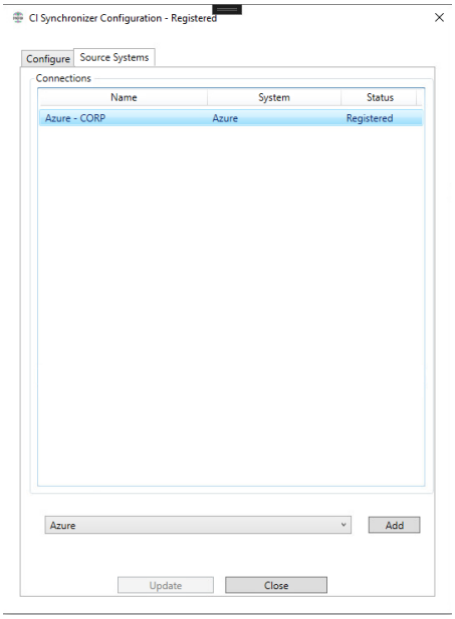


6. Assuming no errors you will see a confirmation as shown below.

Click here to return to steps index



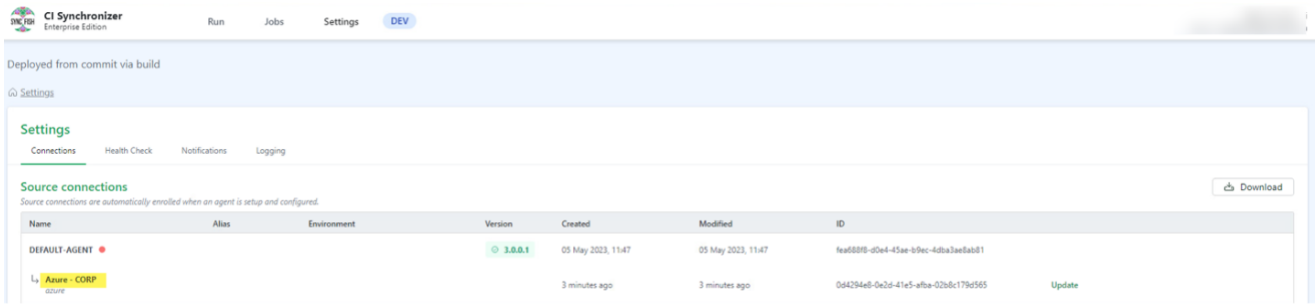
7. Click **OK.**
8. The new connection will appear in the Connections List of the CI Sync (EE) Agent.



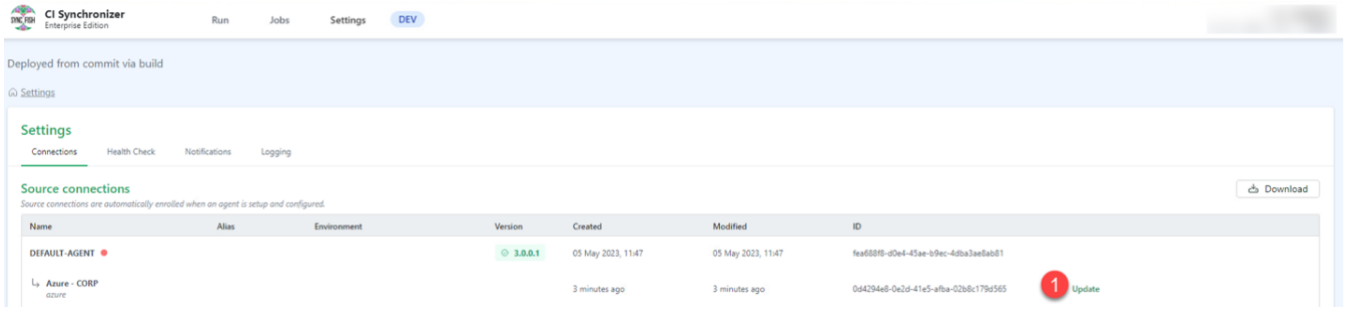
9. You can now proceed to **Task F3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EEE) Agent status and set any remaining connection parameters**.

Task F3: Use the CI Sync (EE) SaaS application User Interface to check the CI Sync (EE) Agent status and set remaining connection parameters.

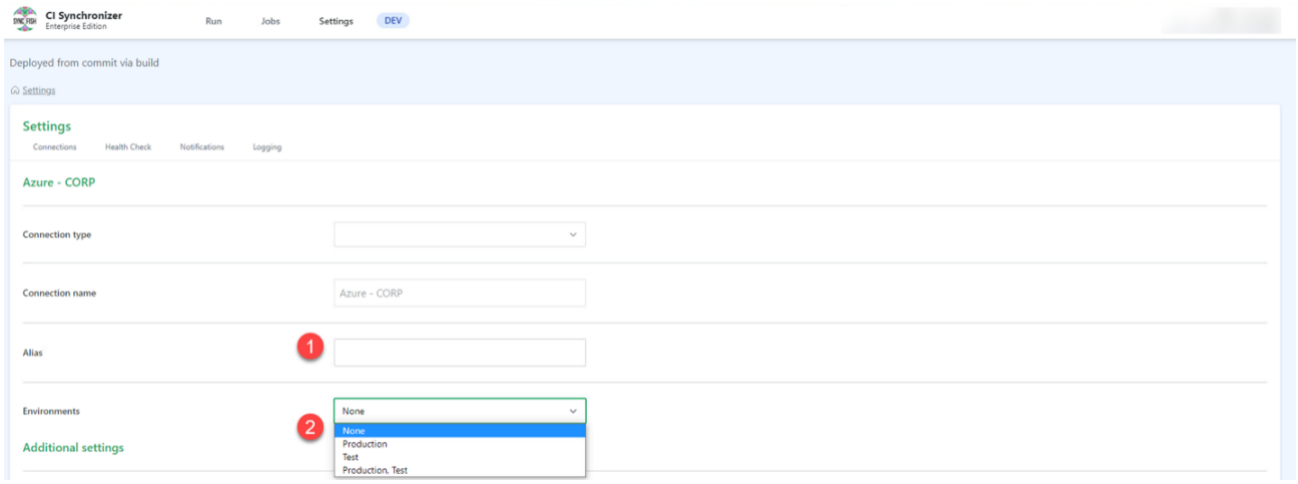
1. Login to your CI Sync (EE) SaaS instance at https://***YourCo***.syncfish.app
2. In the CI Sync UI, navigate to **Settings > Connections**. The new Source System Connection should appear in the Source connections list



3. Click the **Update** hyperlink



4. Use the settings form to as follows:
   - Enter an **Alias** (optional)
   - Select the **Environment**/s the new source connection can be used for. In most cases a Source System Connection is used for both Test and Production sync jobs (as distinct from the Destination Connections which can only be either Test or Production).
   - Check the **consent** checkbox and Click the **Save connection** button.



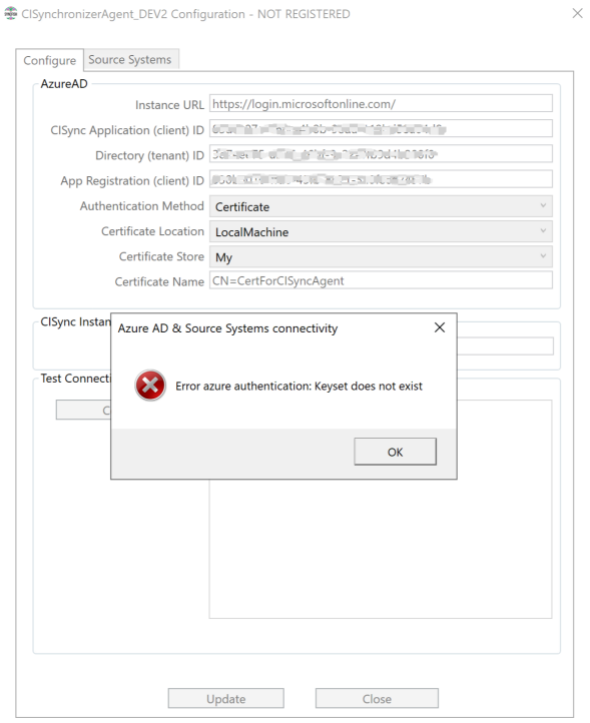
You have now completed all tasks to add Azure as a Source Connection in the CI Sync (EE) Agent. You should now proceed back to Step 6 - Configure your ServiceNow to be ready for CI Sync (EE) (and subsequent steps).

Appendix Z – Trouble Shooting Topics

- Trouble Shooting Topic Z1: Keyset does not exist error using CI Sync Agent "Test Connection" to validate Azure authentication with certificate based authentication.
- Trouble Shooting Topic Z2: Certificate not found error using CI Sync Agent "Test Connection" to validate Azure authentication with certificate based authentication.

Trouble Shooting Topic Z1: Keyset does not exist error using CI Sync Agent "Test Connection" to validate Azure authentication with certificate based authentication.

Error "Keyset does not exist".



Overview and Cause

This error can occur when you use the CI Sync (EE) Agent Configuration utility to test a connection which is using a new Digital Certificate (i.e. one that hasn't been previously validated).

This can happen for two main reasons:

1. **Cause 1 -** The Digital Certificate was imported into the Current User Certificate Store rather than the Local Machine Certificate Store.
2. **Cause 2 -** The CI Sync (EE) Agent user account (i.e. the user account used by the CI Sync (EE) Agent Windows Service) does not have sufficient permission to access the certificate store.

Click here to return to steps index

Fix 1

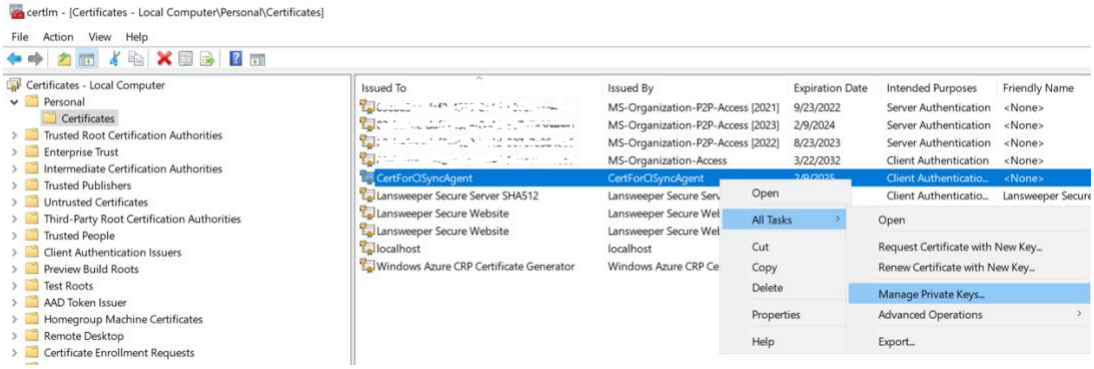
Reimport the Digital Certificate into the Local Machine Certificate Store (and delete it from the Current User Certificate Store).

Fix 2

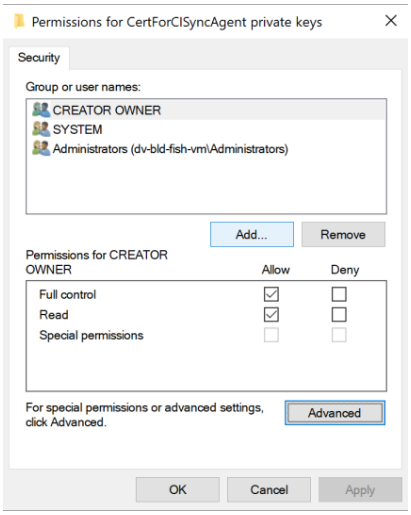
a. Open Microsoft Management Console

b. Navigate to *Certificates - Local Computer → Personal → Certificates*

c. Right mouse click on the installed certificate and select *All Tasks → Manage Private Keys*

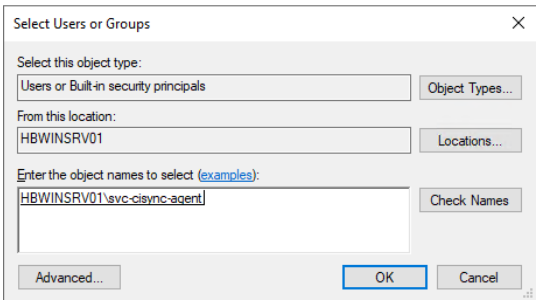


d. On the permissions window, click on the Add button

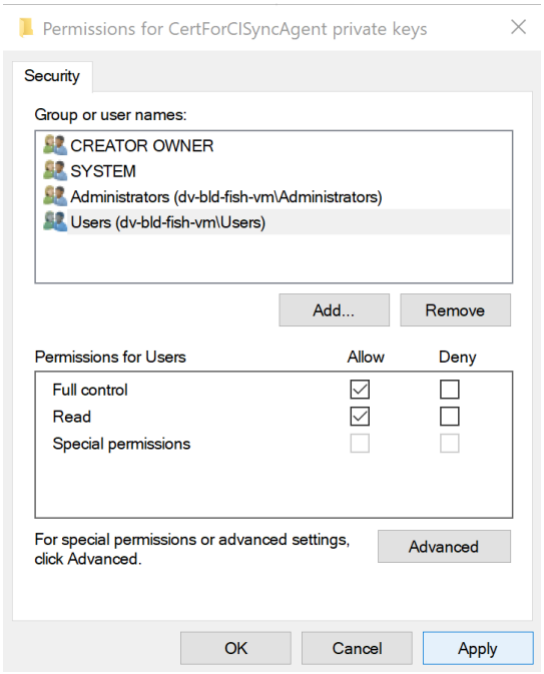


Click here to return to steps index

e. Search for Users, locate the CI Sync (EE) Agent User *(e.g. svc-cisync-agent)*, press Check Names, then press the OK button.



f. Back on the permissions window, make sure Full control and Read are ticked. Then click Apply and then Ok.

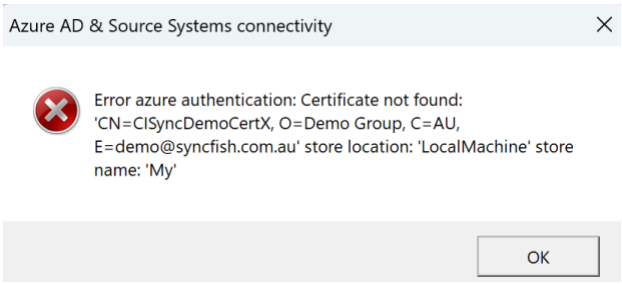


g. Back on the Agent Configuration utility, test the connection which should now be authenticating correctly.

Click here to return to steps index

Trouble Shooting Topic Z2: Certificate not found error using CI Sync Agent "Test Connection" to validate Azure authentication with certificate based authentication.

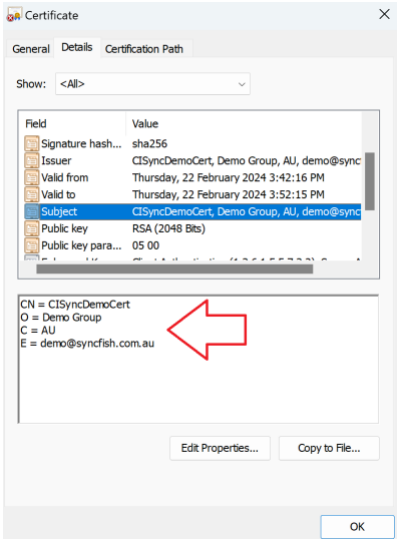
Error "Azure Authentication: Certificate not found".



Overview and Cause

This error can occur if the Certificate Subject (sometimes referred to as the Certificate Name) contains carriage returns when stored in the Windows Certificate Store.

The Subject name of certificates typically include multiple values that compose the distinguished name information, such as common name, country, organization, etc. When certificates have multiple values, the certificate *Subject* will typically include carriage returns when displayed in the Windows Certificate Store (see below).



If the Subject name in the Windows Certificate Store includes carriage returns it impacts the way the CI Sync (EE) Config Utility is able to search and find the certificate using the Certificate Name value you enter into the CI Sync (EE) Config Utility.

In these circumstances you will need to construct the certificate name in a particular manner when you enter it into the CI Sync (EE) Config Utility (as described in the Fix section below).

Click here to return to steps index

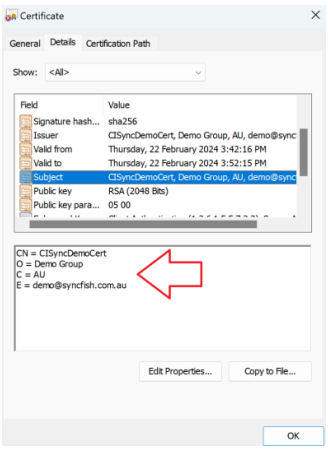
Fix

The individual values of the Subject (as they appear in the Windows Certificate Store) need to be converted into comma separated values **and also** have the blank spaces trimmed before and after the equal sign (as per the example below).

You then need to copy/paste the newly formatted string value into the Certificate Name field in the CI Sync (EE) Config Utility UI form and execute a Test Connection once again.

The example below shows how to construct the correct format:

1. Locate the certificate in the Windows Certificate Store and click the Subject field.



2. Copy the content of the *Subject* field into text editor. It will appear (in the text editor) including carriage returns as shown below.

   CN = CISyncDemoCert
   O = Demo Group
   C = AU
   E = demo@syncfish.com.au

3. Edit the *Subject* string and change the format as shown below:

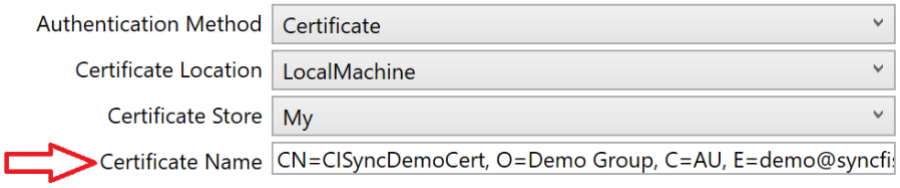
   CN=CISyncDemoCert, O=Demo Group, C=AU, E=demo@syncfish.com.au

   Important Notes:

   - The blank spaces must be removed before and after each equal sign.
   - A single space must be entered after each comma.
   - Make sure there is no trailing space at the very end of the string.

Click here to return to steps index

4. Return to the CI Sync (EE) Config Utility UI and paste the newly constructed string into the Certificate Name field as shown below. Click the Connect button under the Test Connection section of the UI. If in doubt refer to this section of the setup guide: Task 5b: Use the CI Sync (EE) Agent Config Utility to test the connection values and register the CI Sync (EE) Agent with your CI Sync (EE) SaaS Instance.



End of Document