

EcoStruxure Building Operation

Google Calendar

SmartConnector

Installation & User Guide

March 2023



Life Is On

Schneider
Electric

Life Is On



EcoStruxure Building Operation

Google Calendar

SmartConnector

Installation & User Guide

March 2023

Life Is On



Copyright © 2023 Schneider Electric. All rights reserved.

The Schneider Electric brand and any registered trademarks of Schneider Electric Industries SAS referred to in this guide are the sole property of Schneider Electric SA and its subsidiaries. They may not be used for any purpose without the owner's permission, given in writing. This guide and its content are protected, within the meaning of the French intellectual property code (Code de la propriété intellectuelle français, referred to hereafter as "the Code"), under the laws of copyright covering texts, drawings and models, as well as by trademark law. You agree not to reproduce, other than for your own personal, non-commercial use as defined in the Code, all or part of this guide on any medium whatsoever without Schneider Electric's permission, given in writing. You also agree not to establish any hypertext links to this guide or its content. Schneider Electric does not grant any right or license for the personal and non-commercial use of the guide or its content, except for a non-exclusive license to consult it on an "as is" basis, at your own risk. All other rights are reserved.

Trademarks and registered trademarks are the property of their respective owners.

Life Is On

Schneider
 **Electric**

Contents

Contents

1	Functional Overview.....	6
2	Restrictions & Limitations.....	7
2.1	SmartConnector Service Version	7
2.2	EWS Supported Systems.....	7
2.3	Google API Supported.....	7
3	Installation.....	8
4	Getting Started.....	9
5	Configuration & Settings.....	14
5.1	Processor Configuration.....	14
5.2	Adding the Custom Assembly to the Service.....	15
6	Revision History	21
7	References.....	22

1 Functional Overview

The Google Calendar interface serves as a middleware application that facilitates communication between the Google Calendar API service and EcoStruxure Building Operation through EcoStruxure Web Services (EWS). The interface functions by retrieving information about events from a user's Google Calendar through the Google Calendar API and storing it in EcoStruxure Web Services (EWS).

One of the key outcomes of this integration is the creation of an EWS Server that displays values based on a single Gmail calendar. The EWS Server provides real-time updates on ongoing events, occupancy as well as the next schedule event.

This integration allows EcoStruxure Building Operation to access and display event information from a user's Google Calendar, enabling efficient scheduling and coordination of building operations. Additionally, this integration streamlines the communication between the two platforms, minimizing the need for manual data transfer and reducing the likelihood of errors.

The processor leverages the SmartConnector Service framework and details of the application (release history, installation notes etc.) are available separately and are not covered in this manual.

The SmartConnector application is licensed on a single server basis but may also be configured to connect to both Automation Server (AS-P) devices as well as Enterprise Servers (ES).

2 Restrictions & Limitations

2.1 SmartConnector Service Version

The processors have been configured to operate with the SmartConnector version 2.5.4.18, use with any other version of the SmartConnector framework is not supported.

2.2 EWS Supported Systems

The processors can support EcoStruxure systems operating with the EcoStruxure Web Services (EWS) protocol v1.1 and v1.2.

2.3 Google API Supported

The SmartConnector has been tested and validated against Google API version 1.57.0.

Other versions may cause issues and are not supported.

3 Installation

Please refer to the SmartConnector Installation and Configuration Guide.pdf for guidance on SmartConnector installation.

In the release folder, you will find 2 folders called Google Calendar and Credential Generator. Both of these folders will be holding libraries that will have to be deployed separately. To deploy the Google Calendar SmartConnector assembly, copy the following files in the Google Calendar folder:

Google.Apis.Auth.dll
Google.Apis.Auth.PlatformServices.dll
Google.Apis.Calendar.v3.dll
Google.Apis.Core.dll
Google.Apis.dll
Google.Apis.PlatformServices.dll
ISC.GoogleCalendar.dll
NCrontab.Signed.dll
NLog.Targets.Syslog.dll
NodaTime.dll
Polly.Contrib.WaitAndRetry.dll
System.Runtime.CompilerServices.Unsafe.dll

into the service installation directory. Normally “C:\Program Files (x86)\Schneider Electric\SmartConnector”

Copy the following files from the Credential Generator folder and paste into a location of your own choosing. DO NOT paste them into the SmartConnector directory.

Credential Generator.exe
Google.Apis.Auth.dll
Google.Apis.Auth.PlatformServices.dll
Google.Apis.Calendar.v3.dll
Google.Apis.Core.dll
Google.Apis.dll
Google.Apis.PlatformServices.dll
Newtonsoft.Json.dll

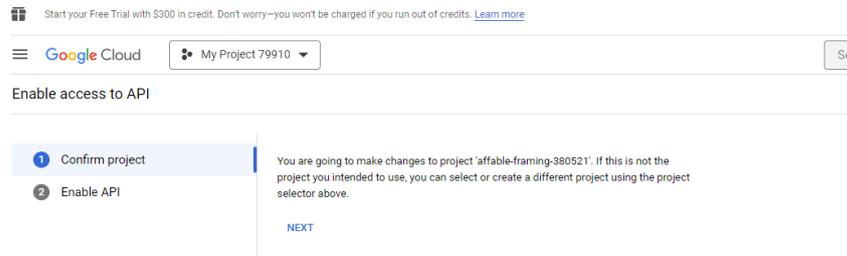
In this directory, create another directory called **.credentials**. In the ‘.credentials’ directory create another directory called **google-calendar.json**. This directory structure is needed when the Credential Generator is executed.

4

Getting Started

To install the Google Calendar integration, you will need to first create a Gmail account to use. Once you have created and are logged in to your Google account you will need to do the following:

- Open the browser and enter the url: <https://console.developers.google.com/start/api?id=calendar> to create or select a project in the Google Developers Console and automatically turn on the API. Click Continue, then Go to credentials.
- In the screen below click on Confirm Project and then Enable API. If there is no project selected, then click on the drop and select one from the list. If there are none available, then click on the Create Project and just give it a meaningful name and then click on the Create button.



- Navigate to the OAuth Consent Screen. Use the screenshot below to help you.
- Select the appropriate user type and click on the Create button

Google Cloud Google Calendar API

API APIs & Services OAuth consent screen

Enabled APIs & services
Library
Credentials
OAuth consent screen
Page usage agreements

Choose how you want to configure and register your app, including your target users. You can only associate one app with your project.

User Type

Internal ⓘ

Only available to users within your organization. You will not need to submit your app for verification. [Learn more about user type](#)

External ⓘ

Available to any test user with a Google Account. Your app will start in testing mode and will only be available to users you add to the list of test users. Once your app is ready to push to production, you may need to verify your app. [Learn more about user type](#)

CREATE

[Let us know what you think](#) about our OAuth experience

- Fill in the mandatory fields as highlighted in the screen below (next page). Click on Save and Continue button once the form is filled.

Google Cloud Google Calendar API

API APIs & Services

- Enabled APIs & services
- Library
- Credentials
- OAuth consent screen**
- Page usage agreements

Edit app registration

App information

This shows in the consent screen, and helps end users know who you are and contact you

App name *

The name of the app asking for consent

User support email *

For users to contact you with questions about their consent

App logo

This is your logo. It helps people recognize your app and is displayed on the OAuth consent screen.

After you upload a logo, you will need to submit your app for verification unless the app is configured for internal use only or has a publishing status of "Testing". [Learn more](#)

Logo file to upload

Upload an image, not larger than 1MB on the consent screen that will help users recognize your app. Allowed image formats are JPG, PNG, and BMP. Logos should be square and 120px by 120px for the best results.

App domain

To protect you and your users, Google only allows apps using OAuth to use Authorized Domains. The following information will be shown to your users on the consent screen.

Application home page

Provide users a link to your home page

Application privacy policy link

Provide users a link to your public privacy policy

Application terms of service link

Provide users a link to your public terms of service

Authorized domains

When a domain is used on the consent screen or in an OAuth client's configuration, it must be pre-registered here. If your app needs to go through verification, please go to the [Google Search Console](#) to check if your domains are authorized. [Learn more](#) about the authorized domain limit.

Developer contact information

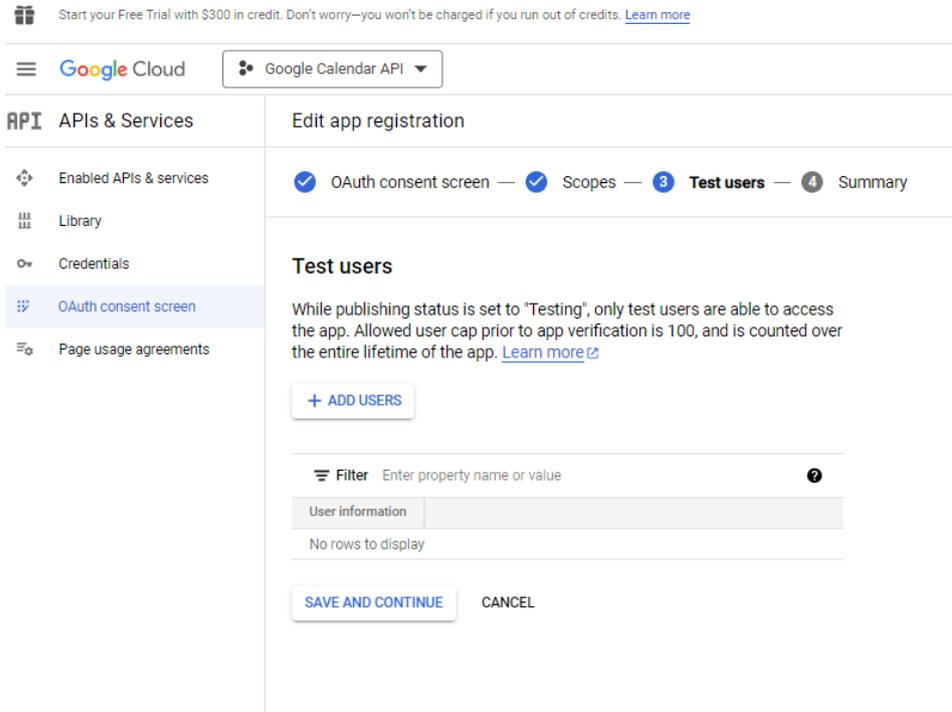
Email addresses *

These email addresses are for Google to notify you about any changes to your project.

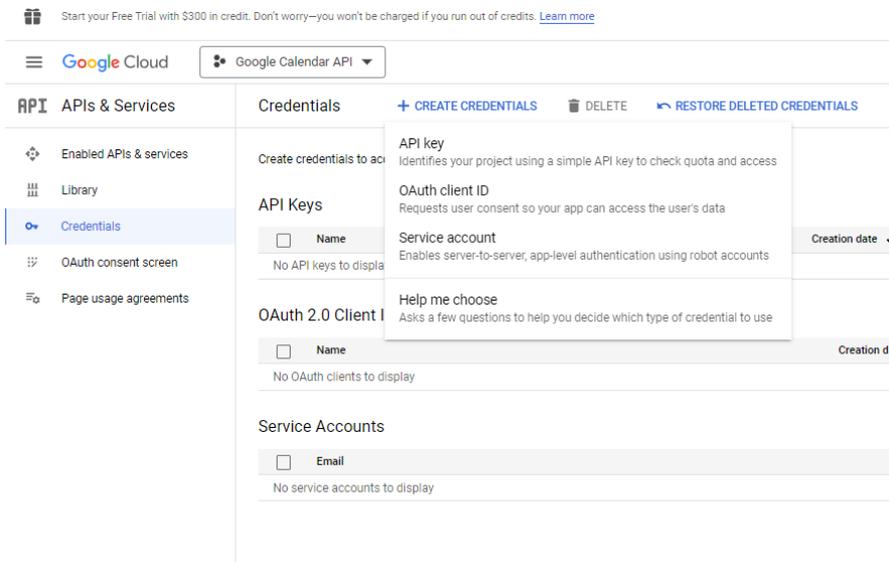
<

- Click on Save and Continue button on the Scopes screen to continue forwards in the process.

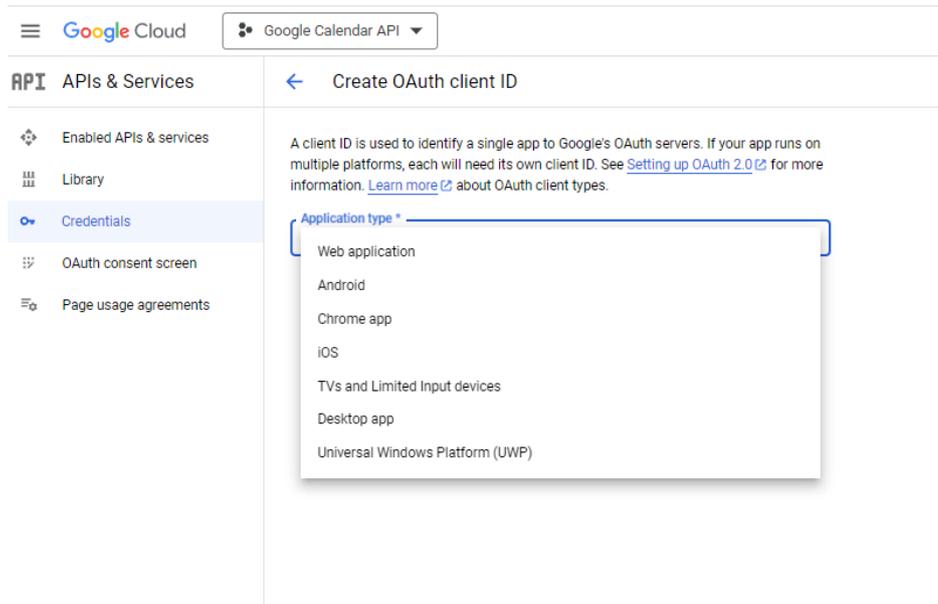
- In the Test Users screen, add a test user by clicking on ADD Users button which will slide out a panel where a user's Google Email can be entered. By entering a google account, you are allowing the smartconnector to access that google accounts calendar. Click on Save and Continue once the google account appears in the grid.



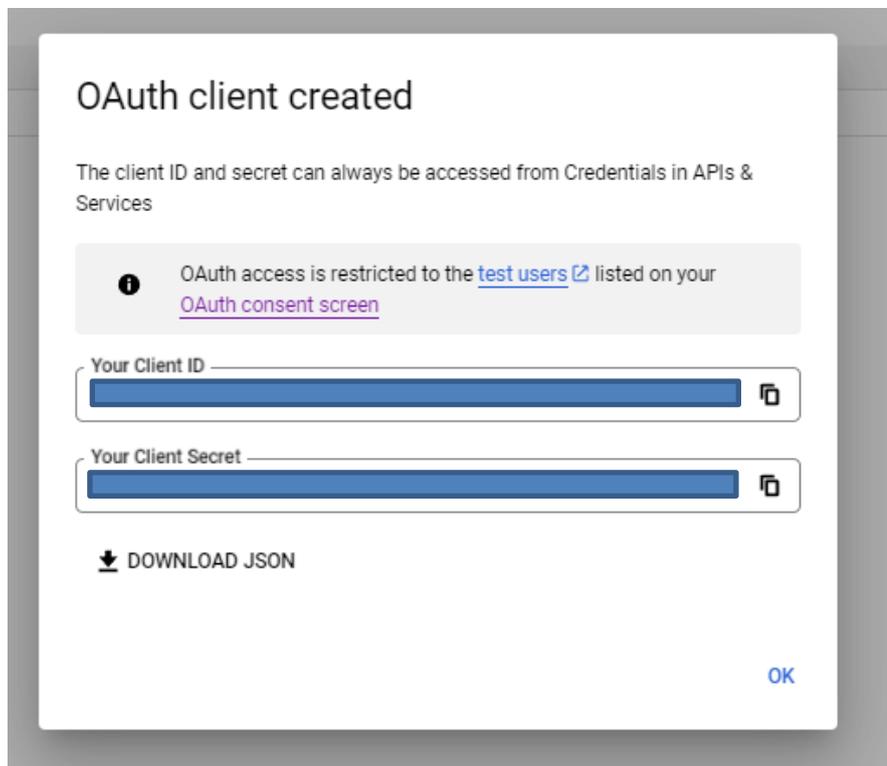
- Select the Credentials item in the menu and then click on the Create credentials tab at the top of the page and select OAuth client ID.



- In the Application Type drop down, select Desktop App and give it a name. Click on the Create button.



- A popup will appear to confirm that the OAuth client being created successfully. In the popup at the bottom left, click on the Download Json link.



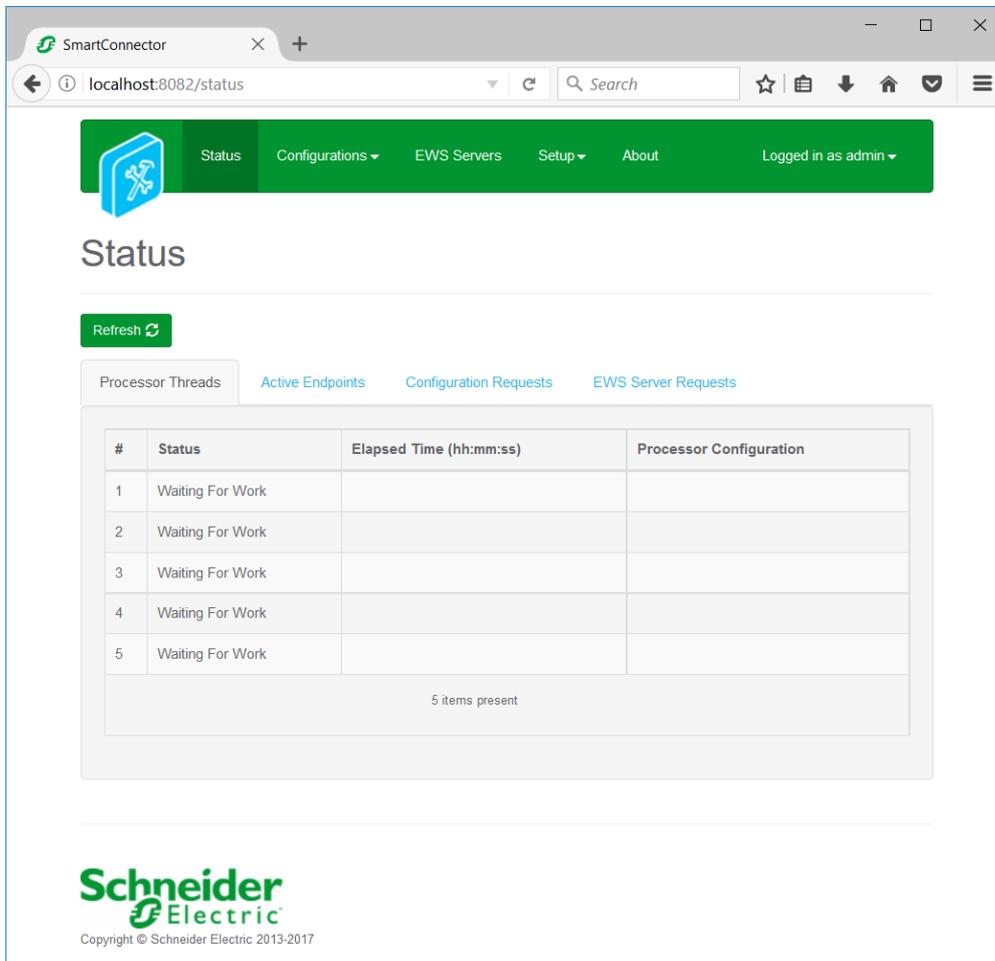
- Rename the downloaded file to **client_secret.json** and move it to the location where the Credential Generator.exe is.

5 Configuration & Settings

5.1 Processor Configuration

With a default installation of SmartConnector, the configuration pages for the server can be reached at the following address on the server the service has been installed on:

<http://localhost:8082/>



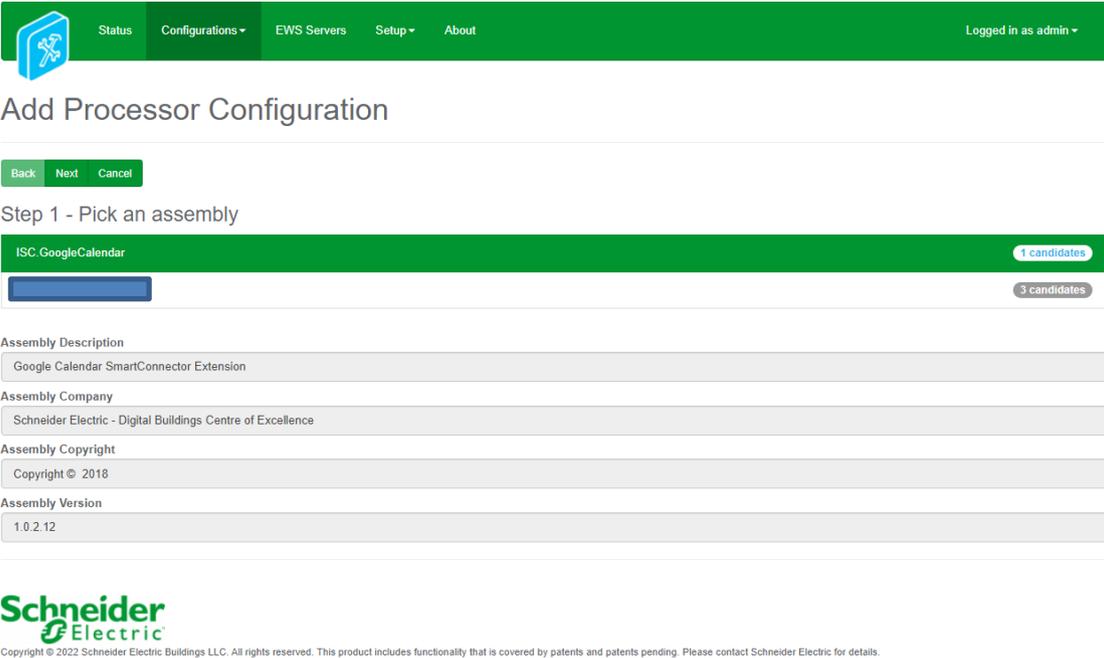
5.2

Adding the Custom Assembly to the Service

Switch to the Configurations tab and select Processor and click on Add New +



At the Add Configuration window, Step 1 – Pick an assembly, select the reference to “ISC.GoogleCalendar” (this will be highlighted green when selected)



The screenshot shows the 'Add Processor Configuration' window. At the top, there is a navigation bar with 'Configurations' selected. Below the navigation bar, there are buttons for 'Back', 'Next', and 'Cancel'. The main heading is 'Step 1 - Pick an assembly'. A search bar is present, and a list of assemblies is shown. The assembly 'ISC.GoogleCalendar' is highlighted in green, and it has '1 candidates' listed next to it. Below the list, there are several fields for assembly details: 'Assembly Description' (Google Calendar SmartConnector Extension), 'Assembly Company' (Schneider Electric - Digital Buildings Centre of Excellence), 'Assembly Copyright' (Copyright © 2018), and 'Assembly Version' (1.0.2.12). At the bottom, there is the Schneider Electric logo and a copyright notice.

Select Next and proceed to Step 2 Choose a Class

Ensure the class “ISC.GoogleCalendar.CalendarEventsProcessor” is selected first

Step 2 - Choose a Class



Select Next and proceed to Step 3 Name Configuration

Enter a meaningful name and description for the Processor which will enable you to identify this process in the configuration window later.

Select Finish and proceed to the Configuration screen.

The screenshot shows a web application interface for configuring a processor. At the top is a green navigation bar with a logo on the left and menu items: Status, Configurations (selected), EWS Servers, Setup, and About. On the right of the bar, it says "Logged in as admin". Below the navigation bar is the title "Processor Configuration". Underneath are several action buttons: "Edit All", "Start", "Validate", "Reset Counter", and "Reset Timer". The main configuration area has two input fields: "Name" with the value "Google Cal New" and "Is Active" with the value "True". Below these is a "Description" field. A tabbed interface shows "Processor" as the active tab, with other tabs for "Details", "Control", "History", and "Schedule". The "Details" tab is expanded, showing several configuration fields: "Class Name" (ISC.GoogleCalendar.CalendarEventsProcessor), "Assembly File" (ISC.GoogleCalendar.dll), "Assembly Description" (Google Calendar SmartConnector Extension), "Assembly Company" (Schneider Electric - Digital Buildings Centre of Excellence), "Assembly Copyright" (Copyright © 2018), and "Assembly Version" (1.0.2.12).

In the configuration window select the Details Tab, you will then be presented with the screen to enter the configuration information. Much of the configuration has default options however they should be checked and validated for the installation. Edit the applicable fields as follows.

Calendar Lead Days

Number of days from where the Smart Connector should start looking for events in the user's calendar.

Max Records

The maximum number of events that will be retrieved from the user's calendar.

Use Reminder Time As Prebook

If set to true, then the notification time will be used that is set by the user in their calendar otherwise it will be set to the default value which is usually 30 minutes.

User Name

This property represents the User Name that will be needed as part of the credentials to create an instance of an EWS server.

Password

As above, this is the password related to the user credentials.

Server Address

This property should be set to the full address required to access the EWS Server being hosted. This is normally in the case of a StruxureWare ES or AS device as follows:

[http://<IPADDRESS>:<PORT\(8093\)>/EcoStruxure/DataExchange](http://<IPADDRESS>:<PORT(8093)>/EcoStruxure/DataExchange)

Server Name

The default name should be 'Google Calendar'

Secret File

The location of the **client_secret.json** file which was downloaded in the Getting Started section for e.g. c:\temp\credential generator

Token File

Follow the following steps to generate a token response file:

- Go to the directory where the Credential Generator.exe. is located
- To retrieve events from your Google Calendar, you need to follow these steps: First, open the Credential Generator.exe file by double-clicking on it. This will launch your web browser and automatically direct you to the accounts.google.com page. Once you're there, choose the Google account that you want Smart Connector to have access to so that it can process events from that accounts calendar.
- Click on Continue if you receive the 'Google hasn't verified this app' message
- Click on Continue on the next screen to give access to the SmartConnector which will then confirm that a verification code has been received. If you now open up the folder **.credentials\google-calendar.json** in the same directory, you will find a token response file. Enter the file's location in this property but DO NOT append the actual file name at the end for e.g.
c:\temp\credentials\google-calendar.json

The Save Button allows the process configuration to be saved to the database.



A complete configuration will appear as follows:

Processor Configuration

[Edit All](#) [Start](#) [Validate](#) [Reset Counter](#) [Reset Timer](#)

Name: Is Active:

Description:

[Processor](#) [Details](#) [Control](#) [History](#) [Schedule](#)

[Expand All](#) [Collapse All](#)

Details

- Calendar Lead Days ***
- Max Records ***
- Use Reminder Time As Prebook ***
- User Name ***
- Password ***
- Server Address ***
- Server Name ***
- Secret File ***
- Token File ***

← → Server 1 ▶ Google Calendar Server ▶ Google Calendar ▶ Next Event ▶

System Tree

- Server 1
 - System
 - Servers
 - Google Calendar Server
 - Google Calendar
 - Event Summary
 - Next Event**
 - Occupancy Object

Next Event x

Basic Semantics References

General Information

Status Information

EWS Id	Next Event
EWS Type	string
Value	Fishing - Starting at: 12/12/2022 12:00:00 PM
EWS Writeable	ReadOnly
EWS State	Good

6 Revision History

Google Calendar Assembly files required:

Version	Assembly File Details	Date
1.0.2.12	ISC.GoogleCalendar.dll	20/03/2023
1.57.0.0	Google.Apis.Auth.dll	28/03/2022
1.57.0.0	Google.Apis.Auth.PlatformServices.dll	28/03/2022
1.57.0.2759	Google.Apis.Calendar.v3.dll	28/07/2022
1.57.0.0	Google.Apis.Core.dll	28/03/2022
1.57.0.0	Google.Apis.dll	28/03/2022
1.57.0.0	Google.Apis.PlatformServices.dll	28/03/2022
3.3.2.0	NCrontab.Signed.dll	17/02/2019
6.0.3.0	NLog.Targets.Syslog.dll	07/11/2021
3.1.5.0	NodaTime.dll	29/10/2022
1.1.1.0	Polly.Contrib.WaitAndRetry.dll	02/05/2020
6.0.21.52210	System.Runtime.CompilerServices.Unsafe.dll	23/10/2021

Credential Generator Assembly files required:

Version	Assembly File Details	Date
1.0.2.12	Credential Generator.exe	20/03/2023
1.57.0.0	Google.Apis.Auth.dll	28/03/2022
1.57.0.0	Google.Apis.Auth.PlatformServices.dll	28/03/2022
1.57.0.2759	Google.Apis.Calendar.v3.dll	28/07/2022
1.57.0.0	Google.Apis.Core.dll	28/03/2022
1.57.0.0	Google.Apis.dll	28/03/2022
1.57.0.0	Google.Apis.PlatformServices.dll	28/03/2022
13.0.2.27524	Newtonsoft.Json.dll	24/11/2022

7 References

SmartConnector Installation and Configuration Guide.pdf
(TDS-M-INSTALLCONFIG-US.BU.N.EN.12.2017.2.30.CC)

SmartConnector Version 2.2 Release Notes.pdf
(TDS-M-RELEASENOTES-US.BU.N.EN.12.2017.2.30.CC)

Schneider Electric

www.schneider-electric.com/buildings

© 2023 Schneider Electric. All rights reserved.

March 2023