CAD/RMS

Deployment Guide

© Safe Fleet | APRIL 2019 | All rights reserved | Part #: 700-1171 R1

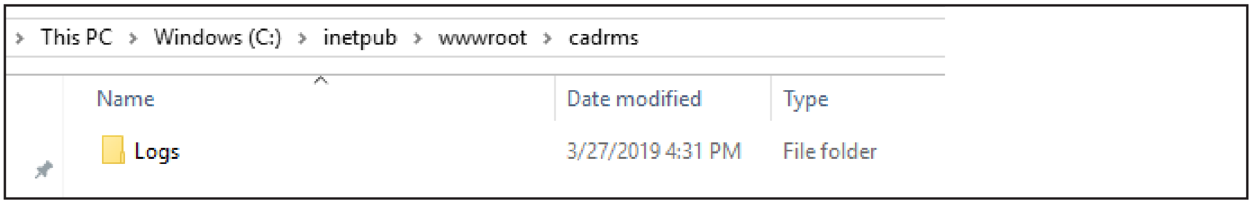
Prerequisites:

- Check that IIS is installed in the server. Click ***here***.
- ***Deploy Single Sign On in the server***.
- Deploy ***Task Manager***.
- V1 of CAD/RMS API is built to run under the DVMS SQL Server Database.
- Dowload the Release Zip that is going to be installed in the server. Installation packages are found ***here***.

Deploying the CAD/RMS API

1. On the hosting system, create a folder named **cadrms** under **C:\inetpub\wwwroot\cadrms** to contain the app's published folders and files.

2. Within the new folder created in Step 1, create a subfolder named **logs** to hold ASP.NET Core Module stdout logs when stdout logging is enabled. If the app is deployed with a **logs** folder in the payload, skip this step.



› This PC › Windows (C:) › inetpub › wwwroot › cadrms

Name	Date modified	Type
Logs	3/27/2019 4:31 PM	File folder

3. Generate a self-signed certificate for the site or use an existing certificate.

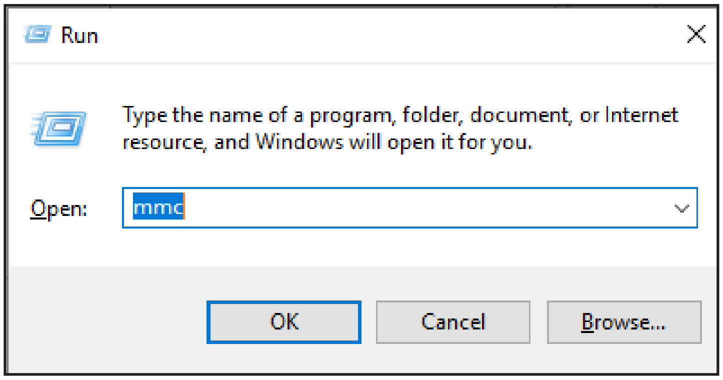
   Using Powershell is an option for generating this certficiate, as shown in the following example.

Using Powershell to Create a Self-Signed Certificate

```
New-SelfSignedCertificate -certstorelocation cert:\localmachine\my -dnsname "test.
domain.com", "localhost"
```

To use the certificate, you must copy and paste it to your **Trusted Certificates**.

a. Open the **Run** command window and enter `mmc`. Click **OK** to open the **Microsoft Management Console**.



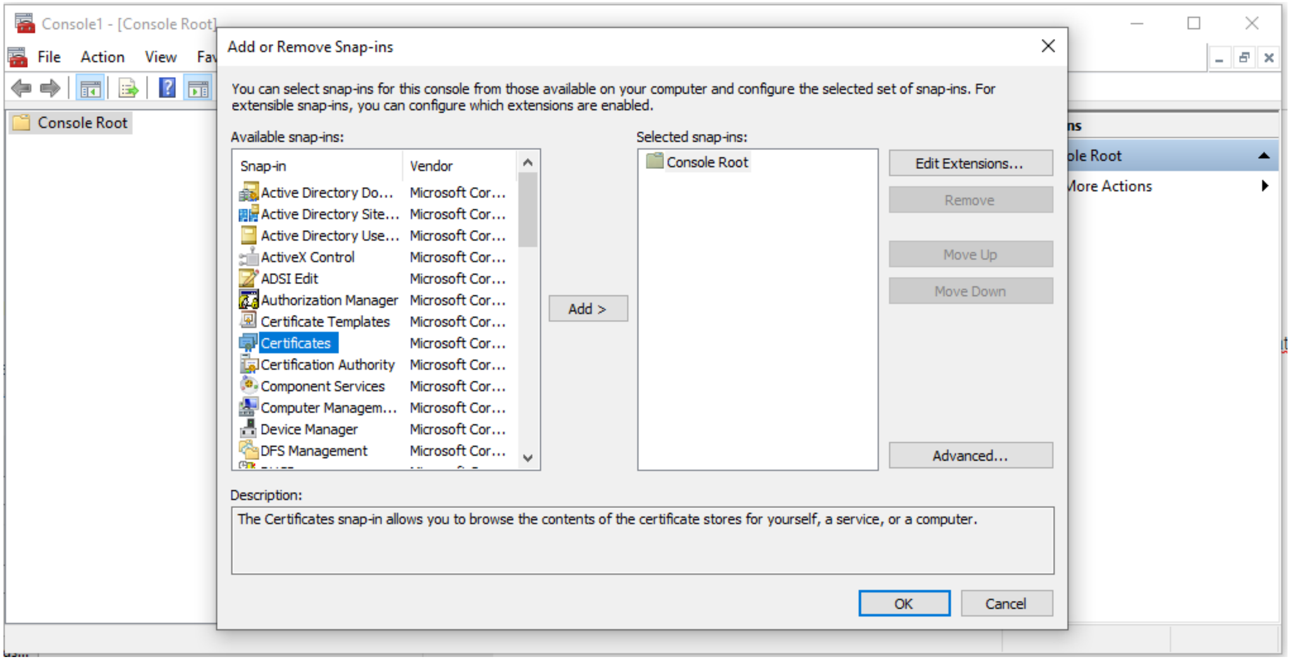
Run

Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: mmc

© Safe Fleet | APRIL 2019 | All rights reserved | Part #: 700-1171 R1

b. Click **File** > **Add/Remove Snap In** to open the **Add or Remove Snap-ins** window.



You can select snap-ins for this console from those available on your computer and configure the selected set of snap-ins. For extensible snap-ins, you can configure which extensions are enabled.

Available snap-ins:

Snap-in	Vendor
Active Directory Do...	Microsoft Cor...
Active Directory Site...	Microsoft Cor...
Active Directory Use...	Microsoft Cor...
ActiveX Control	Microsoft Cor...
ADSI Edit	Microsoft Cor...
Authorization Manager	Microsoft Cor...
Certificate Templates	Microsoft Cor...
Certificates	Microsoft Cor...
Certification Authority	Microsoft Cor...
Component Services	Microsoft Cor...
Computer Managem...	Microsoft Cor...
Device Manager	Microsoft Cor...
DFS Management	Microsoft Cor...

Selected snap-ins:

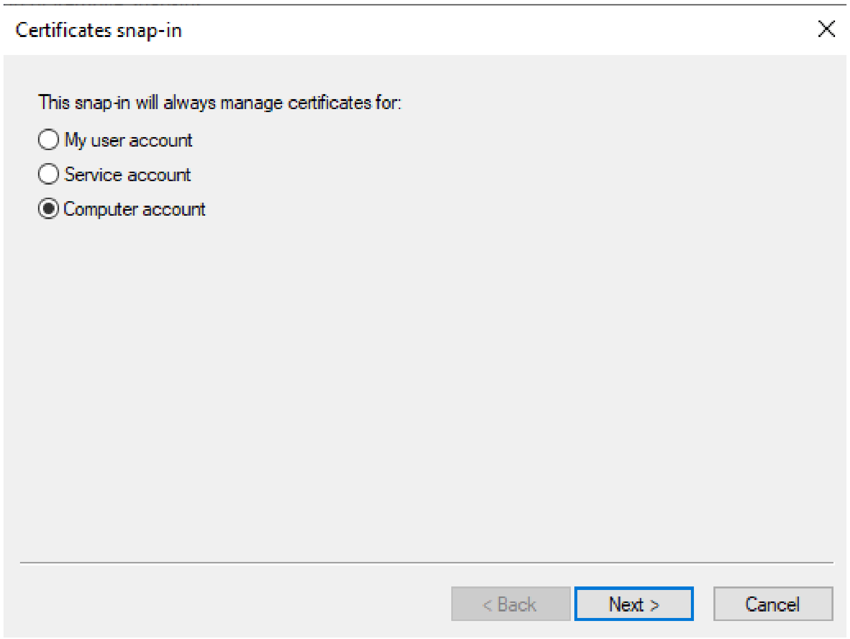
Console Root

Description:

The Certificates snap-in allows you to browse the contents of the certificate stores for yourself, a service, or a computer.

c. Select **Certificates** under **Available snap-ins**, then click **Add** to add the certificate to the **Selected snap-ins** list.

d. Select **Computer account** > **Local Computer** > **Finish**, and then **OK**.



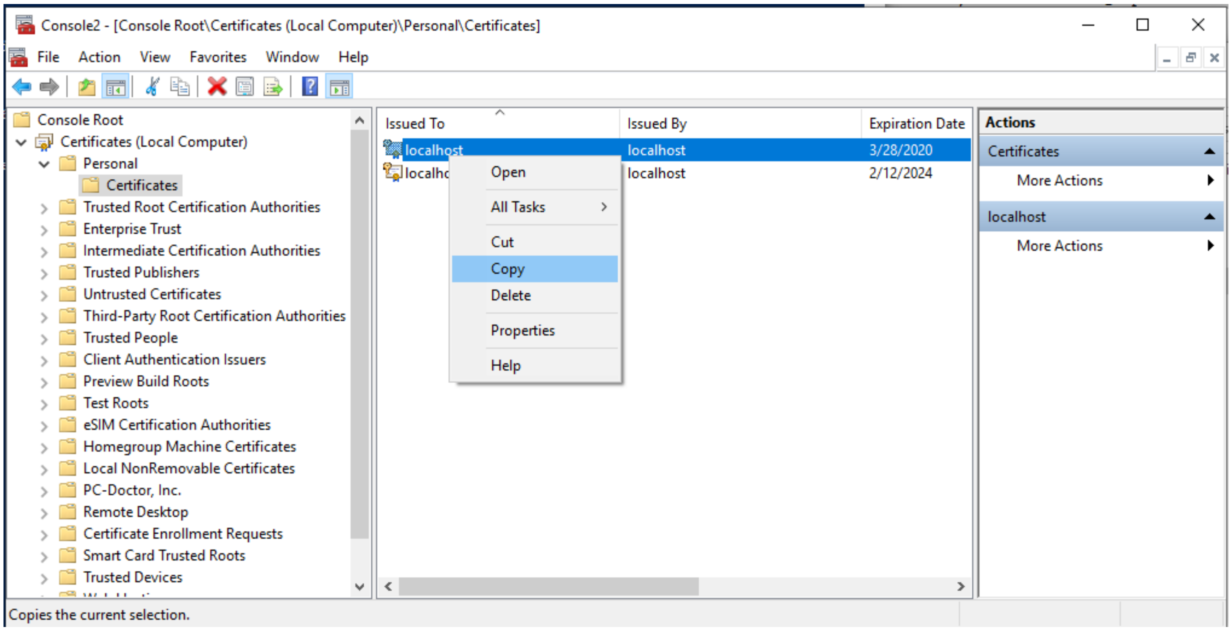
This snap-in will always manage certificates for:

○ My user account

○ Service account

● Computer account

e. Expand **Certificates** on the left panel, then expand **Personal**. Click to select **Certificates,** and then right-click on the certificate that you want to copy to, then click **Copy**.


Console2 - [Console Root\Certificates (Local Computer)\Personal\Certificates]
File
Action
View
Favorites
Window
Help
Console Root
Certificates (Local Computer)
Personal
Certificates
Trusted Root Certification Authorities
Enterprise Trust
Intermediate Certification Authorities
Trusted Publishers
Untrusted Certificates
Third-Party Root Certification Authorities
Trusted People
Client Authentication Issuers
Preview Build Roots
Test Roots
eSIM Certification Authorities
Homegroup Machine Certificates
Local NonRemovable Certificates
PC-Doctor, Inc.
Remote Desktop
Certificate Enrollment Requests
Smart Card Trusted Roots
Trusted Devices
Open
All Tasks
Cut
Copy
Delete
Properties
Help
Actions
Certificates
More Actions
localhost
More Actions
Copies the current selection.

Issued To	Issued By	Expiration Date
localhost	localhost	3/28/2020
localho	localhost	2/12/2024

f. Expand **Trusted Root Certification Authorities** under **Console Root > Certificates (Local Computer)**. Click **Certificate**, then press Ctrl + V on your keyboard to paste the certificate. Check that the certificate was copied in the correct panel.

4. In **IIS Manager**, open the server's node in the **Connections** panel.

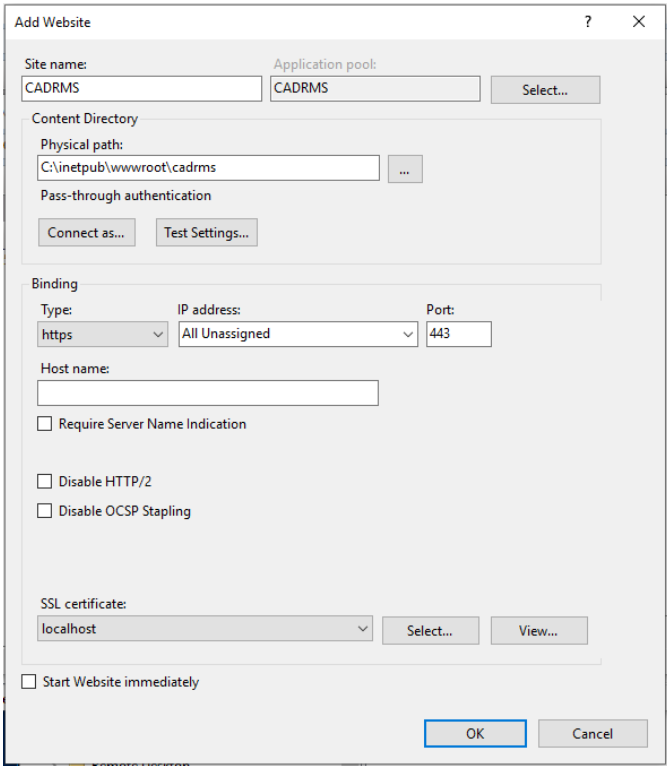
   a. Right-click the **Sites** folder.

   b. Select **Add Website** from the contextual menu.

> **NOTE: Starting the website**
>
> Do ***NOT*** select **Start Website immediately**.

   c. Use a valid certificate and select it in SSL certificate. If you created a valid SSL certificate in *"3. Generate a self-signed certificate for the site or use an existing certificate." on page 2*, select that.

5. Under the server's node, select **Application Pools**.



Site name: CADRMS

Application pool: CADRMS

Physical path: C:\inetpub\wwwroot\cadrms

Type: https

IP address: All Unassigned

Port: 443

Host name:

□ Require Server Name Indication

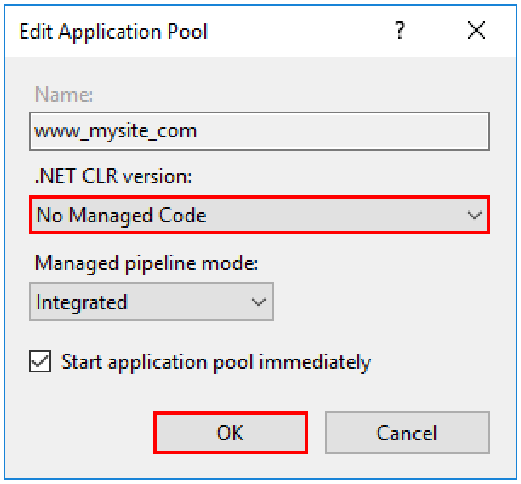
□ Disable HTTP/2

□ Disable OCSP Stapling

SSL certificate: localhost

□ Start Website immediately

6. Right-click the site's **Application Pool**, and select **Basic Settings** from the contextual menu.
7. In the **Edit Application Pool** window, set the **.NET CLR version** to **No Managed Code**:



Name: www_mysite_com

.NET CLR version: No Managed Code

Managed pipeline mode: Integrated

☑ Start application pool immediately

ASP.NET Core

ASP.NET Core runs in a separate process and manages the runtime. ASP.NET Core doesn't rely on loading the desktop CLR. Setting the .**NET CLR version** to **No Managed Code** is optional.

8. For ASP.NET Core 2.2 or later: For a 64-bit (x64) ***self-contained deployment*** that uses the ***in-process hosting*** model, disable the app pool for 32-bit (x86) processes.

   a. In the **Actions** sidebar of IIS Manager's **Application Pools**, select **Set Application Pool Defaults** or **Advanced Settings**.

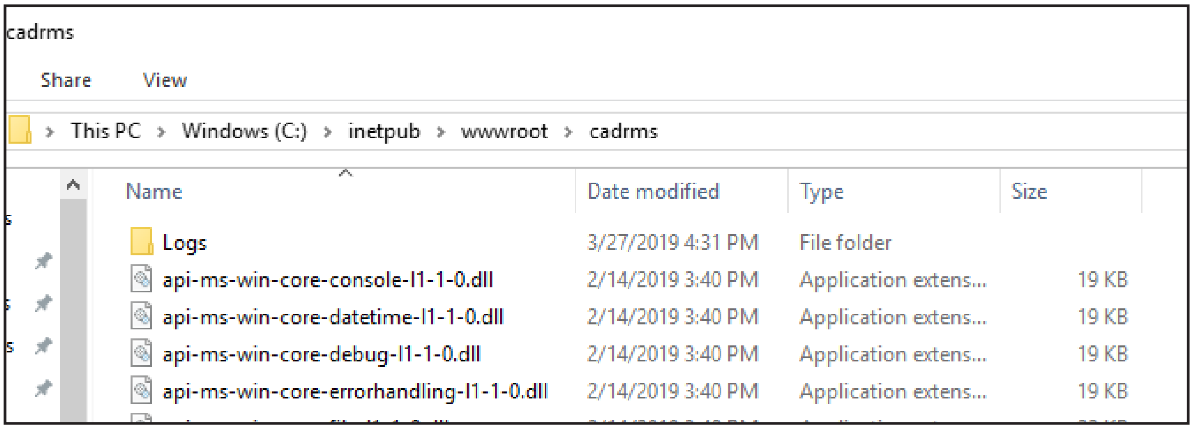
   b. Locate **Enable 32-Bit Applications** and set the value to `False`.

   This setting doesn't affect apps deployed for ***out-of-process hosting***.

9. Confirm that the process model identity has the proper permissions.

   If the default identity of the app pool (**Process Model** > **Identity**) is changed from **ApplicationPoolIdentity** to another identity, verify that the new identity has the required permissions to access the app's folder, database, and other required resources. For example, the app pool requires read and write access to folders where the app reads and writes files.

10. Copy and unzip the release file in the app's folder.



This PC > Windows (C:) > inetpub > wwwroot > cadrms

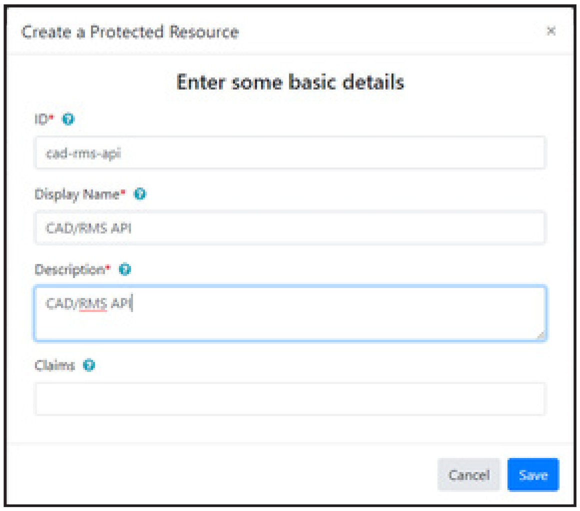
Name	Date modified	Type	Size
Logs	3/27/2019 4:31 PM	File folder	
api-ms-win-core-console-l1-1-0.dll	2/14/2019 3:40 PM	Application extens...	19 KB
api-ms-win-core-datetime-l1-1-0.dll	2/14/2019 3:40 PM	Application extens...	19 KB
api-ms-win-core-debug-l1-1-0.dll	2/14/2019 3:40 PM	Application extens...	19 KB
api-ms-win-core-errorhandling-l1-1-0.dll	2/14/2019 3:40 PM	Application extens...	19 KB

© Safe Fleet | APRIL 2019 | All rights reserved | Part #: 700-1171 R1

11. Register the Application in **SSO**. (Single Sign On)

    a. Log in to **SSO** as an administrator.

    b. Create a new protected resource.



Create a Protected Resource

Enter some basic details

ID*: cad-rms-api

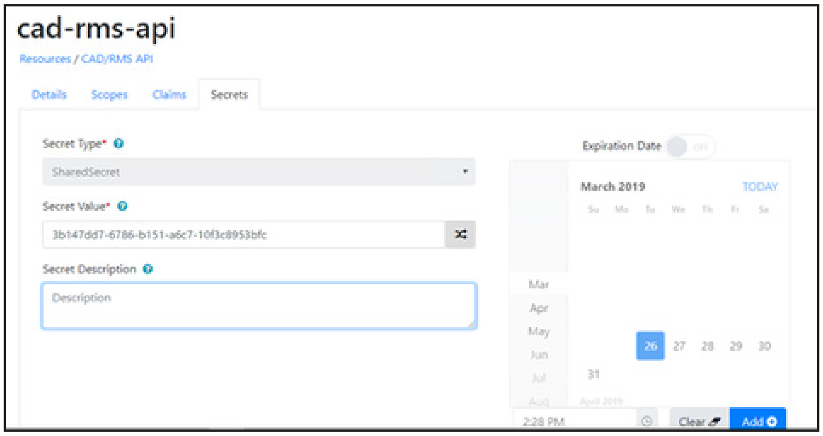
Display Name*: CAD/RMS API

Description*: CAD/RMS API

Claims:

The ID will be used in the configuration file.

    c. Add a secret to this resource.



cad-rms-api

Resources / CAD/RMS API

Secret Type*: SharedSecret

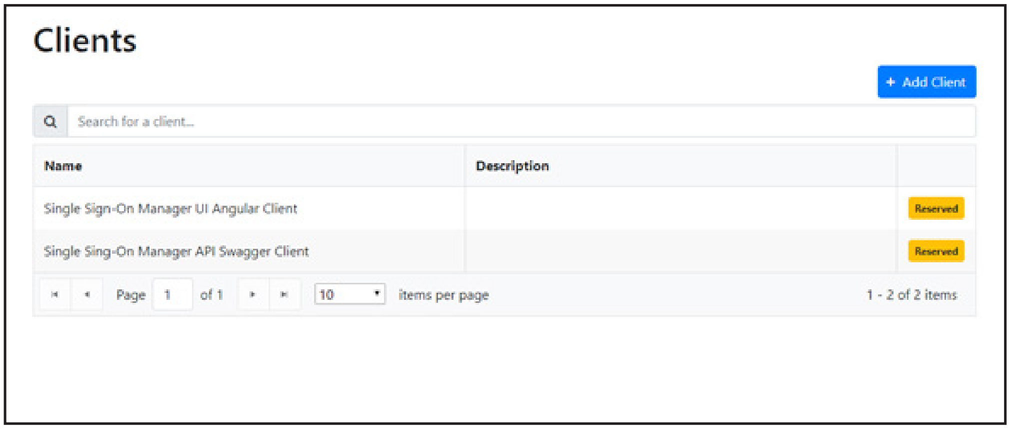
Secret Value*: 3b147dd7-6786-b151-a6c7-10f3c8953bfc

Secret Description: Description

Expiration Date

The secret is a Guide ID that will be used in the config file.

    d. Add a new Client by clicking in **+ Add Client.**

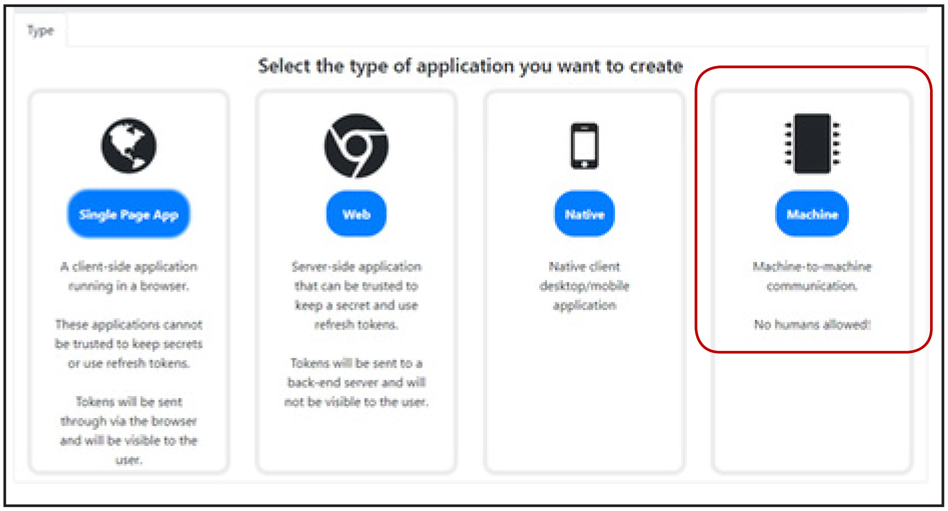


Clients

Name	Description	
Single Sign-On Manager UI Angular Client		Reserved
Single Sing-On Manager API Swagger Client		Reserved

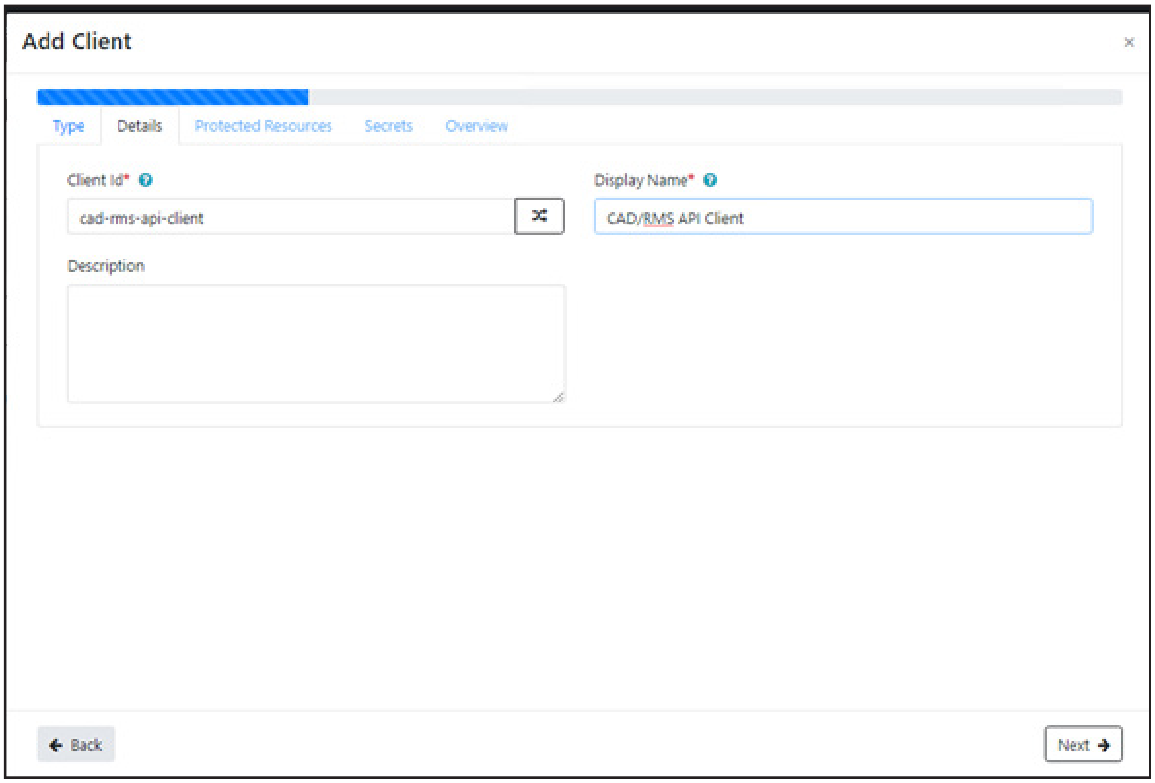
Page 1 of 1 · 10 items per page · 1 - 2 of 2 items

e. Select **Machine** as the type of application you want to create.



Type
Select the type of application you want to create
Single Page App
A client-side application running in a browser.
These applications cannot be trusted to keep secrets or use refresh tokens.
Tokens will be sent through via the browser and will be visible to the user.
Web
Server-side application that can be trusted to keep a secret and use refresh tokens.
Tokens will be sent to a back-end server and will not be visible to the user.
Native
Native client desktop/mobile application
Machine
Machine-to-machine communication.
No humans allowed!

f. Complete the **Client ID**. The Client ID will be used in the config file.



Add Client

Client Id*: cad-rms-api-client

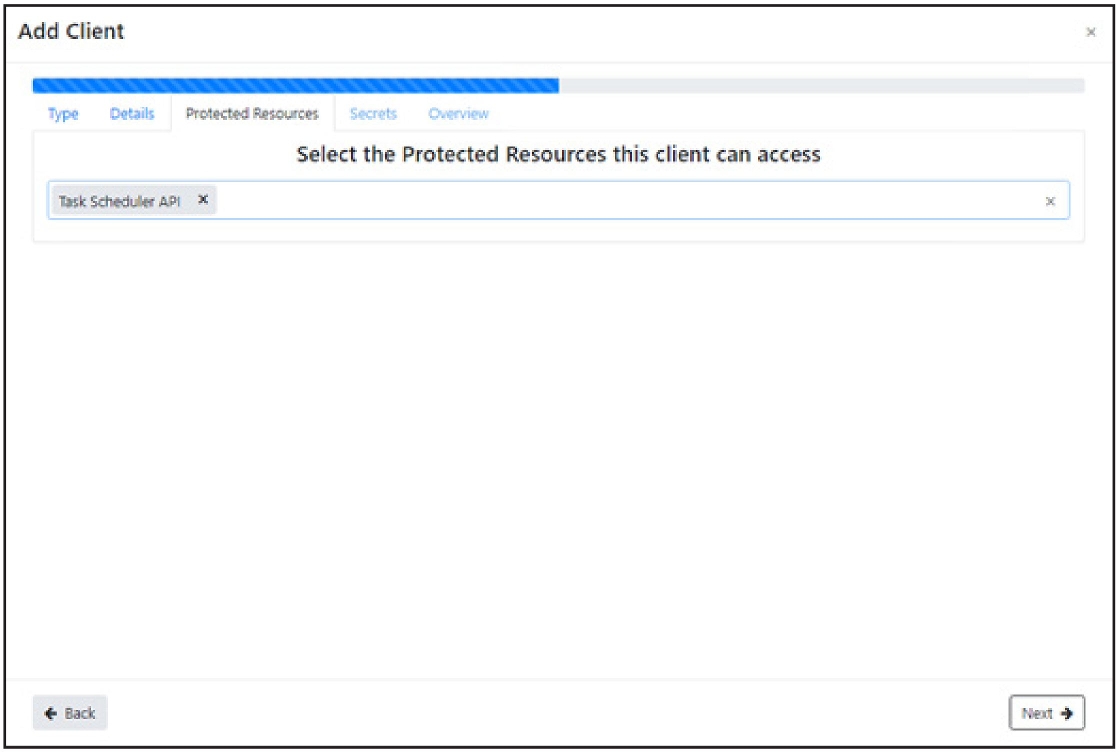
Display Name*: CAD/RMS API Client

Description:

NOTE: Task Manager Resource

Another resource for the task manager should have been created before, in the CAD Client this resource should be related to allow create jobs in Task Manager (See ***Task Manager Deployment***.)

g. In **Protected Resources** add the **Task Manager Resource** to the list.

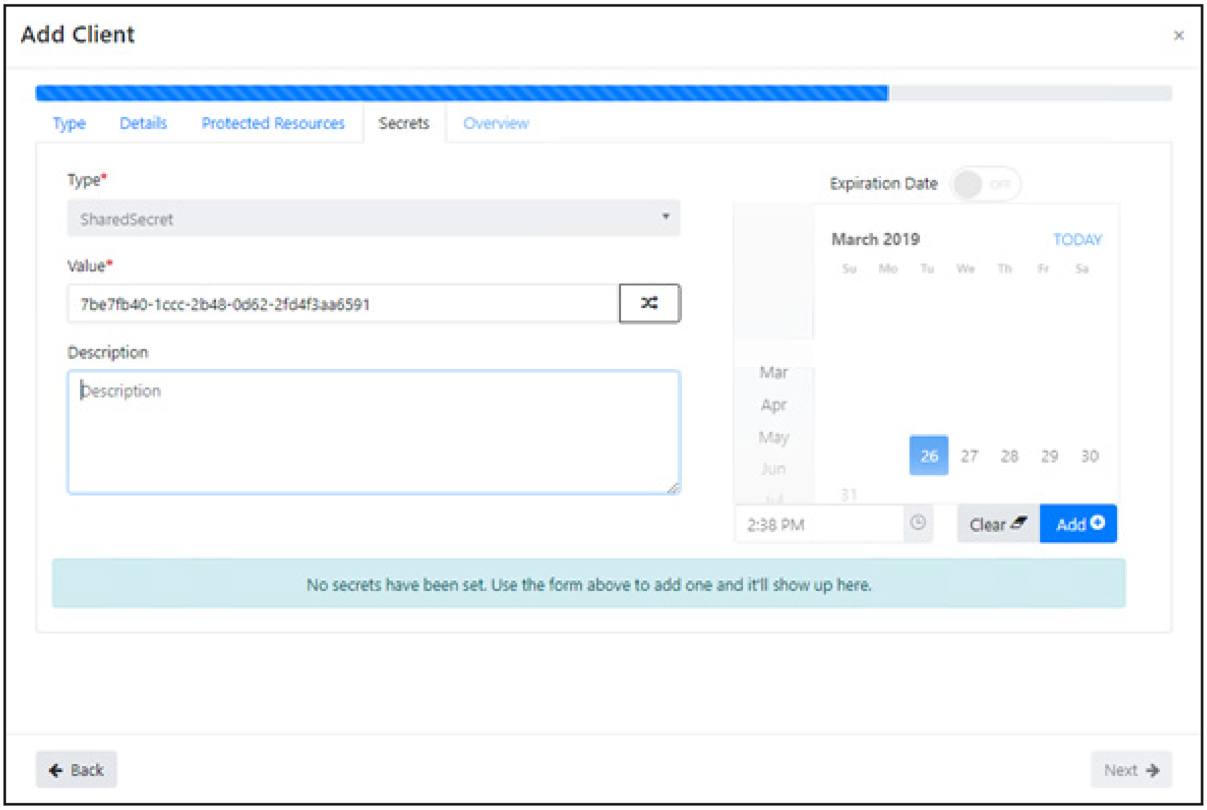


Add Client

Select the Protected Resources this client can access

Task Scheduler API ×

h. Add a **Secret** for this client. This Secret will be used in the config file.



Add Client

Type*: SharedSecret

Value*: 7be7fb40-1ccc-2b48-0d62-2fd4f3aa6591

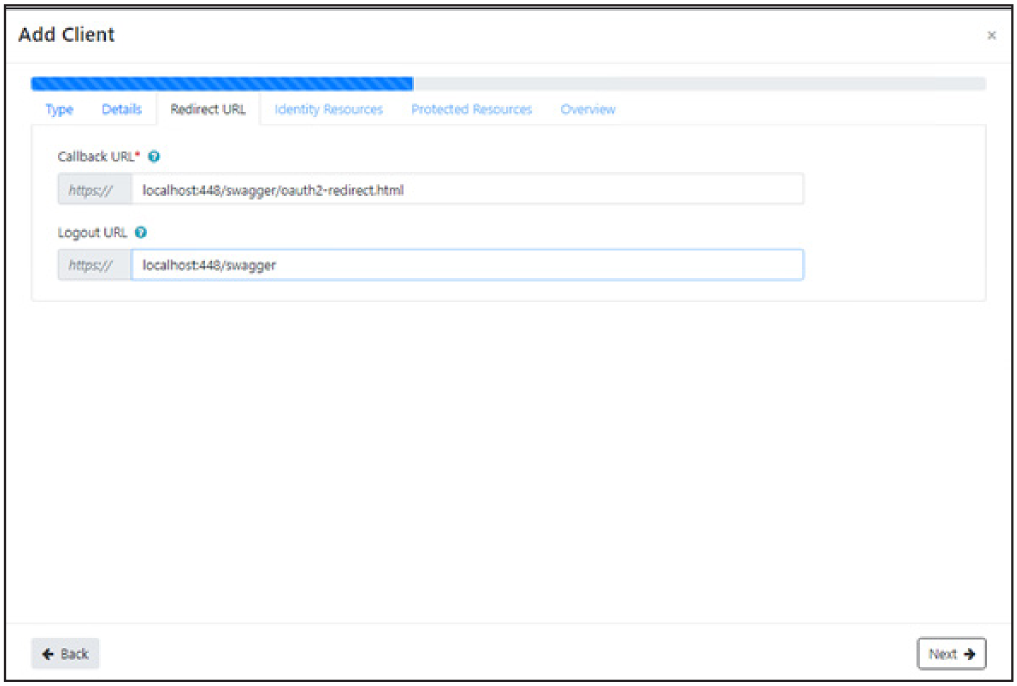
Description: Description

Expiration Date: OFF

No secrets have been set. Use the form above to add one and it'll show up here.

i. [Optional] If you want to enable access to the API via Swagger, create another client with the name `cad-rms-api-swagger`.
This client should be created as a **Single Sign App**.

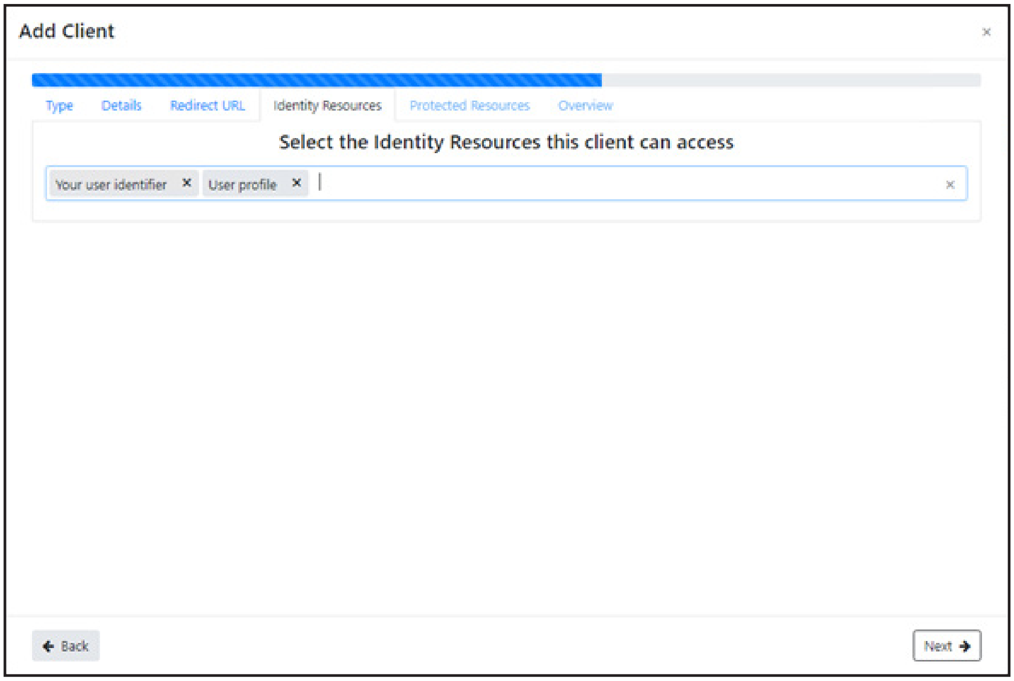
j. Click to select Redirect URL, thendefine the URLs as shown in the image below. Change the localhost to the server **DNS**.



Callback URL* https:// localhost:448/swagger/oauth2-redirect.html

Logout URL https:// localhost:448/swagger

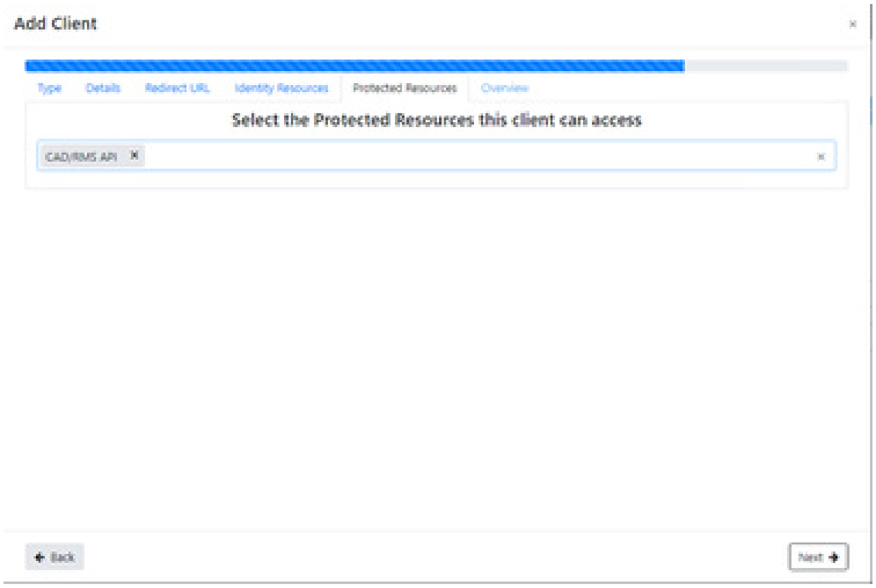
k. Click to select **Identity Resources**, then select **Your user identifier** and **User profile**.



Select the Identity Resources this client can access

Your user identifier × User profile ×

l. Click to select **Protected Resource,** then select CAD/RMS API that was previously created.



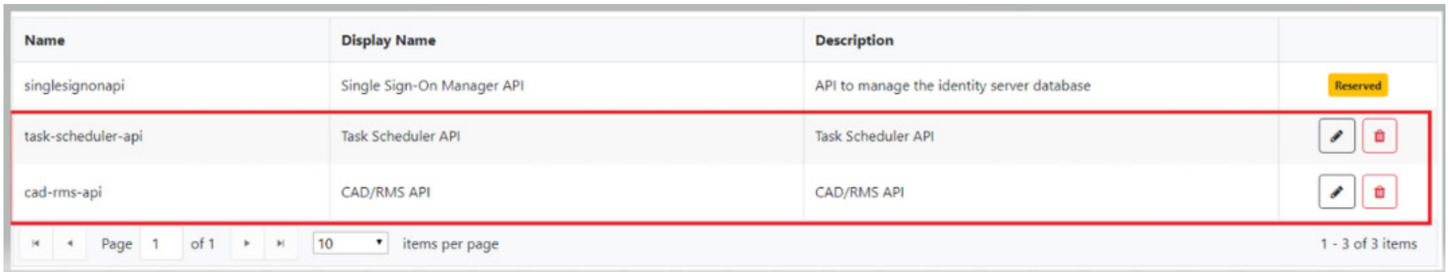
Add Client

Select the Protected Resources this client can access

CAD/RMS API ×

m. Finish, and check that all clients and resources are created as required by CAD/API.
CAD/API requires the resources and clients shown in the following images are created as **Single Sign On**.

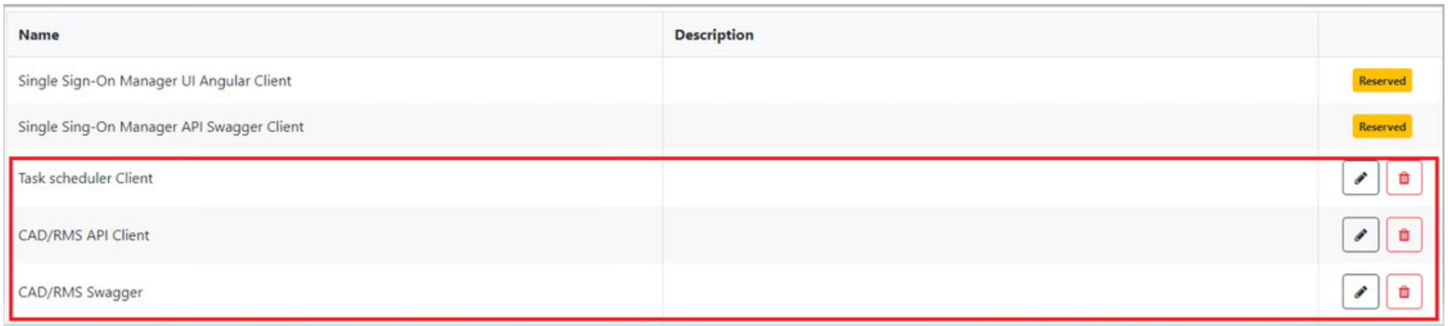
Resources Required to be Single Sign On



Name	Display Name	Description	
singlesignonapi	Single Sign-On Manager API	API to manage the identity server database	Reserved
task-scheduler-api	Task Scheduler API	Task Scheduler API	
cad-rms-api	CAD/RMS API	CAD/RMS API	

Page 1 of 1 · 10 items per page · 1 - 3 of 3 items

Clients Required to be Single Sign On



Name	Description	
Single Sign-On Manager UI Angular Client		Reserved
Single Sing-On Manager API Swagger Client		Reserved
Task scheduler Client		
CAD/RMS API Client		
CAD/RMS Swagger		

12. Locate the file **appsettings.json** in the app's folder and open it as **Administrator** in a text editor to change the following values:

    a. Select **True** if the CAD/RMS API is going to be used for DVMS.

    Coban Product

    ```
    "CobanProduct": {
            "IsDvms": false
        }
    ```

    b. In the section **IdentityServerAuthOptions**, change the values according to the clients and resources that were created on **Single Sign On**.

    - **ClientName**: Client's name for CAD/API Client.
    - **ClientSecret**: Valid secret for CAD/API Client.
    - **Authority**: Url of Single Sign On Server.
    - **ApiName**: Resources' name for CAD/API Resource.
    - **ApiSecret**: Valid secret for CAD/API Resource.

    Identity Server Auth Options

    ```
    "IdentityServerAuthOptions": {
      "ClientName": "cad-rms-api-client",
      "ClientSecret": "7be7fb40-1ccc-2b48-0d62-2fd4f3aa6591",
      "Authority": "https://dev-id.seon.com",
      "RequireHttpsMetadata": true,
      "ClientScopes": [
         "task-scheduler-api"
      ],
      "ApiName": "cad-rms-api",
      "ApiSecret": "3b147dd7-6786-b151-a6c7-10f3c8953bfc"
    }
    ```

c. Add the valid **URL** for the task manager that will be used by CAD/RMS API (Check Prerequisites). Replace **localhost:44369** with the **Task Manager API URL**.

Task Manager

```
"TaskManager": {
        "AddJobUrl": "https://localhost:44369/api/Jobs/http"
    }
```

d. Provide the match **URL**. Replace **localhost:44369** with the **CAD/RMS API Url**.

Cad Rms Api

```
"TaskManager": {
        "AddJobUrl": "https://localhost:44369/api/Jobs/http"
    }
```

e. Provide the following Smtp configurations to allow sending emails:

- **SmtpUserName**: Name of the user. All mail will be sent using this name.
- **StmpHost**: URL of the SMTP server.
- **StmpUser**: User that can send emails using STMP Server.
- **StmpPassword**: Password for the user that will be used to send emails.
- **SmtpEnableSsl**: True if enabled ssl is available for the SMTP Server.
- **StmpPort**: Port that should be used to send emails.
- **AdministratorEmail**: The CAD/RMS administrator email.
- **AdministratorName**: Administrator's name.

Smtp Configuration

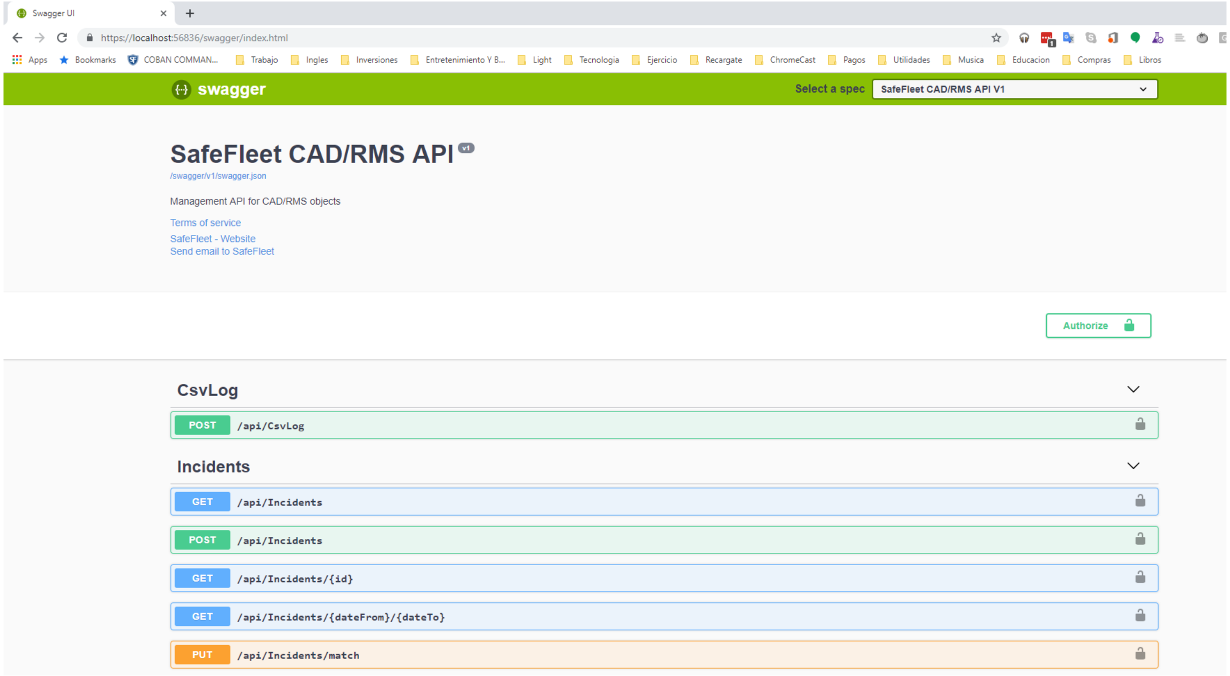
```
"SmtpConfiguration": {
    "SmtpUserName": "Coban Admin",
    "StmpHost": "SETAVALIDHOST",
    "StmpUser": "ASKFORVALIDUSERTOADMIN",
    "StmpPassword": "ASKFORPASSWORDTOADMIN",
    "SmtpEnableSsl": false,
    "StmpPort": 587,
    "AdministratorEmail": "test@coban.com",
    "AdministratorName": "Test Coban"
  }
```

f. Change the connections string to use the DVMS Sql Server Database.

Connection String

```
"ConnectionStrings": {
        "Cad_Rms_SqlDatabase": "data source=localhost;initial catalog=Coban-
BO;integrated security=True;Min Pool Size=100;Max Pool Size=5000;Connect Time-
out=60;MultipleActiveResultSets=True;Encrypt=True;TrustServerCertificate=True"
    }
```

13. Start the site on IIS and open the Swagger app to check that the CAD/RMS API is working. Usually the Swagger URL is https://localhost/swagger, where localhost is a user-defined value. The result should be similar to as shown below:



SafeFleet CAD/RMS API

/swagger/v1/swagger.json

Management API for CAD/RMS objects

Terms of service
SafeFleet - Website
Send email to SafeFleet

CsvLog

POST /api/CsvLog

Incidents

GET /api/Incidents
POST /api/Incidents
GET /api/Incidents/{id}
GET /api/Incidents/{dateFrom}/{dateTo}
PUT /api/Incidents/match

Deploying CAD/RMS Window Service

A windows service must be created to process the CSV Files. Before installing, change the **appsetting** file with the configuration values below.

1. Download from the release notes the zip package that contains all files needed to install the service. Usually are located here.
2. Unzip on the computer that will be used to install the service.
3. Register a new application on Single Sign ON (This new application can use the same server as used for the API).
4. Locate in the app's folder the file **appsettings.json** and open it as administrator in a text editor to change the next values:
   a. Configure **CSVDropbox** to be the local folder that the file monitor is going to be checking. All CSV files that are copied to this folder will be processed by the service.

      CSV Dropbox

      ```
      "CSVDropbox": "D:\\cobanstage\\csv"
      ```

   b. Configure **DeletionRetry**: Define the maximum number of retries when trying to delete a file. We recommended 10.

      Deletion Retry

      ```
      "DeletionRetry": 10
      ```

   c. Configure **CertificatesPaths**: Define the local paths for the public and private keys used to Encrypt and Decrypt the CSV Files. The process to generate this certificates will be described in step *"5. Generate a public and a private certificate to be used during the process. The public key should be used to encrypt the file content, and the private key will be used to read the Encrypted CSV Content." on page 16*.

      Certificates Paths

      ```
      "CertificatesPaths": {
          "Public": "C:\\Temp\\rsa_2048_pub.pem",
          "Private": "C:\\Temp\\rsa_2048_priv.pem"
        }
      ```

d. Configure **CadRmsApi**: Change the **Localhost** to the URL where the CAD/RMS API can be accessed

Cad Rms Api

```
"CadRmsApi": {
    "PostIncidentsUrl": "https://localhost:56836/api/Incidents",
    "PostLogsUrl": "https://localhost:56836/api/CsvLog"
  }
```

e. Configure **IdentityServerAuthOptions**: Change with the values that where registered on **Single Sign On**.

- **ClientName**: Client's name for CAD/RMS File Monitor client.
- **ClientSecret**: Valid secret for CAD/RMS File Monitor client.
- **Authority**: Url of Single Sign On Server.
- **ApiName**: Resources's name for CAD/API Resource.
- **ApiSecret**: Valid secret for CAD/API Resource.

Identity Server Auth Options

```
"IdentityServerAuthOptions": {  "ClientName": "cad-rms-file-monitor-client",
    "ClientSecret": "0b07adc3-a338-989f-3274-b98d5b33fd14",
    "Authority": "https://dev-id.seon.com",
    "RequireHttpsMetadata": true,    "ClientScopes": [
     "cad-rms-api"
    ],
    "ApiName": "cad-rms-api",
    "ApiSecret": "3b147dd7-6786-b151-a6c7-10f3c8953bfc"
 }
```

f. Configure **SFTPSession**: Configure the **SFTP Server** that will be used to check according to the defined interval.

- **HostName** - The server name which will host the SFTP site.
- **UserName** - The login user.
- **Password** - The login password.
- **Fingerprint** - Used as public key to identify the host machine.
- **RemotePath** - The location where files are hosted.
- **CheckIntervalInMin** - Used as the time interval for checking for new files at SFTP.

SFTP Sessions

```
"SFTPSession": {
    "HostName": "sftp.com",
    "UserName": "usr",
    "Password": "pwd",
    "Fingerprint": "ssh-rsa 2048 xx:xx:xx:xx:xx:xx:xx:xx...",
    "RemotePath": "sftp_remote_path",
    "CheckIntervalInMin":  1
}
```

5. Generate a public and a private certificate to be used during the process. The public key should be used to encrypt the file content, and the private key will be used to read the Encrypted CSV Content.

a. Install **Open SSL**. Open **SSL Light**.

b. Open a **Powershell** console as Administrator.

c. Run the following command to generate the private certificate:

Generate Private Certificate

```
openssl genrsa -out rsa_2048_priv.pem 2048
```

d. Run the following command to export the public key:

Export public key

```
openssl rsa -pubout -in rsa_2048_priv.pem -out rsa_2048_pub.pem
```

e. Copy to the **Path** that will be used in the appsetting file.

The public key should be used by the clients that will encrypt the CSV File Content.

6. Install the service in the server.

    a. Open a Powershell Console as Administrator and open the path where the Windows service files are located. See *"2. Unzip on the computer that will be used to install the service." on page 14.*

    b. Install the service running the following Command:

Install Windows Service Command

```
.\RegisterService.ps1
     -Name MyService
     -DisplayName "My Cool Service"
     -Description "This is the Sample App service."
     -Exe "c:\svc\SampleApp.exe"
     -User Desktop-PC\ServiceUser
```

NOTE: Naming the Service

We recommend that you use a name that reflects what the service is, such as **COBAN CADRMS API**. Using such an obvious name will help in the unlikely event that you require IT help. A familiar name will be easier for IT to find.

If you do use a name other than suggested, we recommend that you email that name to ***hwsupport@cobantech.com*** for documentation purposes. Failure to notify Support in advance could delay service.

    c. Use the following real command to install the service. The user who installed the service should have full access over the folder that the **Monitor** is going to be watching.

Install Windows Service Command (Real)

```
.\RegisterService.ps1 -Name CadFileMonitor -DisplayName "Cad File Monitor" -Description "Cad File Monitor" -Exe "C:\CadFileMonitor\SafeFleet.Cad-Rms.FileMonitor.exe" -User CBSQA5\cobanadmin
```

    d. Use the next command to start the service on **Powershell** or in the **Services Desktop App**.

Start the service

```
Start-Service -Name CadFileMonitor
```