

# Eze EMS xAPI .NET Sample Application Guide

This document contains information about getting started with Eze EMS xAPI using .NET.



**Confidentiality Notice:** The information included in this document is confidential information of SS&C Eze and is intended only for SS&C Eze and its affiliates, Eze EMS clients, and their respective employees.

## Table of Contents

<b>Revision History</b>	<b>3</b>
<b>Introduction</b>	<b>4</b>
Eze EMS xAPI Basics	4
Eze EMS xAPI Use Restrictions	4
Eze EMS xAPI Version	5
Download EMS xAPI	5
Developer Support	5
<b>Getting Started Using .NET (C#)</b>	<b>6</b>
Prerequisites	6
Start Using .NET	6
Visual Studio Initial Setup	6
Connect using .NET	7

## Legal Information

### Copyright

This document is the copyrighted work of Eze Castle Software LLC ("SS&C Eze"). SS&C Eze distributes this document pursuant to a subscription agreement containing confidentiality and license provisions and is solely for the benefit of its authorized licensees. This documentation may not be copied or transmitted, in whole or in part, in any form or by any means without the express written consent of SS&C Eze.

© 1997 to 2024 Eze Castle Software LLC. All Rights Reserved.

### Content

Information in this document is subject to change without notice. In the event that you are using a version of SS&C Eze products other than the most recent version, there may be discrepancies between the content of this documentation and the operation or visual presentation of your older version of the product. SS&C Eze does not warrant that this documentation is error free.

### Trademarks

SS&C Eze is a trademark of SS&C Technologies, Inc. All SS&C Eze company and product names are trademarks or registered trademarks of SS&C Technologies, Inc. or SS&C Eze.

All other company or product names mentioned herein are the trademarks or registered trademarks of their respective companies.

SS&C Eze

<http://www.ezesoft.com/>

## Revision History

The table below provides a snap-shot of the updates in each revision of this document. A bar is displayed on the right side of the page to help you identify updates in the current release.

Version No.	Date	Summary of Update
v2024.7.1.1651, v2024.7.0.1636, v2024.6.1.1634, v2024.6.0.1615, v2024.5.0.1576, v2024.4.0.1473, v2024.2.0.1355, v2024.1.0.1209, v2023.8.1.1167, v2023.8.0.1146, v2023.7.0.887, v2023.6.0.723	NA	There are no documentation updates in these EMS xAPI releases.
v2023.5.0.947	Sep 04, 2023	Initial release.

# Introduction

The purpose of this document is to help clients get started with the EMS xAPI application using .NET. This document provides a step-by-step process of using EMS xAPI with the Visual Studio (2022 or later) built with .NET to operate the APIs.

Eze EMS xAPI is robust and easy-to-use application that allows programmers and trading businesses to complete various trading workflows, and also access key information, including:

- Automating order routing - to smart order routers, algorithms and other trading systems.
- Routing orders to multiple brokers, dark pools, ATS, and MTFs via the Eze EMS Global Routing Network - across asset classes.
- Staging or routing single or pairs orders.
- Accessing balances, positions, executions, and other order details.
- Accessing comprehensive list and basket capabilities.

Although EMS xAPI can operate with all gRPC compatible languages, only Java language references are provided in this document as an example. Refer to this [link](#) for more information on gRPC.

## Eze EMS xAPI Basics

The Eze EMS xAPI operates in conjunction with your existing Eze EMS account permissioning and entitlements. **The Eze EMS xAPI is not a standalone data feed application that is provided to you independent of the Eze EMS.** Please contact Eze Client Service if you need to request or make changes to appropriate permissions for your account.

## Eze EMS xAPI Use Restrictions

As an Eze EMS xAPI user, you are prohibited from retransmitting any Eze Market Data using the Eze EMS xAPI, without the express prior written consent of Eze EMS and the exchanges or other third-party data providers (referred to as “**Sources**” in your end user agreement). Any unauthorized retransmission of Eze Market Data is a breach of your end user agreement and will cause immediate termination of your use of the Eze EMS, Eze Market Data, and the Eze EMS xAPI.

Any non-display usage of Eze Market Data, such as use of real-time data in algorithmic trading or program trading, is subject to the rules, regulations, and policies of the applicable exchanges and additional exchange fees may apply. In addition, you may have a non-display usage of Eze Market Data even if a display of real-time data occurs. Please review your Eze EMS end user agreement, and the exchanges’ and third-party data providers’ rules, regulations, and policies that apply to your use of the Eze EMS API (which apply to Eze EMS xAPI) and/or Eze Market Data. It is the sole responsibility of the Eze EMS xAPI user and each user receiving, directly or indirectly accessing or otherwise using Eze Market Data to determine whether your receipt, access or use is reportable and/or fee liable.

## Eze EMS xAPI Version

This document covers all the APIs and updates to the Eze EMS xAPI that are part of 2024.7.1.1651 release.

## Download EMS xAPI

Contact your SS&C Eze client service representative for downloading Eze EMS xAPI.

## Developer Support

- If you are an existing Eze EMS user, [log in](#) to access developer support documentation and sample code.
- You can [contact us](#) or [request a demo](#) if you want to explore more about EMS xAPI.
- You can send us an e-mail [apisupport@ezesoft.com](mailto:apisupport@ezesoft.com) or call +1 312-442-8122.

# Getting Started Using .NET (C#)

You can now launch Eze EMS xAPI using the .NET (C# or C-sharp) language, which is hosted on the Visual Studio.

## Prerequisites

The following prerequisites are needed to successfully run the EMS xAPI on .NET:

- Visual Studio (version 2022 or later) installed on your machine.
- Latest version of Eze EMS xAPI .NET (C# or C-sharp) package. Contact your SS&C Eze client service representative for the latest package.

## Start Using .NET

The Eze EMS xAPI can be launched using .NET language. The following sections provide information on how to create and run a .NET console application in Visual Studio (version 2022 or later).

**To start using EMS xAPI on .NET:**

1. [Visual Studio Initial Setup](#)
2. Connect using .NET
  - [Unsecured connection](#)
  - [Secured connection](#)

## Visual Studio Initial Setup

**To install and set up visual studio:**

1. Visit <https://visualstudio.microsoft.com/downloads/>, select the version to download.
2. Navigate to your default downloads folder and open the downloaded file from step 1. Follow the prompts to install Visual Studio.
3. Launch Visual Studio.
4. Click **Open a local folder**, as seen on the following page. Select the **CSharp** folder you received from the SS&C Eze client service representative, as described in the [Prerequisites](#) section.



**Note:** To create a .NET console application using Visual Studio, refer to the link <https://learn.microsoft.com/en-us/dotnet/core/tutorials/with-visual-studio>.

## Connect using .NET

With the Visual Studio (2022 or later) installed on your machine, you can create the console using .NET to operate on EMS xAPI.

### Establish unsecured connection

To establish an unsecured connection (i.e., http):

1. Open the Visual Studio with setup for .NET Core 6.0.
2. Click **config.cfg** file from the **Solution Explorer** section on the right and add your account specific details for the following:

```
password = <password>
server = <server>
user = <username>
domain = <domain>
locale = <locale>
port = <port>
ssl = false
```

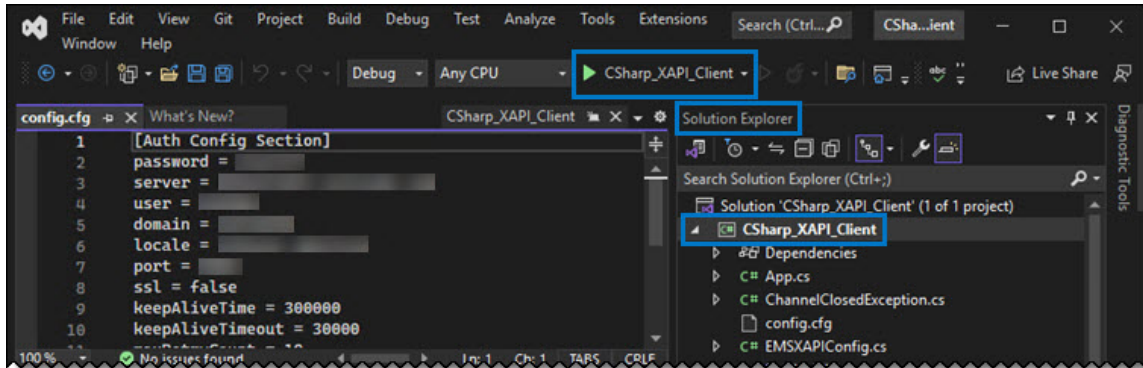


**Note:** To establish a unsecured connection make sure the value is set as **ssl = false**.



**Note:** Contact your SS&C Eze client service representative for assistance with your account details.

- Click **CSharp\_XAPI\_Client** file from the **Solution Explorer** section on the right and then click **CSharp\_XAPI\_Client** in the top ribbon, as shown below.



On successful connection, a success message is displayed on the screen, shown below.

```
Hello World!  
06670980_7fa4dcdb4b9dad0e  
[2023-07-25 20:13:46] HeartBeat status: Live | Connection is Live.  
No records found  
[2023-07-25 20:13:51] HeartBeat status: Live | Connection is Live.
```

## Establish secured connection

To establish a secured connection (i.e., https):

- Open the Visual Studio with .NET Core 6.0 setup.
- Click **config.cfg** file from the **Solution Explorer** section on the right and add your account specific details. Contact your SS&C Eze client service representative for assistance with your account details.

```
password = <password>  
server = <server>  
user = <username>  
domain = <domain>  
locale = <locale>
```



```
port = <port>
ssl = true
```



**Note:** Make sure the value is set as **ssl = true** to install the certificate and establish a secured connection.

3. Download the latest roots.pem file from [gRPC GitHub](#) repository.
4. Make sure the downloaded **roots.pem** file is placed in the same CSharp folder you received from the SS&C Eze client service representative, as described in the [Prerequisites](#) section.
5. Run the following command in command prompt to install certificate from the **roots.pem** file:

```
certutil.exe -addstore -f "Root" "C:\CSharp\roots.pem"
```

6. Click **CSharp\_XAPI\_Client** file from the **Solution Explorer** section on the right and then click **CSharp\_XAPI\_Client** in the top ribbon.

On successful connection, a success message is displayed on the screen, shown below.

```
Hello World!
06670980_7fa4dcdb4b9dad0e
[2023-07-25 20:13:46] HeartBeat status: Live | Connection is Live.
No records found
[2023-07-25 20:13:51] HeartBeat status: Live | Connection is Live.
```