



OPC UA **Historian**

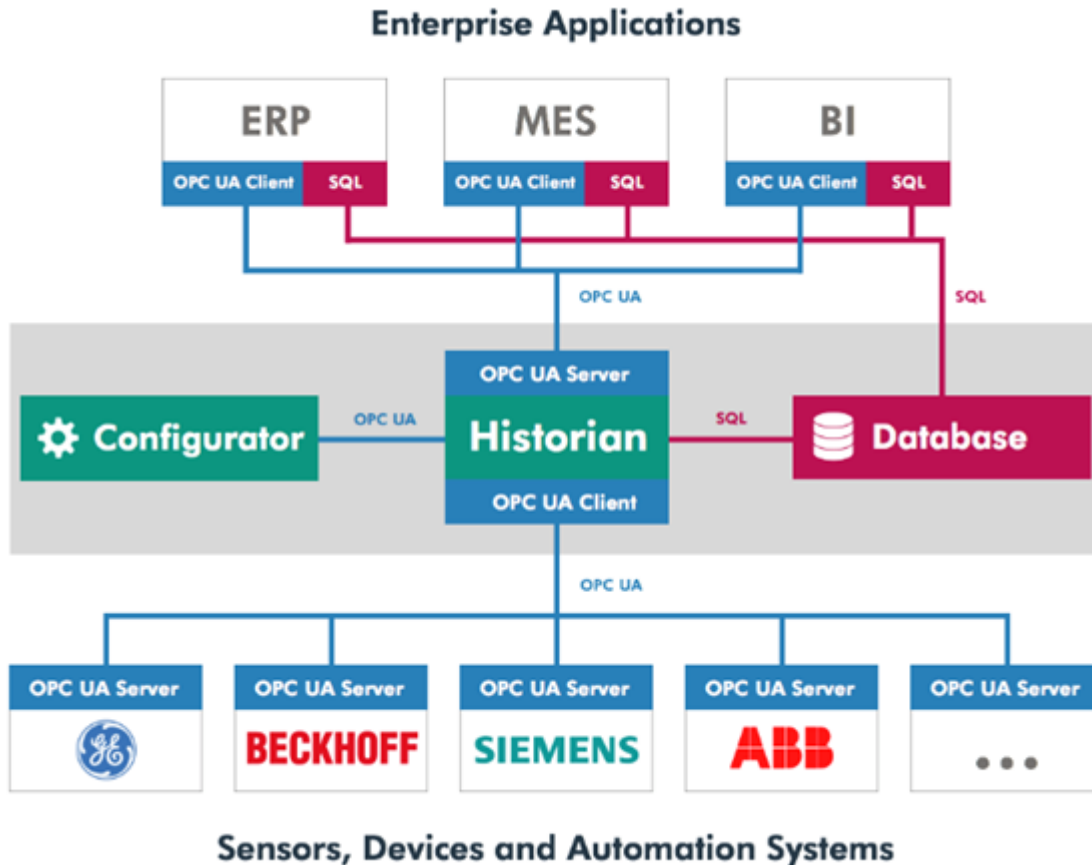
User Manual
Version 1.1.0

Table of Contents

Introduction	1
OPC UA Client and Server	1
Historian Modules	1
1. Installation	2
1.1. Historian Installation	2
1.1.1. Windows Installation	2
1.1.2. Linux Installation	2
1.2. Database Installation	3
1.2.1. MySQL	3
1.2.2. MariaDB	3
1.2.3. Microsoft SQL Server	3
1.2.4. Performance tips	4
2. Uninstallation	5
2.1. Windows Uninstallation	5
2.2. Linux Uninstallation	5
3. Base Configuration	6
3.1. Database Connection	6
3.2. Server Endpoints Configuration	7
3.2.1. UA TCP Transport Protocol	8
3.2.2. UA HTTPS Transport Protocol	8
3.3. Application Parameters	9
4. Running Historian	10
4.1. Starting Historian Server	10
5. Application License	12
6. Configuring Source Servers	14
7. Configuring Data Collection	17
8. Connecting to Historian from an OPC UA Client Application	21
9. Secure Connections	23
10. Configuration Folder	24
10.1. Windows Configuration Folder	24
10.2. Linux Configuration Folder	24
11. History Database Format	25

Introduction

Prosys OPC UA Historian is used to provide history data on top of OPC UA servers that are not capable of storing history data by themselves. It also provides a single access point for data from several underlying servers.



OPC UA Client and Server

Historian acts as an OPC UA client towards the underlying servers. These servers are called Source Servers in the application. On the other hand, it also acts as an OPC UA server and can be accessed from OPC UA client applications. All current values of the underlying servers are available for OPC UA clients via Historian. In addition, the stored history data is available for OPC UA clients through the same OPC UA interface.

Historian Modules

Historian consists of two modules:

- **Configurator** is the user interface part of the application. It's used to manage the configuration of the Historian Server.
- **Historian Service** is the part of the software that runs in the background and takes care of storing the history data into the database. It is installed as a service and starts automatically when the computer restarts.

1. Installation



If upgrading from version 1.0.0, you should first manually uninstall the previous version.

1.1. Historian Installation

Prosystech OPC UA Historian is available for download at <http://www.prosysopc.com> upon request. After a successful request, you are directed to the product download page, from which you can select the correct installation package for your target environment.

1.1.1. Windows Installation

On Windows, run the installer executable `prosys-opc-ua-historian-windows-x64-x.y.z-b.exe` and follow the instructions. By default, the application is installed in the folder *Program Files/ProsystechOPC/Prosystech OPC UA Historian*.

The installer also installs the application as a Windows service, making it available in unattended operation.

1.1.2. Linux Installation



The application requires a GUI (Linux Desktop Environment) in order to run.

On Linux, first open the terminal and navigate to the directory of the downloaded `.sh` file. Then add a file permission to make the installation shell script executable with the command

```
sudo chmod u+x prosys-opc-ua-historian-linux-x.y.z-b.sh
```

Then run the installation shell script with the command

```
sudo ./prosys-opc-ua-historian-linux-x.y.z-b.sh
```

This will open the installer where you can follow the steps to complete the installation. By default, the application is installed in the folder *opt/prosys-opc-ua-historian*.

The installer also installs the application as a Linux service, making it available in unattended operation. If you do not have a graphical user interface available, this unattended operation mode is the only one you can run (which means you have to create the configuration elsewhere and transfer it).

1.2. Database Installation

Historian supports MySQL, MariaDB and Microsoft SQL Server database solutions. One of these must be installed in order for the Historian to work. All configuration settings as well as history data are stored in the database.

1.2.1. MySQL

Download MySQL from <https://www.mysql.com/>. Select suitable package and install it using instructions given on the page.



To increase the MySQL performance, you can increase the InnoDB buffer pool size. See the following links:

- <https://dev.mysql.com/doc/refman/5.5/en/innodb-buffer-pool.html>
- <https://www.percona.com/blog/2007/11/01/innodb-performance-optimization-basics/>

1.2.2. MariaDB

MariaDB is available from <https://mariadb.org/>. Select a suitable package and install it using instructions on the page.



To increase the MariaDB performance, see following link: <https://mariadb.com/kb/en/mariadb/optimization-and-tuning/>

1.2.3. Microsoft SQL Server

You can install Microsoft SQL Server with any instance name. Once installed, you must ensure that the **TCP/IP Protocol** is enabled and **SQL Server Browser** service is running. Also make sure that **SQL Server Authentication** is enabled.

For enabling TCP/IP Protocol, use the SQL Server Configuration Manager ([Figure 1, "Microsoft SQL Configuration manager."](#)).

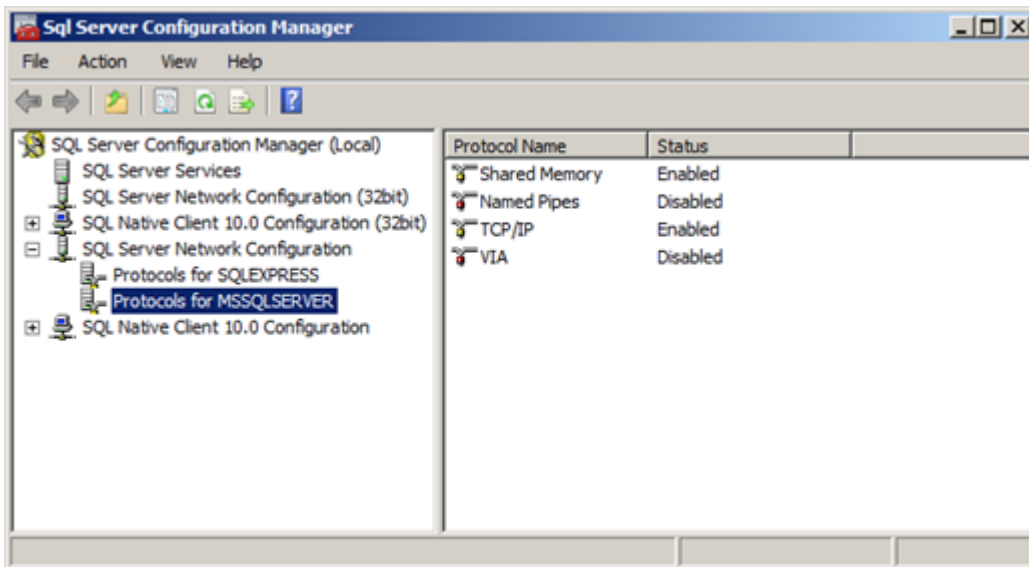


Figure 1. Microsoft SQL Configuration manager.

The SQL Server Browser service can be started from Windows Services (Figure 2, “Microsoft SQL Service.”). Follow the SQL Server instructions to enable SQL Server and Windows Authentication mode in the database.

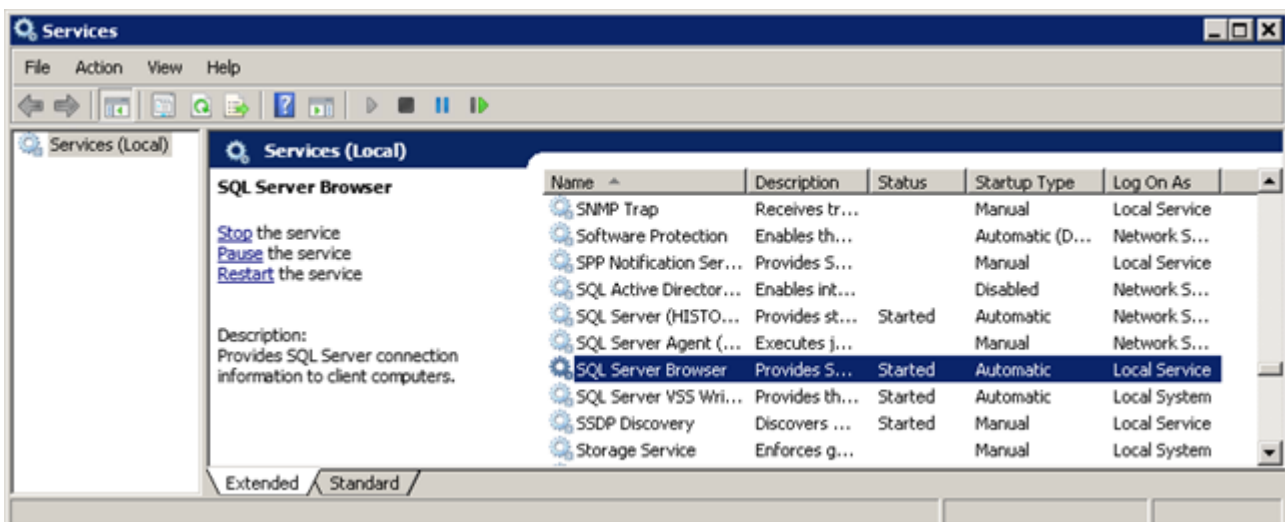


Figure 2. Microsoft SQL Service.



To increase the Microsoft SQL Server performance, you can optimize disk caching. See following link: <https://support.microsoft.com/en-us/kb/234656>

1.2.4. Performance tips

If you have a virus scanner running on the machine, note that it might scan the writes to the database, which will slow it down remarkably. Also, note that on the latest Windows operating systems Windows Defender service is active by default – consider switching that off as well, if possible. The best case would be to allow database write from the virus scanner.

2. Uninstallation



Before uninstalling the OPC UA Historian make sure that **Configurator** is not running.

2.1. Windows Uninstallation

On Windows the application can be uninstalled through the Control Panel or the *Apps & features* menu, or optionally with the uninstaller that is located in the installation folder.

2.2. Linux Uninstallation

On Linux, open the terminal and navigate to the installation folder (default folder is *opt/prosys-opc-ua-historian*) and use the command `sudo ./uninstall`.

3. Base Configuration

Before you can start configuring OPC UA Historian for data collection, you must first perform a base configuration. This consists of defining the database connection, validating the server endpoints and optionally setting the application parameters. First you need to start up the Configurator.

On Windows, you can start the Configurator from the Desktop shortcut or Windows Start Menu.

On Linux, you can start it from the command line as:

```
sudo /opt/prosys-opc-ua-historian/prosys-opc-ua-historian-config
```

3.1. Database Connection

Database Connection must be configured first (Figure 3, “Database Connection Dialog.”). The following dialog will pop up automatically, when you run Historian for the first time.

Figure 3. Database Connection Dialog.



Make sure that you have installed the database that you want to use as your data store. See [Section 1.2, “Database Installation”](#).

First you need to define the *Database* that you are using.

If you use [MySQL](#) or [MariaDB](#), you must configure the *Host* (computer name or IP address) and *Port* in which the Database Server is running (keep the default value, if you haven't configured them in the database itself). For [MS SQL Server](#), you must configure the *Host*, *Port* and also the name of the instance that you wish to use.

You must also configure the user account that Historian is using to access the database. The user account must have enough privileges to create a new database and tables to the database server. The user account must be configured using SQL Authentication, also in MS SQL Server (make sure that option is enabled as instructed by the database user manual).

The database created by the Historian is named as **ProsystechOPCUAHistorian** in all database instances. It is created automatically, when Historian starts up for the first time after the database connection is configured. As mentioned in the Database Configuration part, you must ensure that the user account configured for the database connection has enough privileges to enable creation of a new database.

3.2. Server Endpoints Configuration

The server endpoints define how OPC UA client applications can connect to Historian ([Figure 4, "Server Endpoint Configure."](#)).

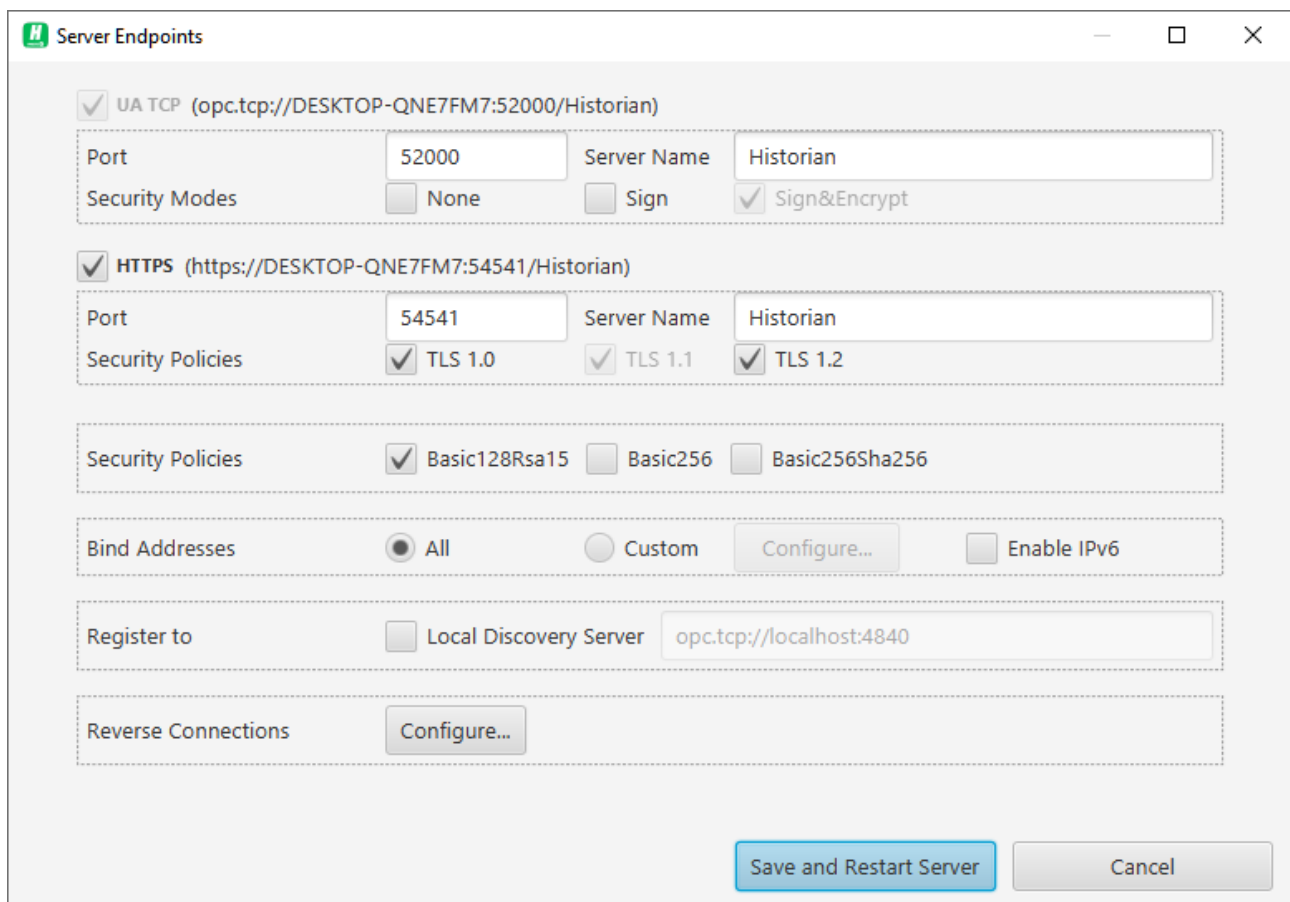


Figure 4. Server Endpoint Configure.

By default, Prosystech OPC UA Historian enables two transport protocols:

- UA TCP
- UA HTTPS

3.2.1. UA TCP Transport Protocol

UA TCP is a mandatory transport protocol in Historian. It is an OPC UA specific binary communication, including full OPC UA specific security implementation. The Port and Server Name define the exact connection address, which is displayed at the top (opc.tcp://<hostname>:53510/Historian).

In addition to the connection address, you can define the security modes that the server accepts. The client applications will decide which mode they wish to use, so the server can only configure which options are available.

Security Mode Sign will ensure that all traffic can be validated by the client and server application and may not be modified during transfer. Security Mode **Sign&Encrypt** will also make all communication between the client and server encrypted, which means that it cannot be seen by any third party that might be monitoring the network traffic. If the client decides to use one of these modes, the client and server application will also need to trust to the certificates of each other.

If you wish to enable insecure connections for testing the server, you can select the **None** option from Security Modes. Using this option in publicly available networks is not recommended.

Security Policies define alternative algorithms that the client applications may choose from. It is important to enable algorithms that the client applications support.

3.2.2. UA HTTPS Transport Protocol

UA HTTPS is an alternative transport protocol, which is not required by OPC UA, but which can enable an alternate communication pathway for some installations.

Security in UA HTTPS is based on TLS. There are different version of TLS and the client and server applications will negotiate the version that they use, based on the ones that they support. The OPC UA applications that support UA HTTPS may define different TLS versions and you will need to make sure that there is at least one common TLS version that both of them support.

In any case, be careful when modifying the settings, since the client applications rely on the connection address and may not be able to connect any more, if you change the endpoint configuration after the system has already been in production.

3.3. Application Parameters

In addition to the connection settings, there are also *Application Parameters*. You can configure settings on the respective dialog. By default, you will find the following settings:

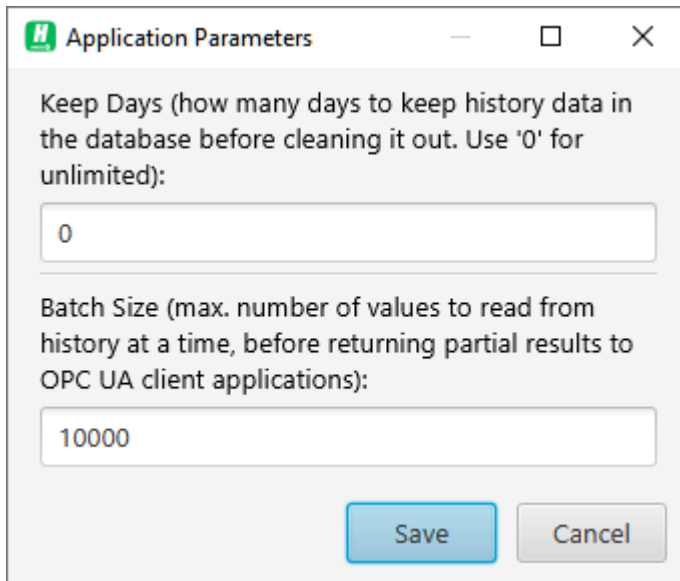


Figure 5. Application Parameters Dialog.

Keep Days tells how many days to keep historical data in the database before cleaning it out. The default value is 0, which means the data is kept indefinitely.



Keep in mind that you are collecting a large amount of data. You might end up filling up your disk space if Keep Days is not configured. Or at least make sure that you have enough disk space to store data in long term.

With **Batch Size**, you can limit the number of values to read from history at a time before returning results to OPC UA client applications.

The batch size affects the performance of data retrieval. If you use a small size, the clients may need to perform many request calls to the server. On the other hand, if the batch size is too big, the clients may experience timeouts. In practice, OPC UA clients may also limit this number themselves, when requesting history data from Historian, so in the end, this is just a safety limit on behalf of Historian.

4. Running Historian



If you are running the application for the first time, you need to start up the Configurator to create the base configuration ([Section 3, "Base Configuration"](#)).

4.1. Starting Historian Server

Once you have configured the [Database Connection](#) and verified the [Server Endpoints](#), you can start Prosys OPC UA Historian Service. This is done by pressing the **Start** button on the Status page ([Figure 6, "Status Page."](#)).

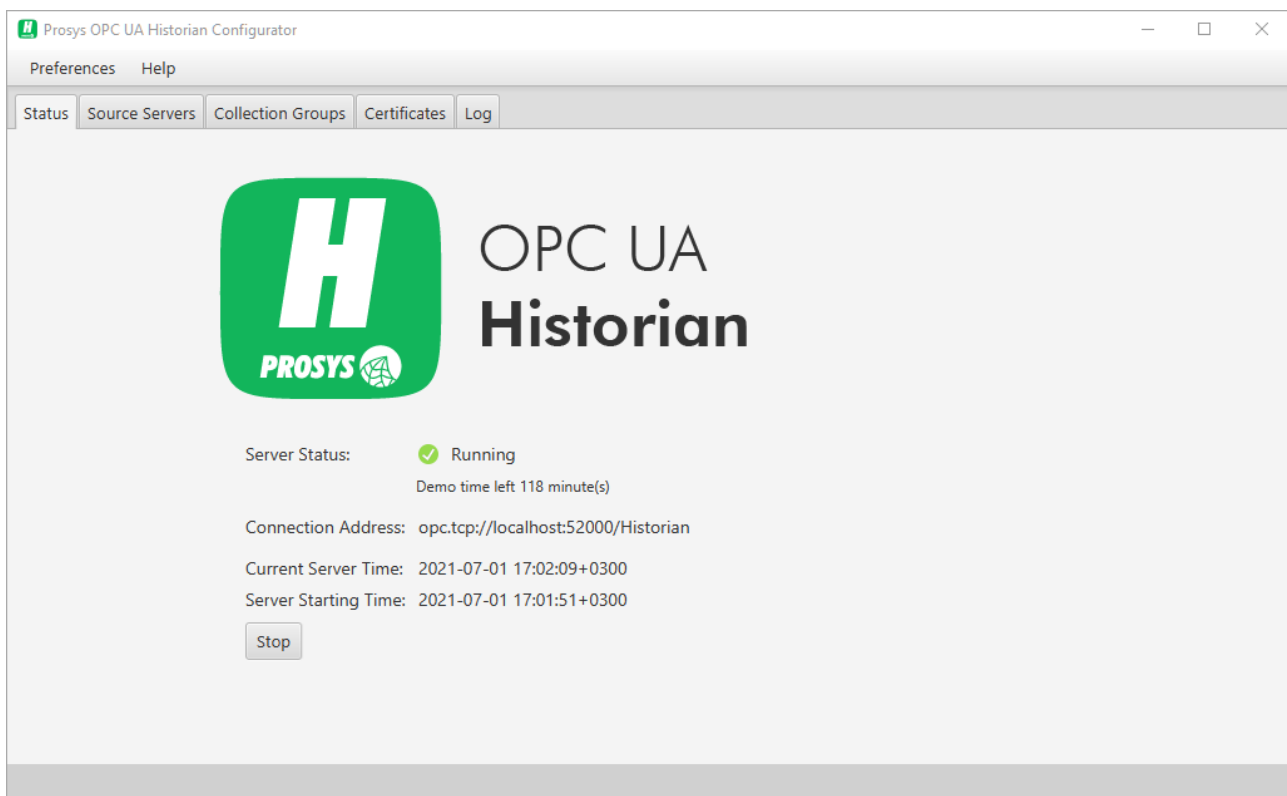


Figure 6. Status Page.

There are alternative ways to start up the service:

On Windows, you can start the service from Service Tab in the Task Manager.

On Linux, you can control the service with the Linux service tools. Or you can use command line as:

```
sudo /opt/prosys-opc-ua-historian/prosys-opc-ua-historian-service <parameter>
```

You can perform four different actions by replacing the **<parameter>** with:

- **start**: Start the service.
- **stop**: Stop the service.
- **restart**: Restart the service.
- **status**: Check the status of the service.

Note that the application requires Administrator privileges, so you have to run it with 'sudo'.

5. Application License

Prosystech OPC UA Historian is a commercial application and requires a license to be purchased for continuous operation. The application may be evaluated in Demonstration Mode for 60 days to validate the feasibility for the installation in question. In Demonstration Mode, it will run 120 minutes, after which it will be stopped. The commercial license is required to let the application run continuously.

The Application is licensed per installation and tied to the hostname of the computer in which it is being run. The license cannot be changed to a new hostname later on.

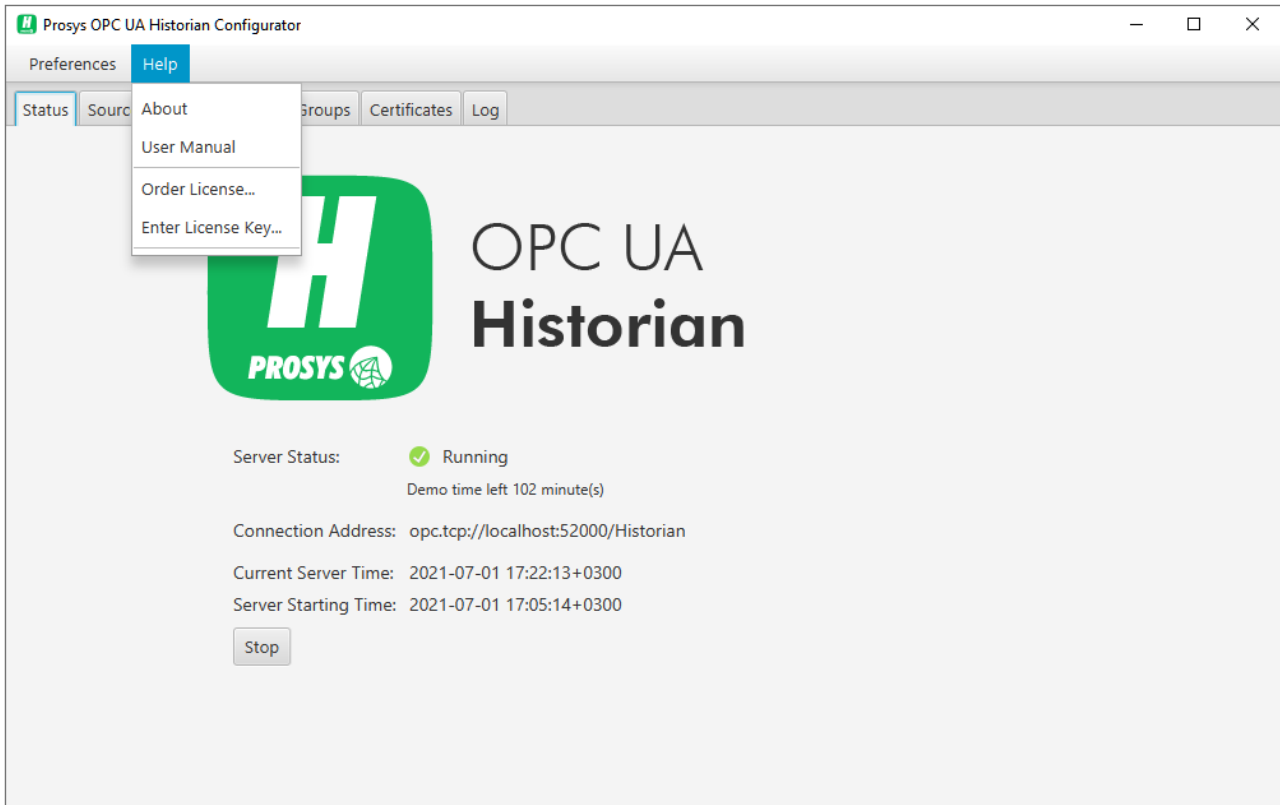


Figure 7. Order License and Enter License are available from the Help menu.

When you are ready to order the license, you will need to find the hostname of the target computer and the name of the Licensee (Person or Organization). The best way to ensure that the information is correct is to use the Order License Dialog that is available from the Help menu (Figure 7, "Order License and Enter License are available from the Help menu."). You can generate a License Request File from the dialog that opens (Figure 8, "Order License Dialog").

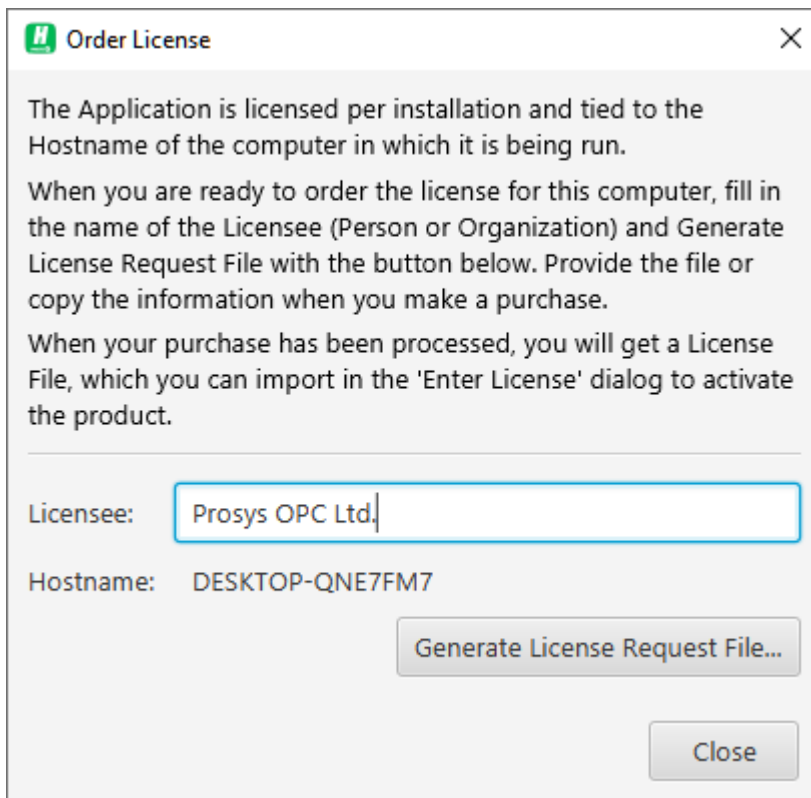


Figure 8. Order License Dialog.

Send the License Request File or the respective information to sales@prosysopc.com when you make a purchase.

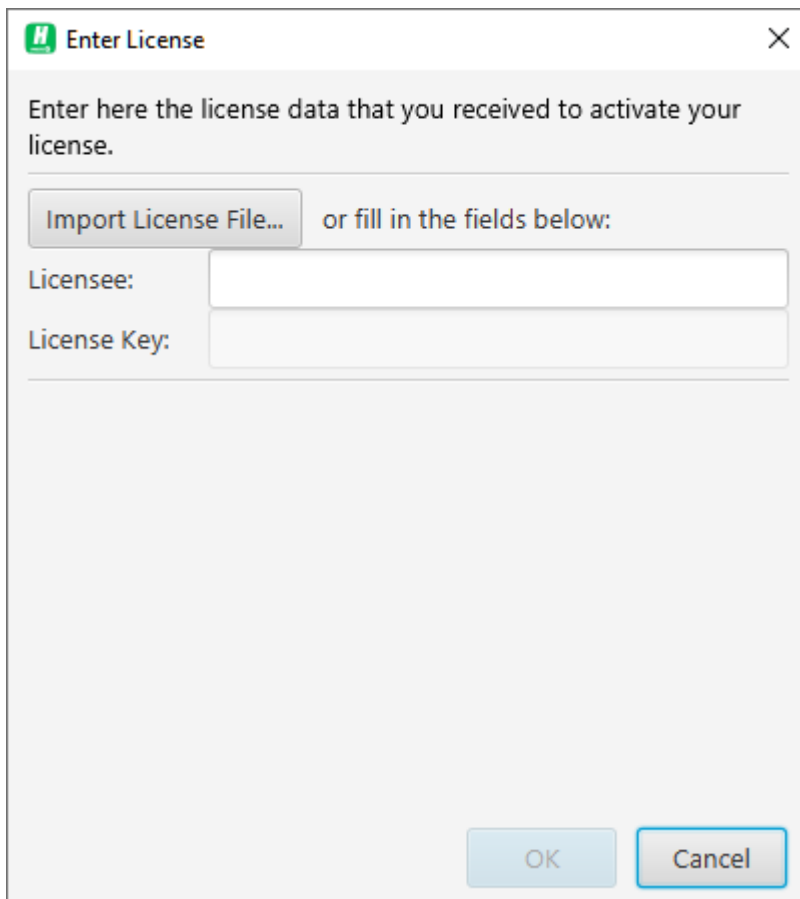


Figure 9. Enter License Dialog.

When your purchase has been processed, you will get a License File, which you can import in the Enter License Dialog (Figure 9, “Enter License Dialog.”, also available from the Help menu) to activate the product.

6. Configuring Source Servers

Once you have the Historian running, you can start configuring the data sources for it. Remember that Historian is acting as an OPC UA Client towards the Source Servers, so the connection configuration is performed respectively.

The Source Servers view shows all existing Source Servers and their current connection status (Figure 10, “Source Servers.”).

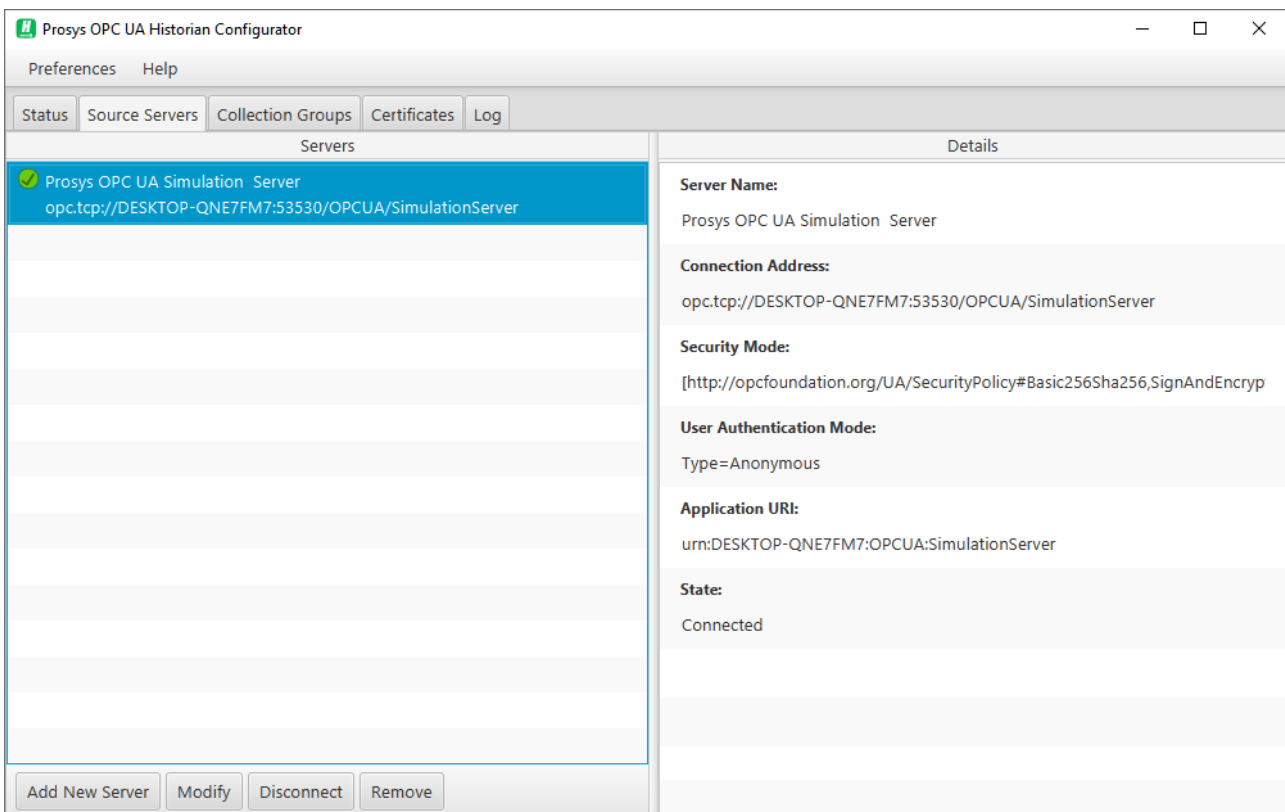


Figure 10. Source Servers.

Use the **Add** button (at the bottom left of the window) to add a new Source Server (Figure 11, “Add new Source Server Dialog.”).

You must provide a Connection Address and select a Security Mode for the connection. After these two settings are configured, the connection is automatically verified. If the connection is successfully accomplished, the Server Name and Server Application URI are automatically fetched from the server. You can modify the Server Name if you like, but it must be unique in the Historian configuration, since it is used for identification purposes in the UI and also in the OPC UA Server Address Space. Server Application URI is used to identify the server internally. It is defined by the Source Server application itself and cannot be changed.



You can test the Source Server connections using Prosystech OPC UA Simulation Server, which is available from <https://www.prosysopc.com/products/opc-ua-simulation-server/>

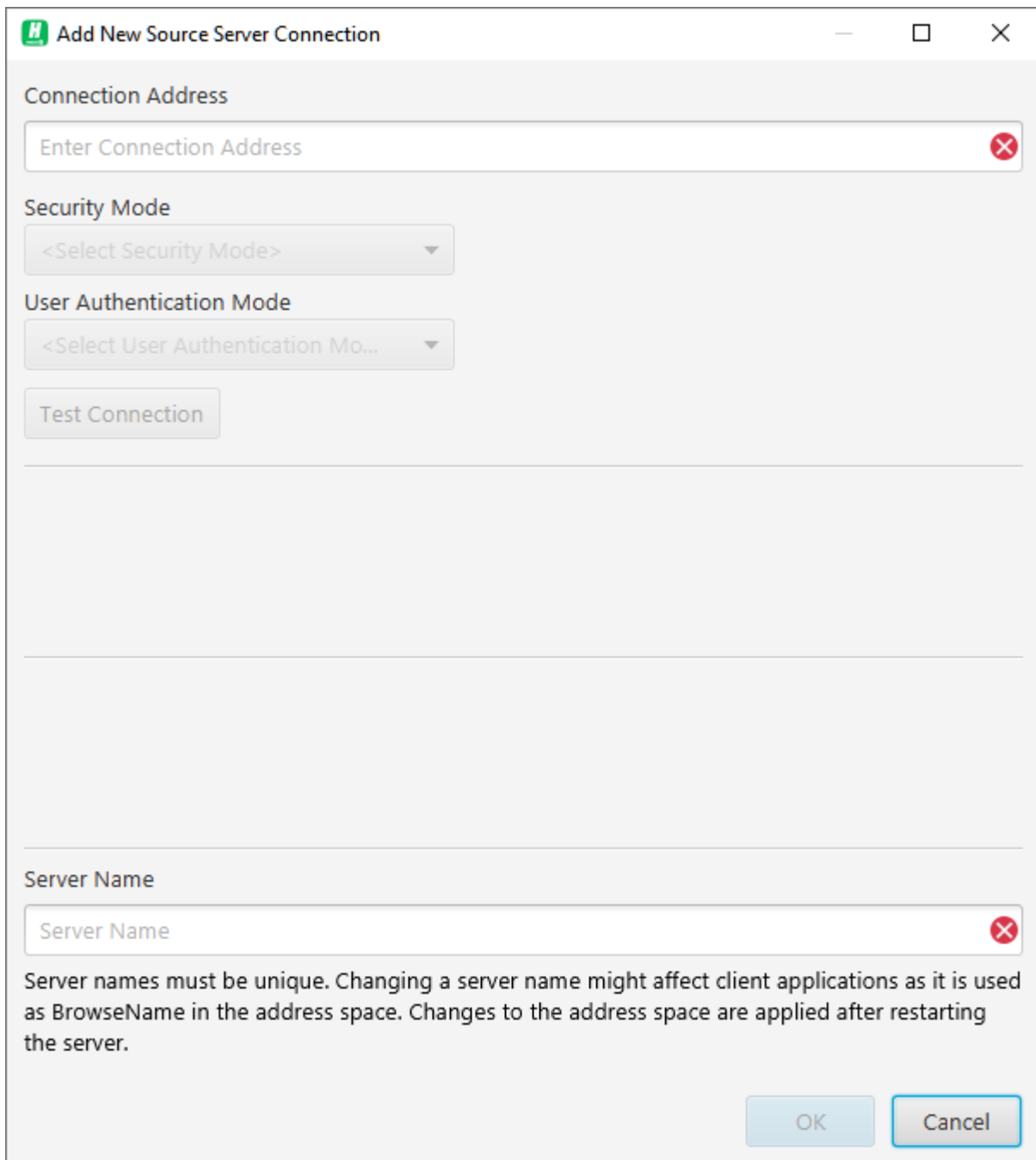


Figure 11. Add new Source Server Dialog.

The connection address normally contains `opc.tcp://`, which defines the connection to be accomplished with the UA TCP binary protocol. The exact address can usually be copied from the configuration of the Source Server itself. For example, for the Prosystech OPC UA Simulation Server, the address is typically `opc.tcp://<hostname>:53530/OPCUA/SimulationServer`, where `<hostname>` is the actual name of the computer in which the server is running.

If you select a secure connection, you must ensure first that the server trusts (the certificate of) the Historian. Otherwise, you will see a `Bad_SecurityChecksFailed` error (Figure 12, “Source Server certificate is not trusted.”). Follow the instructions of the server, to find out how to make it trust the Historian.

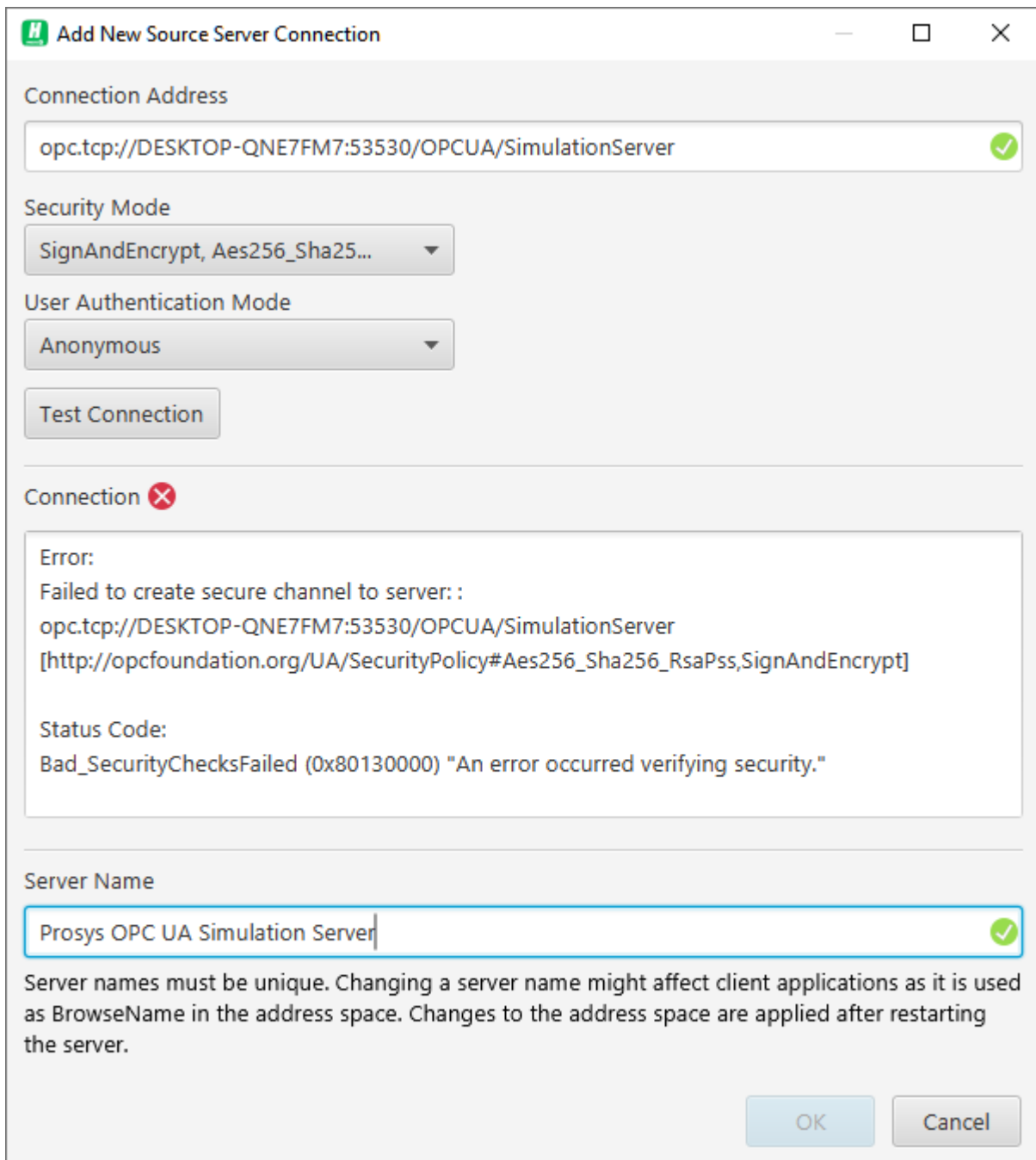


Figure 12. Source Server certificate is not trusted.

Once the Source Server trusts the Historian, you will see the main information of the Source Server's certificate in the dialog (Figure 13, "Source Server certificate is trusted."). Validate that the server is the one that you want to connect. Click 'OK' to confirm the addition of the new Source Server connection.

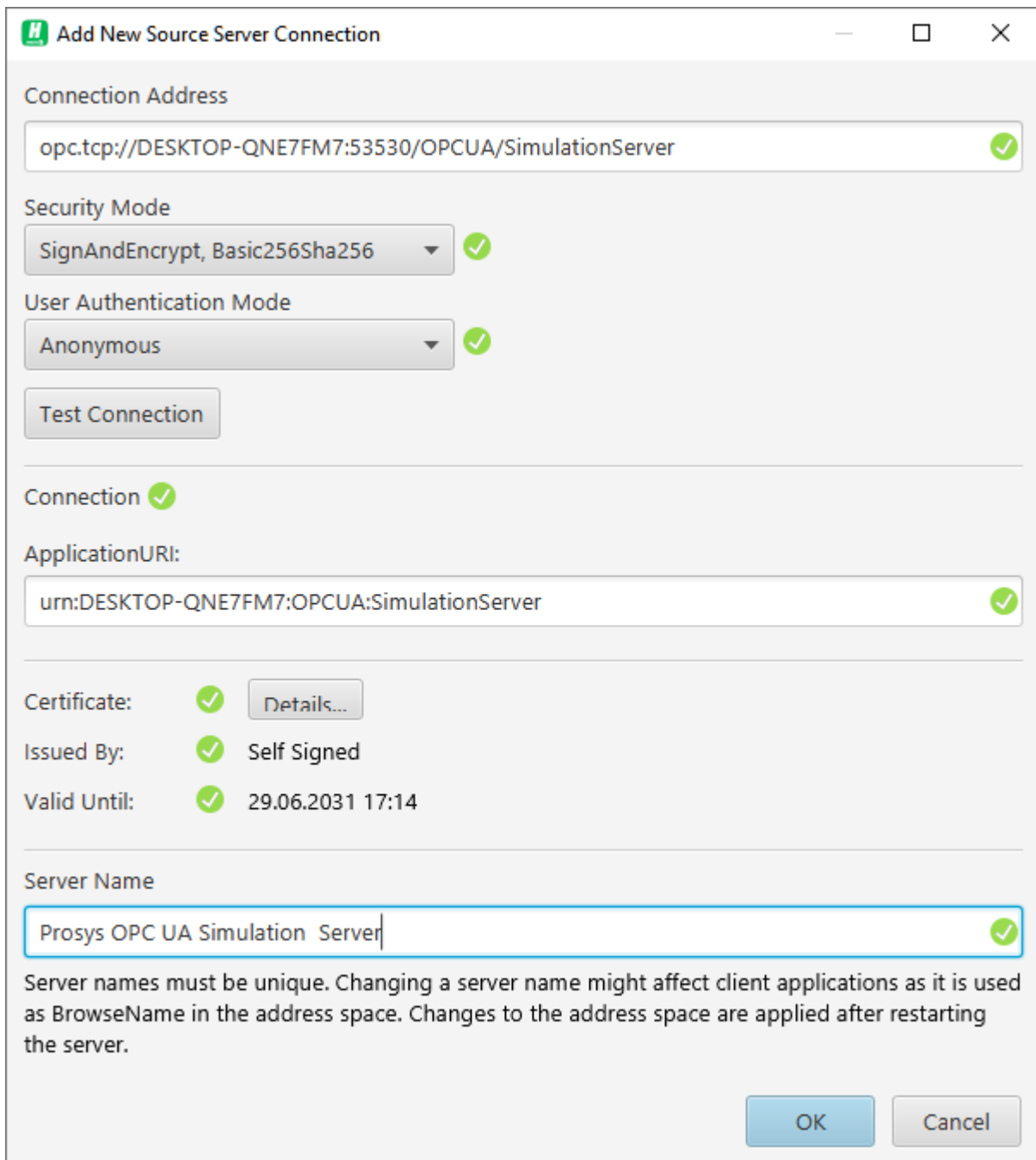


Figure 13. Source Server certificate is trusted.

You can disconnect from the Source Server by selecting the server pressing the **Disconnect** button. When the Source Server is disconnected it does not collect data from the server's items.

Also, you can modify or remove the Source Server by pressing the respectively buttons.

7. Configuring Data Collection

Once you have defined Source Server connections, you can start creating Collection Groups by pressing the **Add New Group** button (Figure 14, "Collection Group."). With the Collection Groups, you can organize your items effortlessly. Also, you can define group recording intervals in milliseconds (Figure 15, "New Collection Group.").

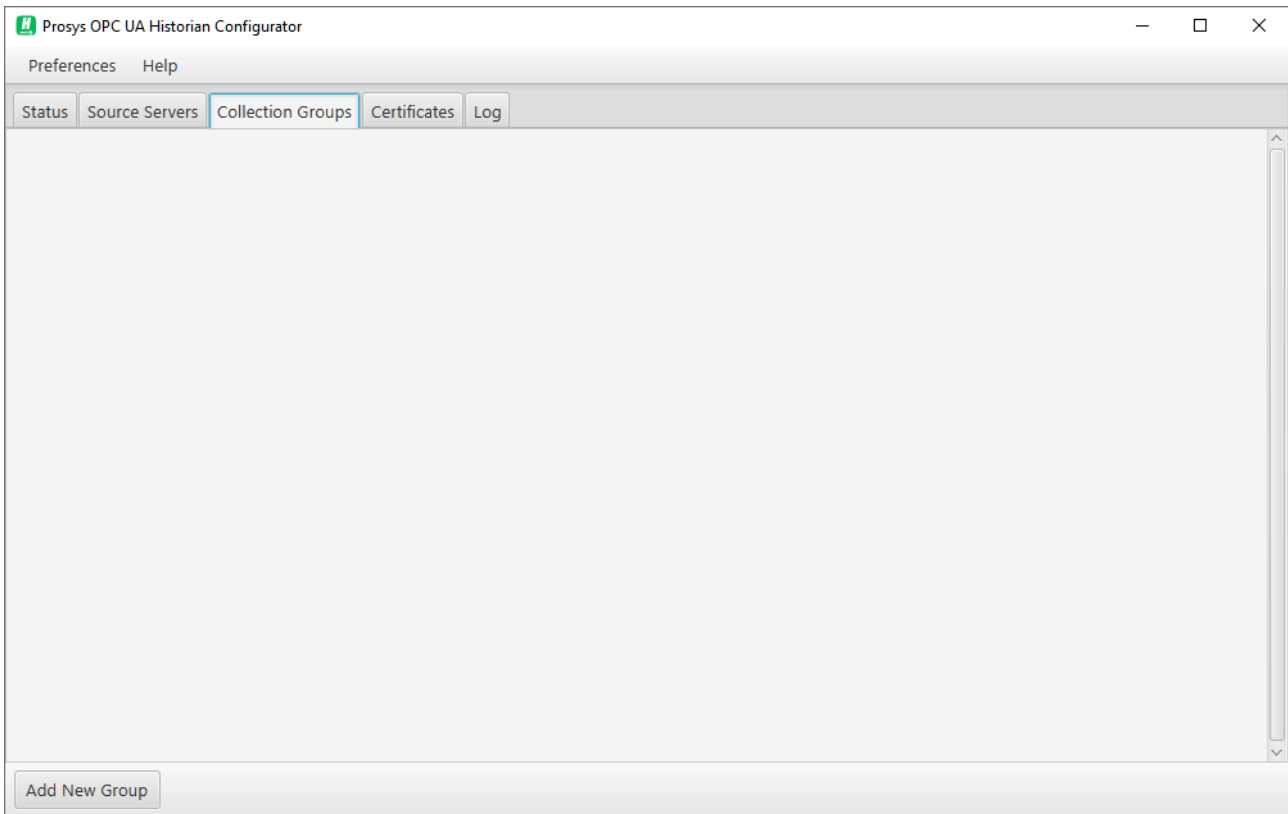


Figure 14. Collection Group.

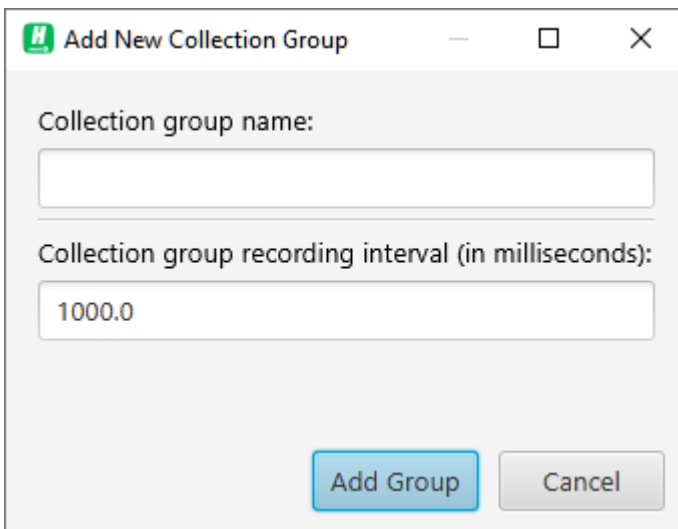


Figure 15. New Collection Group.

The minimum collection recording interval is 1000 ms. Every collection group can have different recording intervals.

After creating at least one Collection Group you can start adding Collection Items from the Source Servers (Figure 16, "Empty Collection Group."). Then, you need to expand the desired Collection Group and press the + button, which opens the Add Items dialog (Figure 17, "Add Items Dialog."). In the dialog, you can select desired items under the Source Servers and finally press the Add button.

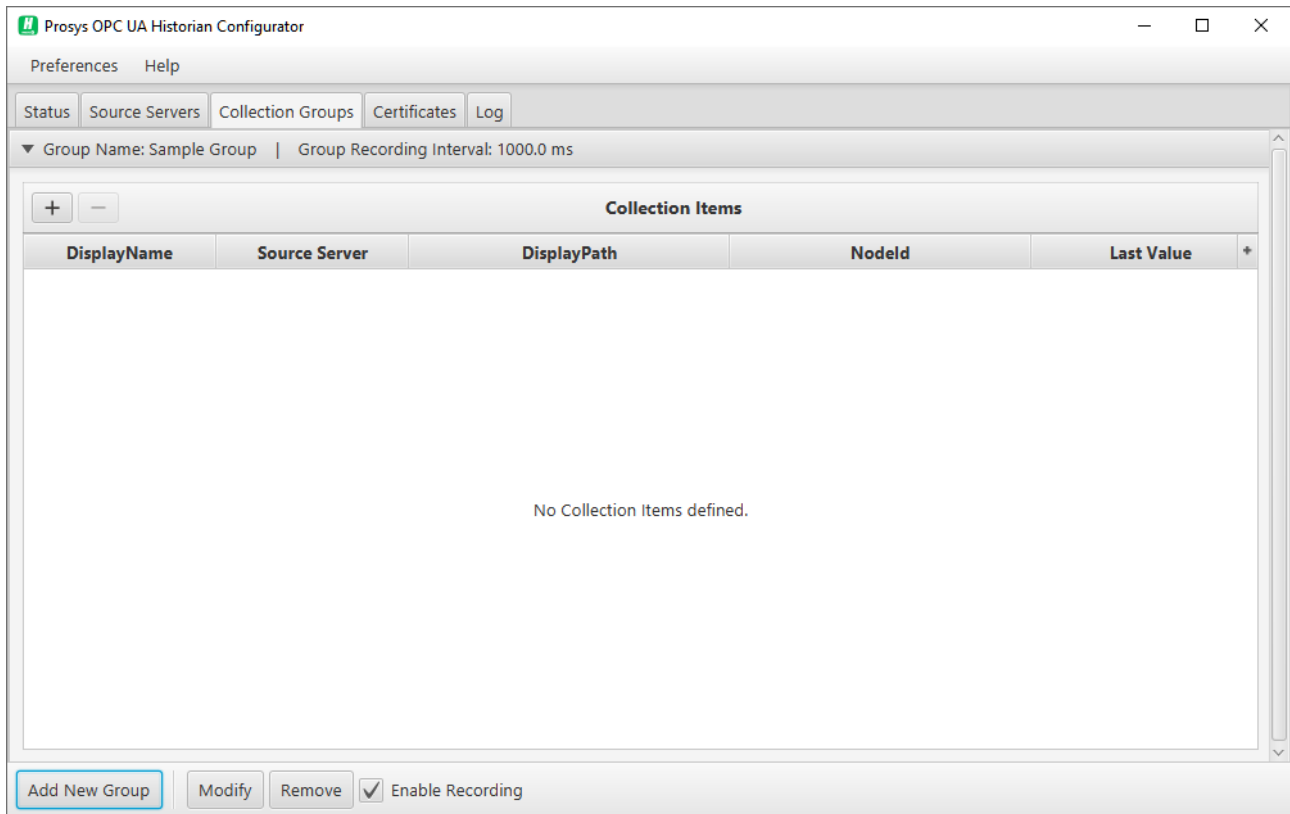


Figure 16. Empty Collection Group.

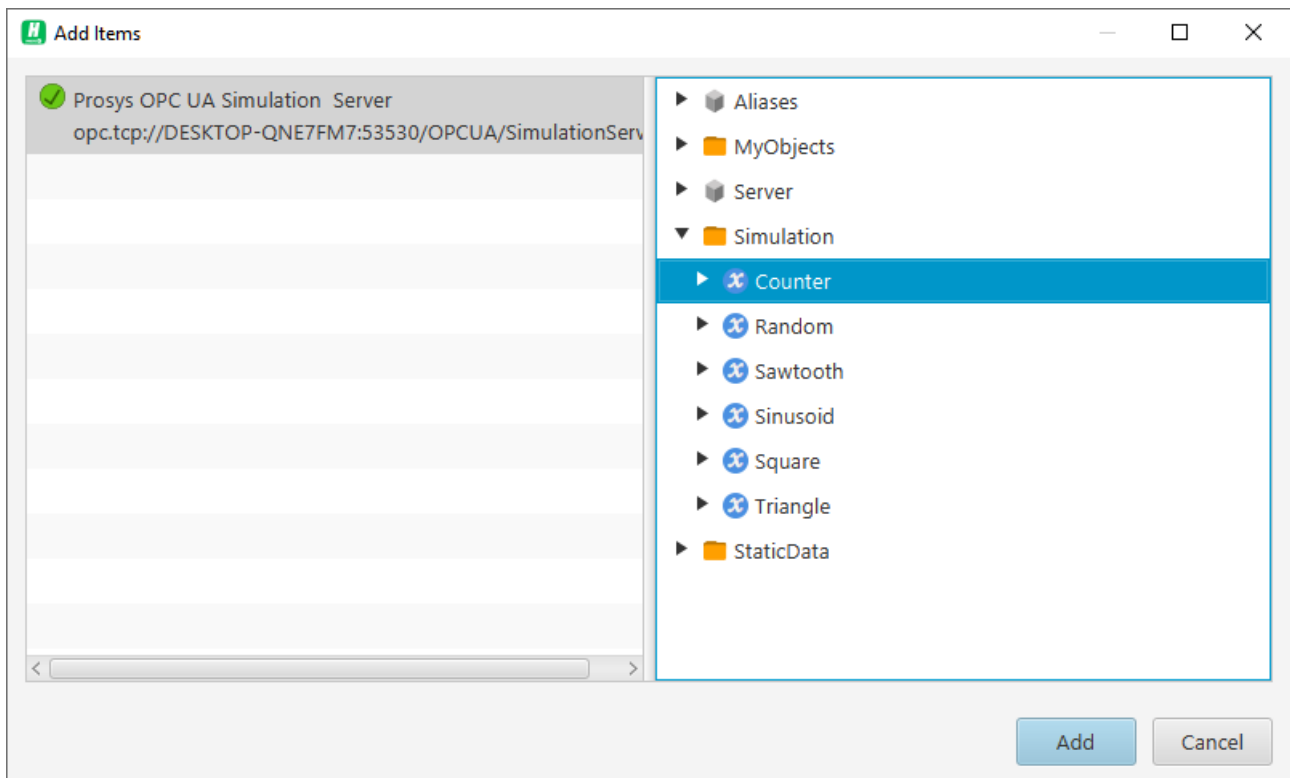


Figure 17. Add Items Dialog.

Now you have added the Collection Items into the Collection Group (Figure 18, "Add Items Dialog."). You can remove items by selecting them and pressing the - button. Removing an item stops data collection for it, but does not remove it or its history from the database. The item also remains accessible for OPC UA client applications through the OPC UA interface.

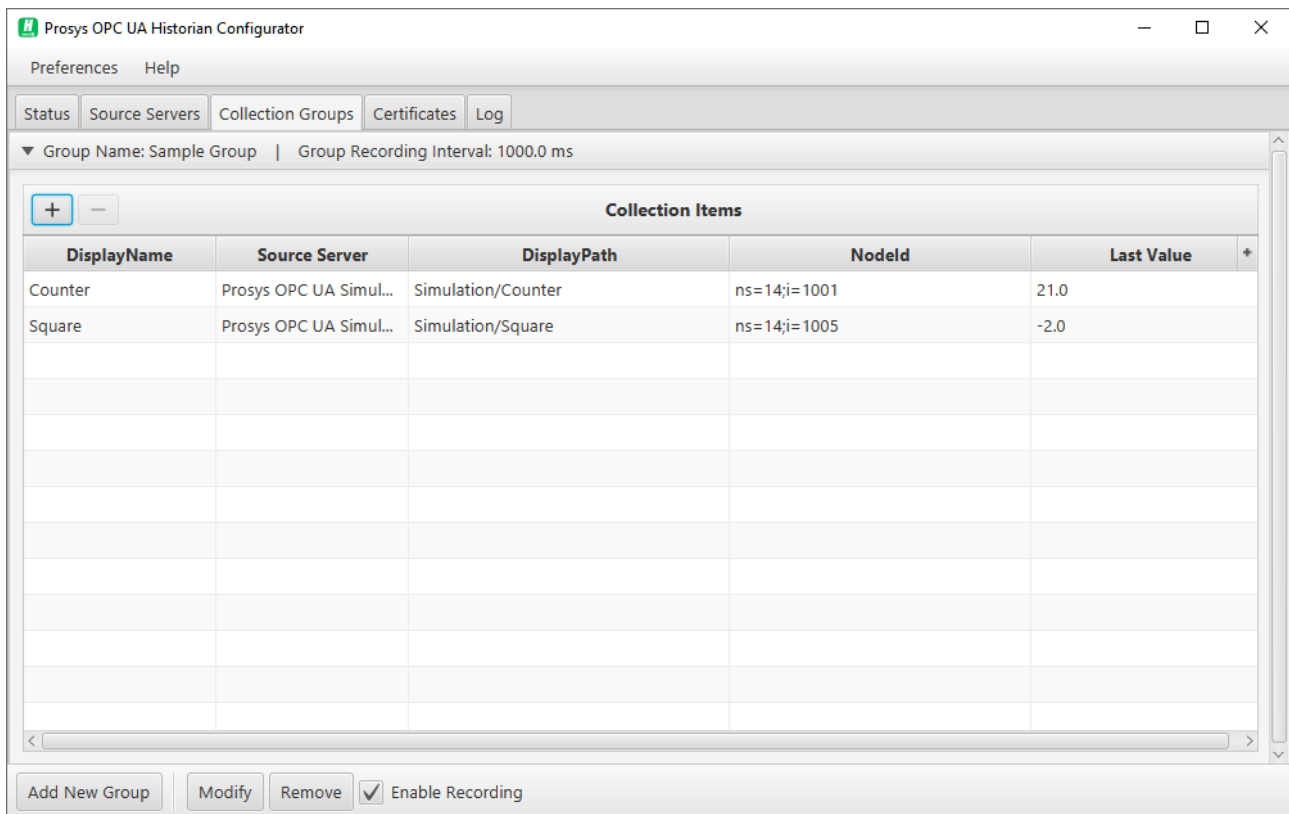


Figure 18. Add Items Dialog.



You can enable or disable recording for individual Collection Group.

History is collected via OPC UA subscriptions. The minimum collection recording interval was 1000 ms. This means that changes that are occurring faster than 1 second cannot be stored. On the other hand, if the items are changing slower, only the samples corresponding to the actual changes are stored in the database.

Historian supports the following datatypes:

- String
- ExpandedNodeId
- QualifiedName
- LocalizedText
- XmlElement
- Boolean
- SByte
- Byte
- Int16
- UInt16
- Int32
- UInt32
- Int64
- UInt64
- Integer

- UInteger
- Float
- Double

8. Connecting to Historian from an OPC UA Client Application

Whenever the Historian is running, you can connect to it using any OPC UA client application (Figure 19, “Prosys OPC UA Browser connected to Historian.”). The client applications can access current data from the Source Servers and also the history data for the variables that have been configured for history data collection in the Historian.



You can examine the data that is collected into the Historian using Prosys OPC UA Browser, which is available from <https://www.prosysopc.com/products/opc-ua-browser/>

The connection address to the Historian is seen on the Status page – and more detailed on the Section 3.2, “Server Endpoints Configuration”.

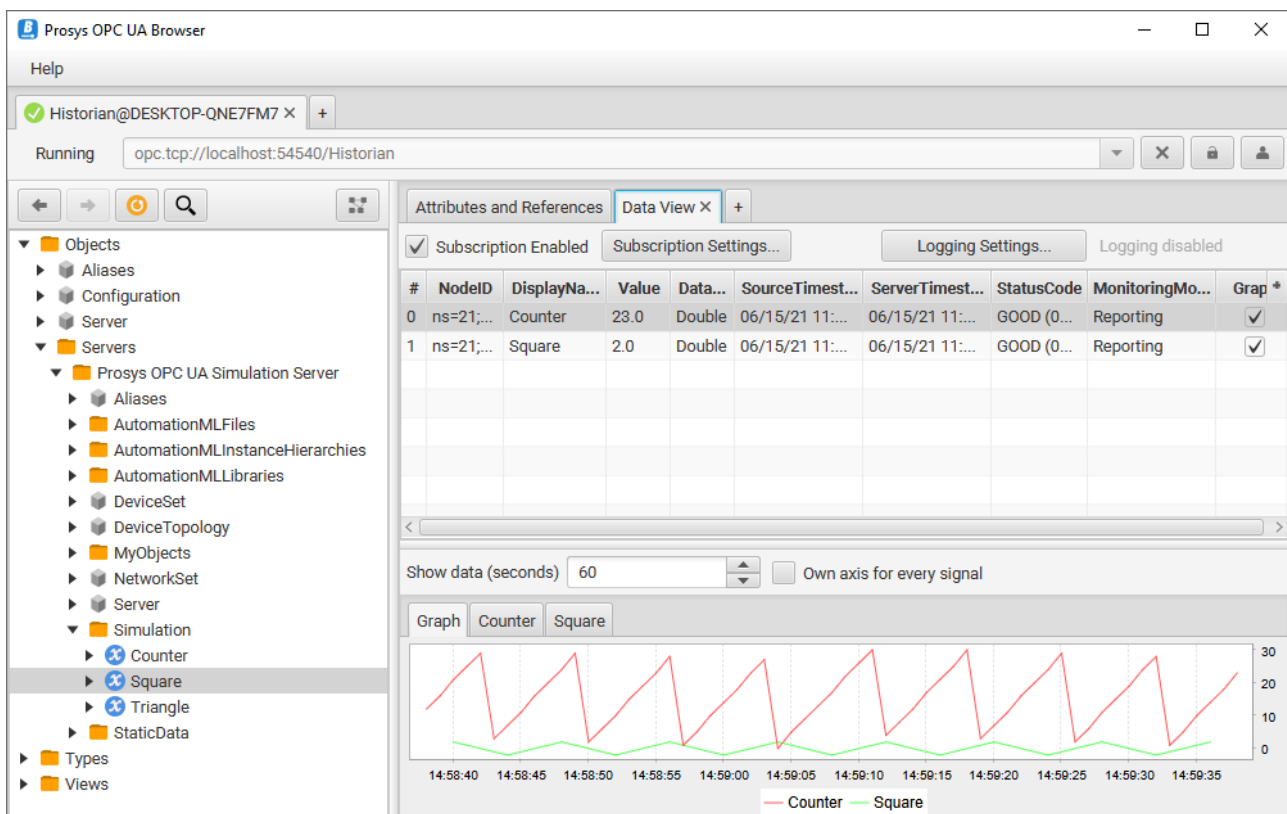


Figure 19. Prosys OPC UA Browser connected to Historian.

By default, the OPC UA client reads raw values from the server. Historian supports reading processed history in the form of Aggregates (OPC UA Specification - Part 13 Aggregates). For example, you can select **Average** aggregate from Prosys OPC UA Browser and read average values from the historical data (Figure 20, “Prosys OPC UA Browser connected to Historian.”).

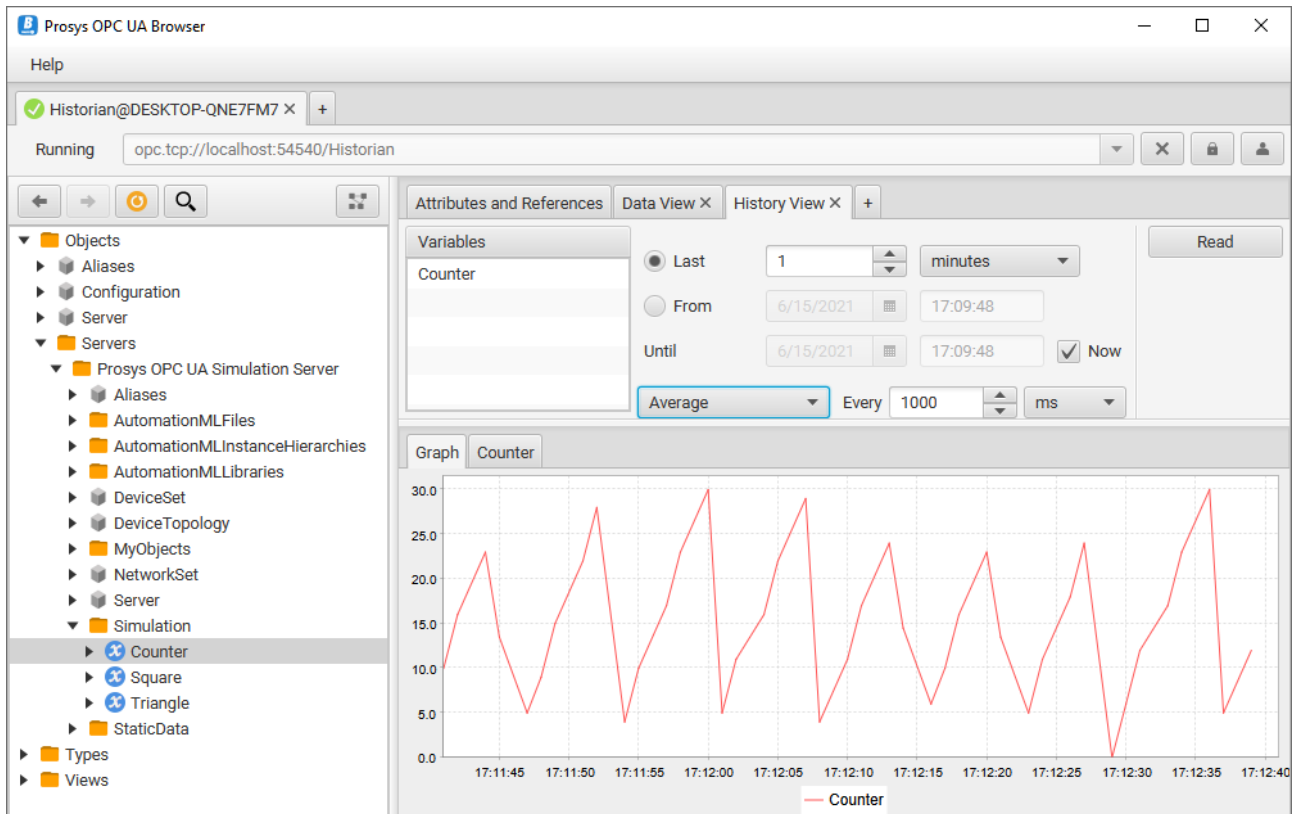


Figure 20. Prosys OPC UA Browser connected to Historian.

9. Secure Connections

Secure connections require that the applications authenticate and trust each other. This is accomplished with the Application Instance Certificates. In order to let client applications connect to Historian securely, it must trust the certificates of those applications. By default, when applications connect, the connection is rejected with OPC UA error code: **Bad_SecurityChecksFailed**. In order to let a client application connect, you must go to the Certificates page and Trust the certificate of the client (Figure 21, "Accepting OPC UA client application certificate.").

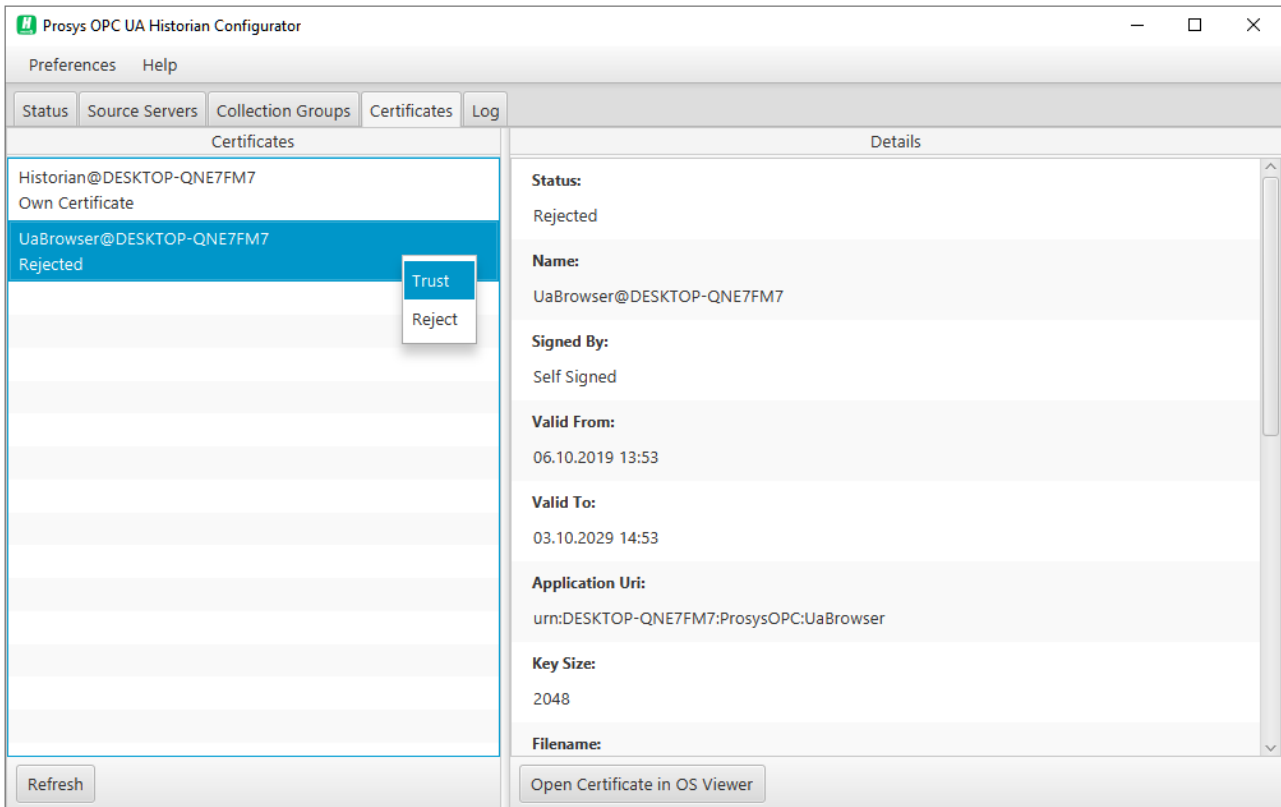


Figure 21. Accepting OPC UA client application certificate.

10. Configuration Folder

The configuration files for the Prosystech OPC UA Historian are stored in different locations, depending on operating environment.

The database connection is defined in [DatabaseSettings.xml](#). All other configuration parameters are stored in the database.

The certificates are stored in sub-directories under the settings directory. There is a specific directory for the application's own keys, for client certificates and for server certificates.

You can reset the complete configuration by removing the settings directory.



You should not normally need to modify the settings manually, since everything can be configured with the Configurator.

10.1. Windows Configuration Folder

On Windows, the configuration folder is located in [%ProgramData%\ProsystechOPC\Historian](#). Note that [%ProgramData%](#) is normally directed to [C:\ProgramData](#), which is a hidden directory by default. If you write the path to Windows Explorer, you can still open the directory. Alternatively, you change your Explorer settings so that it displays hidden folders.

10.2. Linux Configuration Folder

If you installed the application to your Linux from the Debian or RPM Installation Package, the configuration files are located in [/etc/ProsystechOPC/Historian](#)

11. History Database Format

The database stores the history data in **valuehistory** table. It contains the following columns ([Table 1, "SQL table of valuehistory"](#)):

Table 1. SQL table of valuehistory

Column name	Type	Notes
id	BIGINT	Primary key
collectionItemId	INT	Reference to collectionitem
timestap	DATETIME(6)	SourceTimestamp of the value or ServerTimestamp if SourceTimestamp is not available
statusCode	INT	Status Code of the value (OPC UA defined unsigned integer stored as raw signed integer bytes)
value	VARCHAR(4000)	Value

In addition, the table has an index of **timestap** and **collectionItemId** for faster searching. **collectionItemId** is a reference to the **collectionitem** table.

The **collectionitem** table has the following columns ([Table 2, "SQL table of collectionitem"](#)):

Table 2. SQL table of collectionitem

Column name	Type	Notes
collectionItemId	BIGINT	Primary key
datatype	VARCHAR(255)	DataType of the item (NodeId as a string)
name	VARCHAR(255)	Display name of the item
nodeId	VARCHAR(255)	NodeId (OPC UA NodeId) from which history is collected (in Historian namespace)
removed	BIT	1 if removed, 0 if not
collectionGroupId	BIGINT	Reference to a collection group
namespaceId	BIGINT	Reference to a namespace
sourceServerId	BIGINT	Reference to a source server

The **nodeId** column defines the OPC UA NodeId from which history is collected. This is in Historian server namespace (i.e. the namespace indexes are the ones used by the Historian). If you wish to know the original (Source Server) NodeId, see the referred **namespaceId** table entry.



The database structure may change at future updates of the product. So, if you define SQL queries to the database, make sure to validate them after any updates.