

# EVENT LOG EXPLORER ELODEA USER'S GUIDE

HOW TO SETUP, CONFIGURE AND USE ELODEA

## Table of Contents

1. Introduction .....	2
2. Understanding Event Log Explorer Elodea.....	3
2.1. Deployment approaches.....	4
2.2. Feeds and subscriptions.....	7
3. System requirements .....	10
4. Installation .....	11
5. Uninstall Event Log Explorer Elodea .....	13
6. Configuring Event Log Explorer Elodea.....	14
6.1. Configuring Elodea Event Collector Service.....	15
6.2. Configuring Database Connection .....	16
6.3. Configuring SMTP Connection .....	19
6.4. Configuring feeds and subscriptions.....	20
6.5. Managing database objects .....	26
6.6. Managing credentials.....	27
6.7. Extra options.....	29
7. Building XML queries.....	30
8. Viewing event tables.....	33
9. Feed table format .....	34
10. Files and folders .....	35
11. Elodea Event Collector Log.....	37

## 1. Introduction

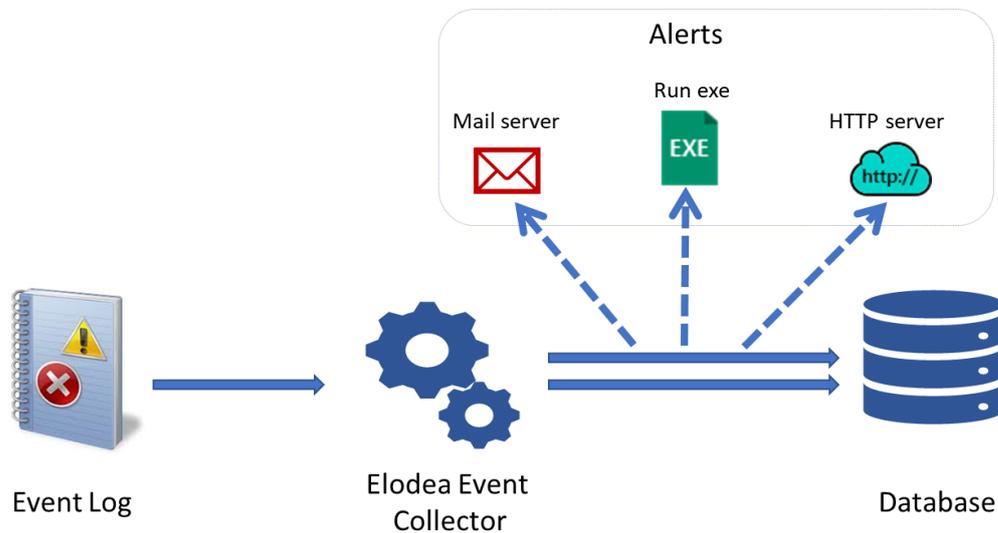
**Event Log Explorer Elodea** lets you get all activity across your network and respond to threats and incidents.

Elodea monitors network events in real time, dispatches them into a database and notifies you when an incident (important event) occurs. Your own rules let you collect only relevant events.

This guide helps you to install, configure and use Elodea in your environment.

## 2. Understanding Event Log Explorer Elodea

The central component of Elodea is **Event Collector**. It runs as a Windows service named "Elodea Event Collector Service" (ElodeaEventCollectorService).



Event Collector receives events from Windows event logs, processes them and stores them into SQL Server database. For specific events, it can send a notification message (alert).

Event Collector can collect events from different workstation and servers concurrently. It can be configured to collect only specific events and ignore all the others. You can configure it as an independent software or use it along with Windows Event Forwarding service.

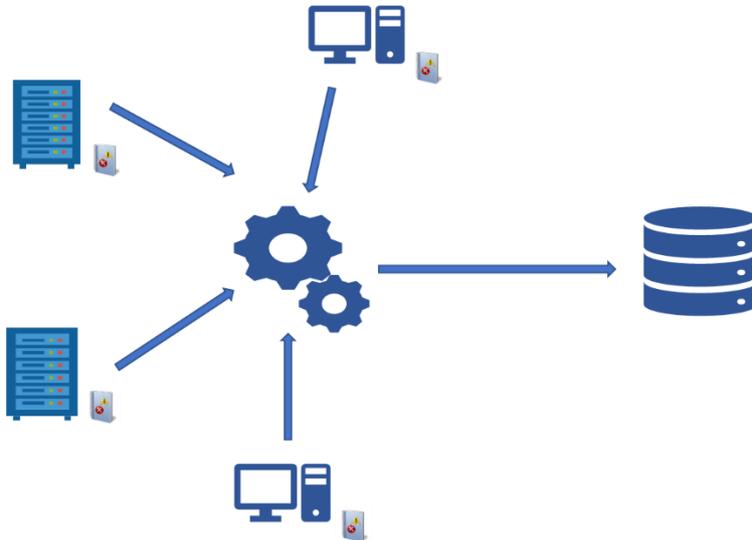
Database is a SQL Server database that receives events from Event Collector and keeps them in the SQL tables. The events are stored in a clear well-documented form and you can use Event Log Explorer or third-party tools to analyze events or generate reports on events.

Elodea can alert on specific events by sending email message, running a program or posting an HTTP request.

## 2.1. Deployment approaches

You can install one or several Event Collectors in your network. Here are some possible deployment options.

### Single-collector deployment



With single-collector approach you install one collector which connects all the event sources and forwards them into the database. Optionally, to reduce network traffic, you can install the collector directly on the database server.

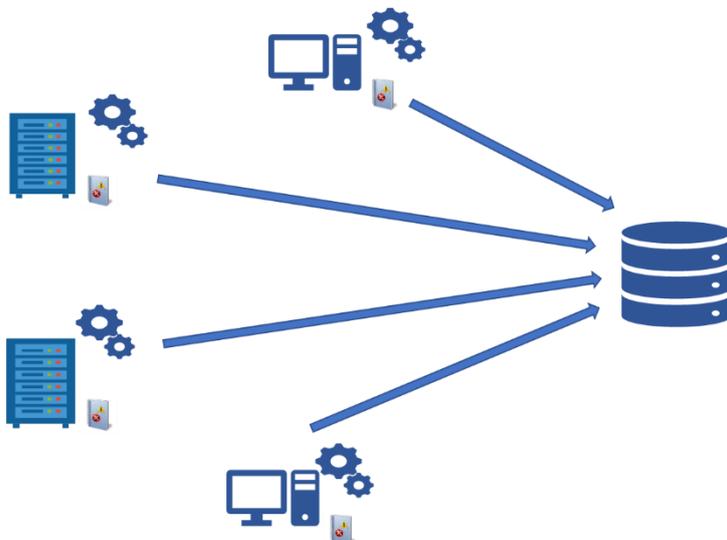
Pros: you setup and manage only one collector.

Cons:

higher requirements to the hardware;

the whole event processing depends on the only one service in the network.

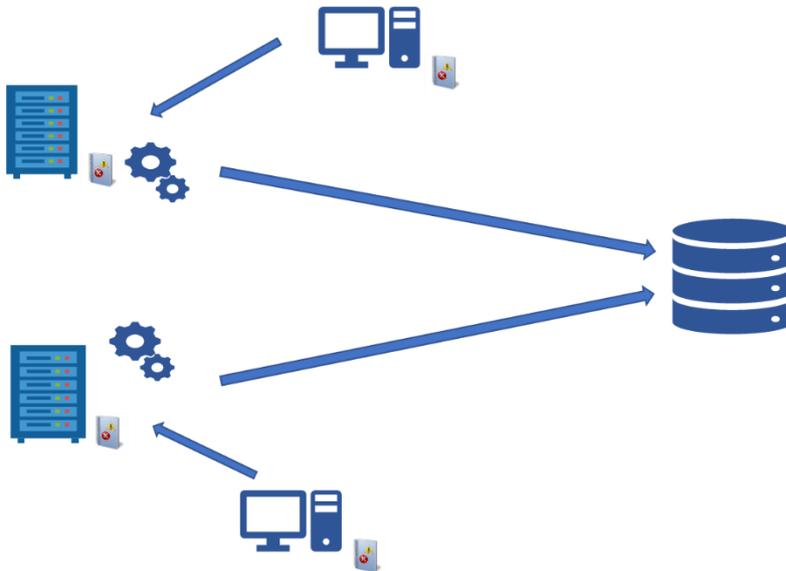
### Multi-collector deployment



Multi-collector approach requires to setup event collector on each event source, the collectors gather events locally and forward them directly to the database.

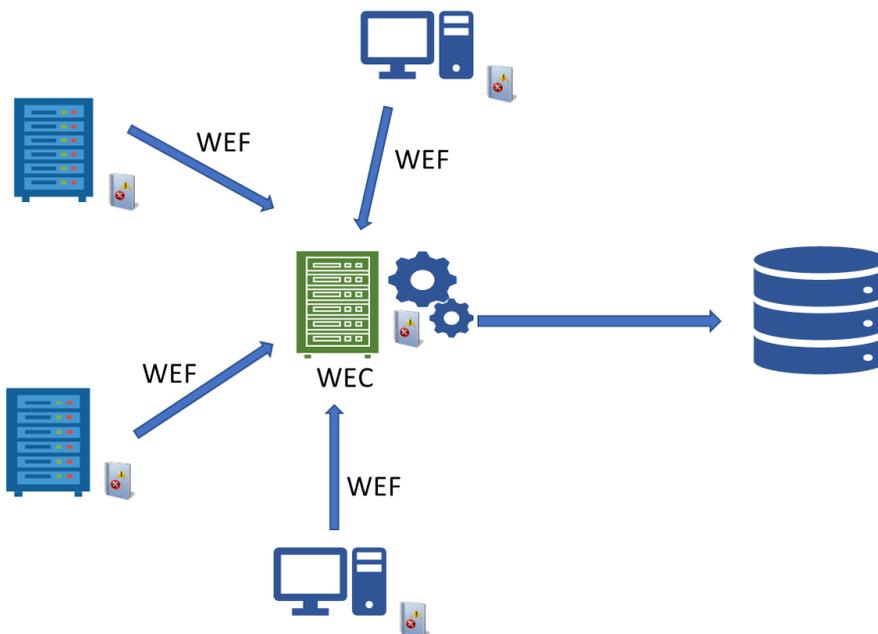
Pros: very reliable;  
no needs to use credential manager or setup special service user;  
Cons: hard to manage in large networks.

### Balanced deployment



Balanced approach combines single-collector and multi-collector approaches. You can select a cluster of computers served with one collector.

### Windows Event Forwarding integration



This approach lets you integrate Elodea into Windows Event Forwarding infrastructure.

Windows Event Forwarding (WEF) is a feature of Windows OS to collect events from different event sources into one computer running Windows Event Collector service (WEC) in one event

log (**Forwarded Events** by default). You can setup Elodea Event Collector to collect events from this Forwarded Events event log and forward them to the database.

With this approach you can install Elodea Event Collector directly on WEC computer.

Pros: if WEF already in use, it's the easiest way to setup Elodea with it;  
Event description is commonly rendered by Windows and saved as text in the Forwarded Events, so Elodea Event Collector may not render descriptions which is time-consuming;  
Better Windows integration and independence on Elodea or other event log managers;  
Can collect Windows 2003 and Windows XP events.

Cons: if WEF not configured, it may require more efforts to set it up rather than configuring Elodea.

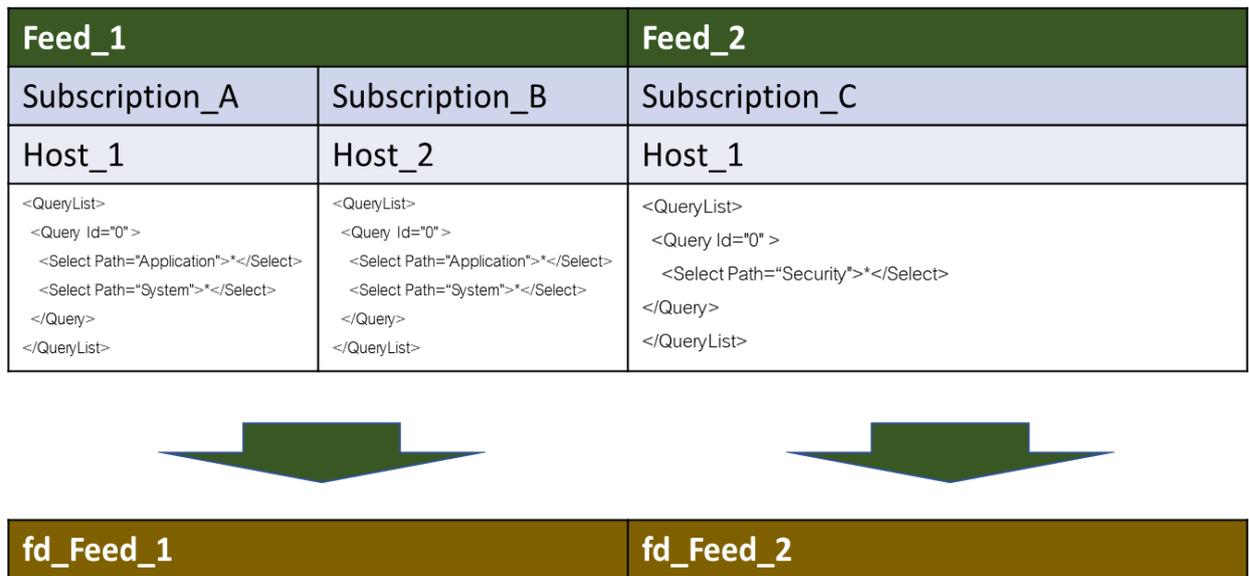
## 2.2. Feeds and subscriptions

A core thing of Elodea is a **feed**.

**Feed** is a special Elodea entity that describes what events and from what sources Elodea will collect and dispatch into the database table. From the database point of view, feed is an SQL **table** with the collected events. Elodea feeds can be **Active** and **Inactive**. If a feed is not active, Event Collector will ignore it and will not collect events.

One feed consists of one or more **subscriptions**.

**Subscription** is a special request to one host to receive events from this host. Subscription uses XML query to receive events.



This picture displays how feeds and subscriptions work.

There are 2 feeds (Feed\_1 and Feed\_2) that maps into 2 SQL tables (fd\_Feed\_1 and fd\_Feed\_2). Elodea creates these tables automatically.

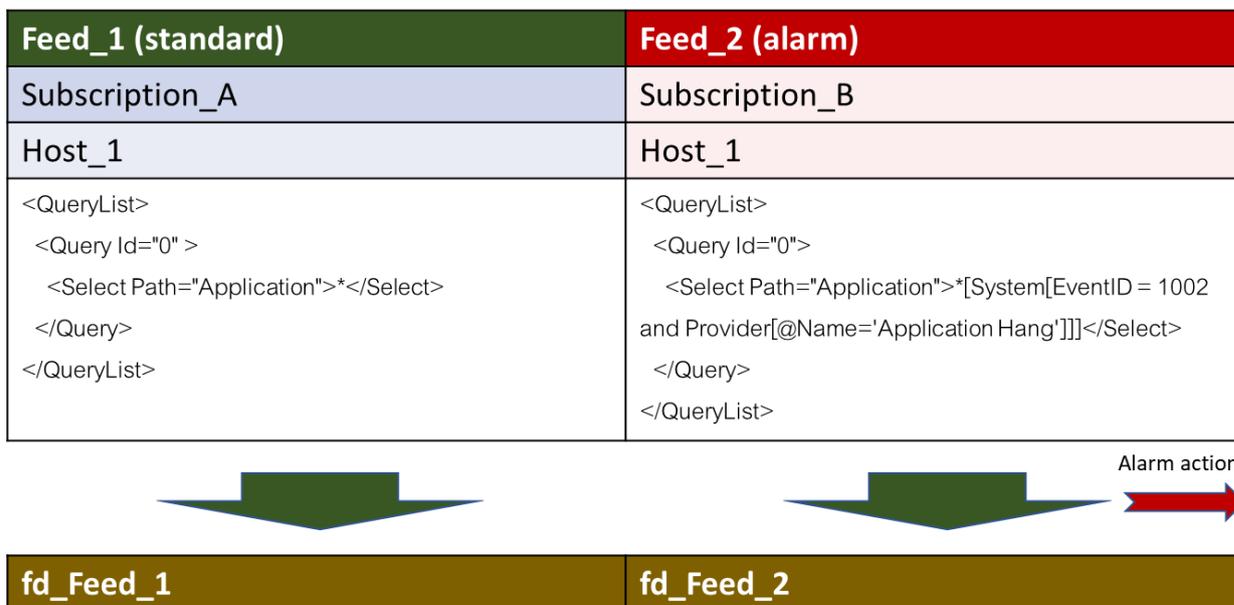
Feed\_1 collects Application and System events from Host\_1 and Application and System events from Host\_2. Feed\_2 collects Security events from Host\_1.

### Feed types

There are 2 types of feeds in Elodea. **Standard feeds** are intended only to collect events and forward them into the database. In addition to the standard behavior, **Alarm feeds** execute an action when an event is forwarded into the database.

#### *Example:*

You want to save all Application events and get notification on Application Hang (event id: 1002) events.



In this you should create 2 feeds. Standard feed contains a subscription which gets all application events:

```
<QueryList>
  <Query Id="0" >
    <Select Path="Application">*</Select>
  </Query>
</QueryList>
```

Alarm feed gets only specific events:

```
<QueryList>
  <Query Id="0">
    <Select Path="Application">*[System[EventID = 1002 and
Provider[@Name='Application Hang']]</Select>
  </Query>
</QueryList>
```

Elodea will create two database tables for these feeds: fd\_Feed\_1 and fd\_Feed\_2.

When an event occurs in the Application log, Elodea will save this event in the fd\_Feed\_1. When a specific event (Event ID = 1002, Event Provider (Source) = "Application Hang") occurs in the Application log, this event will be saved into fd\_Feed\_1 and into fd\_Feed\_2. Elodea will also run a specific action specified in Feed\_2.

## Subscription models

There are 2 subscription models in Elodea.

**Push model** – a dedicated thread for each subscription will be created and when the subscription events occur in the log, Elodea will immediately get the event in this thread.

**Poll model** – When a new event appears in the log, Elodea will only get a notice of it, and then it will read subscription event(s) from the log.

We do not recommend creating more than 1000 Push subscriptions in Elodea.

### **Working subscriptions**

If a subscription already dispatched at least one event into the database, it is called **working**. There are limitations on modifying working subscription. More information about modifying subscription is available in **Configuring feeds and subscriptions** chapter.

### 3. System requirements

Elodea Event Collector can be installed on any computer running Windows Server 2008 or higher (Windows 2012, Windows 2016, Windows 2019) or Windows Vista or higher (Windows 7, Windows 8, Windows 10).

Elodea can collect events from computers running Windows Server 2008 or higher or Windows Vista or higher.

To save events into the database it requires Microsoft SQL Server 2008 or higher (SQL Server 2012, 2014, 2016, 2017).

To access event logs from Windows XP or Windows Server 2003, you should use Windows Event Forwarding integration (WEF can forward events from old Windows logs into modern event log which can be accessed by Elodea).

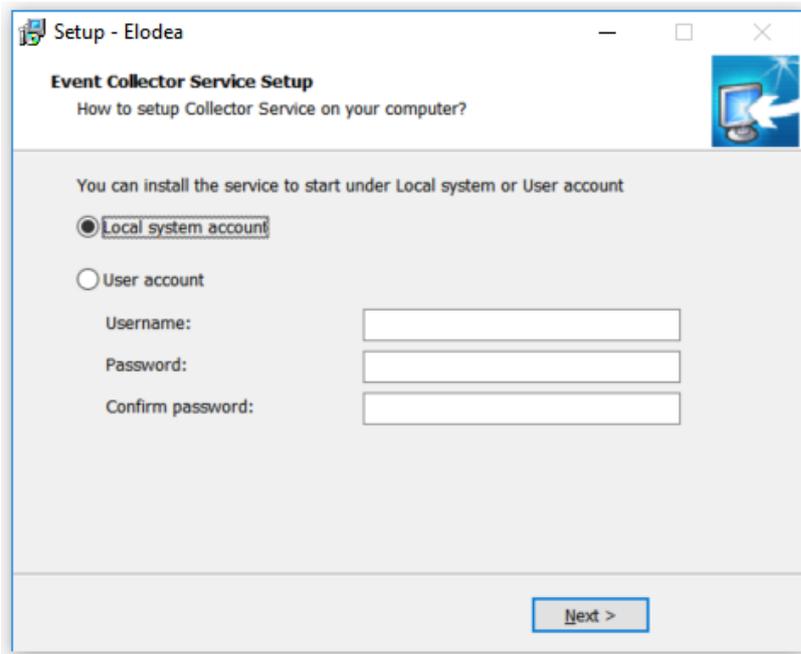
To access event logs stored in the database, it's recommended to use Event Log Explorer 4.7 or higher.

## 4. Installation

To start install process, run `elodea_setup.exe` file. The setup wizard will guide you through the installation process. At the last step of the installation, it sets up Elodea Collection Service.

### Elodea collector service setup

You should select how to setup the service yourself.



You can setup the service to run under Local system account or under a user account.

To run the service under the local system account, select **Local system account** radio button. To run the service under a user account, select **User account** radio button, then enter user credentials – name and password. To specify domain user name, enter it as `DOMAIN\USER`. To specify a local user name, just input it without prefixes or prepend it with `.\`. Note that Services applet always displays local user names prepended with `.\`.

As a rule, domain user account is preferable, however you can use local system account in test environment or when you configured Elodea event collector to receive events from local computer only (multi-collector deployment or WEF integration).

In some cases, using local system account can be prohibited by Group Policies – in this case you can only run the service under a user account.

Running services under a user account requires this account to have right "Logon as a service". Elodea setup adds this permission automatically. If you changed the account name using Windows services applet, you should modify "Logon as a service" policy yourself. You can do it e.g. using Local Security Policy. To configure "Logon as a service", start Local System Policy, browse to Security Settings, Local Policies, User rights assignment. Find **Logon as a service** and double click on it to view/modify it.

**Notes.** If you input a nonexistent or incorrect user name, Elodea setup will warn you about this problem. However, if you specify a wrong password, Elodea setup will not check password validity and setup the service with incorrect password.

It is recommended to add this user account to built-in **Event Log Readers** group.

### **Setting up the service manually**

In some cases, you may want to setup Elodea Collection Service manually. The service can be set up from the command line. You should run **Windows Command Prompt** as administrator (elevated).

To setup the service running under Local System account, run:

```
EventCollector.exe /Install
```

To setup the service running under a user account, run:

```
EventCollector.exe /Install /User <Username> /Password <Password>
```

## 5. Uninstall Event Log Explorer Elodea

To uninstall Event Log Explorer Elodea, just open Programs and Features Control Panel applet, find Elodea and doubleclick on it. Follow the Uninstall wizard instructions.

**Note:** If you setup Collection Service to run under user account and Elodea Setup added "Logon as a service" permission to the user account, Uninstall wizard would not remove this permission from the account. You can review and modify "Logon as a service" accounts in Local Security Policy or in Group Policies. In Local Security Policies navigate to Security Settings -> Local Policies -> User Rights Assignment. A similar path is in Group Policy Editor (Local Computer Policy -> Computer Configuration -> Windows Settings -> Security Settings -> Local Policies -> User Rights Assignment).

To remove "Logon as a service" permission from the user account, open Local Security Policy or Group Policies, browse to User Rights Assignment. Find **Logon as a service** and double click on. Find the account name and click Remove button.

## 6. Configuring Event Log Explorer Elodea

Execute CollectorSettings.exe (or run **Elodea Event Collector Settings** from Windows Start Menu). Hereinafter, it will be referred as **the Application**.

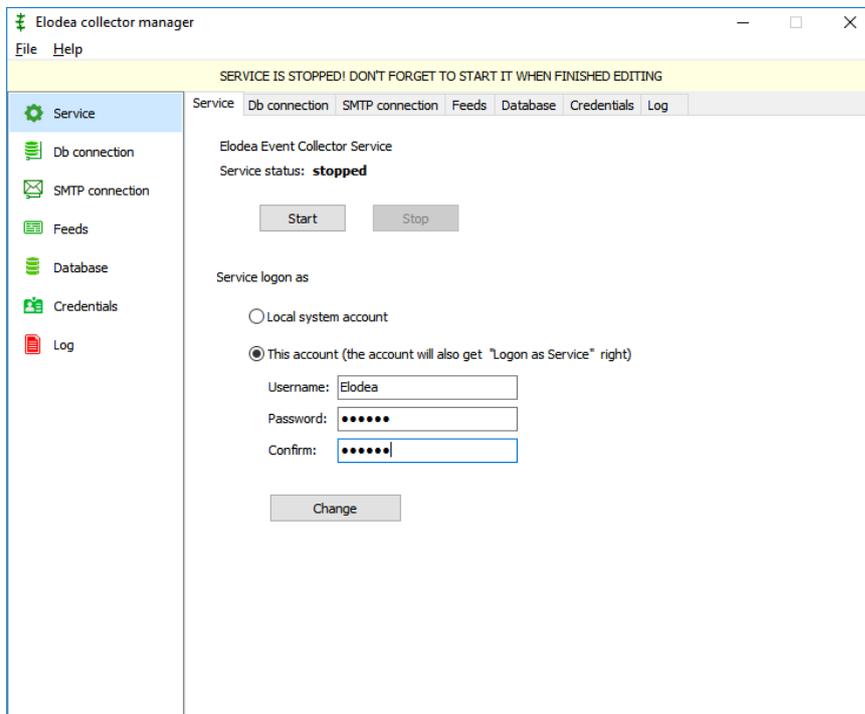
On the Service tab you will see status of the service. It should be stopped. If an error message that the service not exists appeared, close the application and install Elodea Event Collector Service as described in the **Installation** chapter.



***You cannot modify Elodea settings while Event Collector Service is running. Stop the service before changing the settings and do not forget to start it when finished.***

## 6.1. Configuring Elodea Event Collector Service

You can configure Collector Service from the Application. Select Service tab to review/configure the Service.



You can setup the service to run under Local system account or under a user account as described in the **Installation** chapter. If you setup the service to run under a user account, it is recommended to grant this user permissions to read the required event logs.

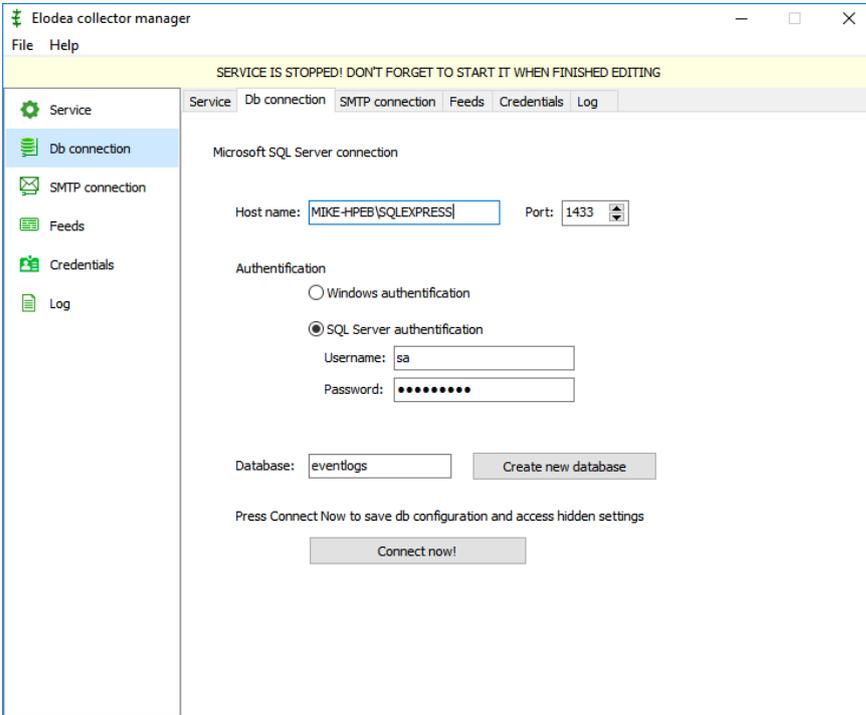
It is not necessary to grant administrator's permissions to this account. It is enough to add this account to **Event Log Readers** security group.

Changing account name automatically adds **Logon as a service** right to the account. You can review and modify "Logon as a service" accounts in Local Security Policies or in Group Policies.

Refer to **Uninstall Elodea** chapter to get more information how to modify "Logon as a service" accounts.

## 6.2. Configuring Database Connection

Elodea requires SQL Server connection to forward events into the database. To set the database connection, click on **Db connection** tab.



The screenshot shows the 'Elodea collector manager' application window. The 'Db connection' tab is selected in the left sidebar. The main area is titled 'Microsoft SQL Server connection'. It contains the following fields and controls:

- Host name:** A text input field containing 'MIKE-HPEB\SQLEXPRESS'.
- Port:** A dropdown menu showing '1433'.
- Authentication:** Two radio buttons: 'Windows authentication' (unselected) and 'SQL Server authentication' (selected).
- Username:** A text input field containing 'sa'.
- Password:** A password input field with masked characters.
- Database:** A text input field containing 'eventlogs'.
- Create new database:** A button next to the database field.
- Connect now!:** A button at the bottom of the configuration area.

A yellow warning banner at the top of the window reads: 'SERVICE IS STOPPED! DON'T FORGET TO START IT WHEN FINISHED EDITING'.

In **Host name** enter the database server name (instance name). For SQL Server, the default instance is the computer name. For SQL Server Express, the default instance is named <COMPUTER\_NAME>\SQLEXPRESS.

Enter SQL Server Port in the **Port** field. The default value is 1433.

In Authentication section choose between Windows or SQL Server authentication by selecting the corresponding radio buttons and type the username and the password if required.

Type the database name in the **Database** field.

If you haven't created the database yet, you can create it from the Application. To create the database, click **Create New Database** button, type server name and your credentials and click Connect button. Then type database name, its file parameters and review or modify database creation script. Click **Create** button to start the creation script.

Click **Connect Now** to test the database connection and get access to extra Elodea settings.



***If you use Windows authentication, pay attention to the fact that the Service will connect the database using the account name specified on the Service tab, but the Application connects the database using your current account.***

To make Elodea correctly write into the database, you may need to grant DBO privileges to the user account specified in the Authentication section. The easiest way to manage SQL Server

database users is to use Microsoft SQL Server Management Studio (available at <https://docs.microsoft.com/sql/ssms/download-sql-server-management-studio-ssms> ).

To create a new SQL Server Login, start Management Studio, connect the server, browse to Security->Logins, press right mouse button and select New Login.

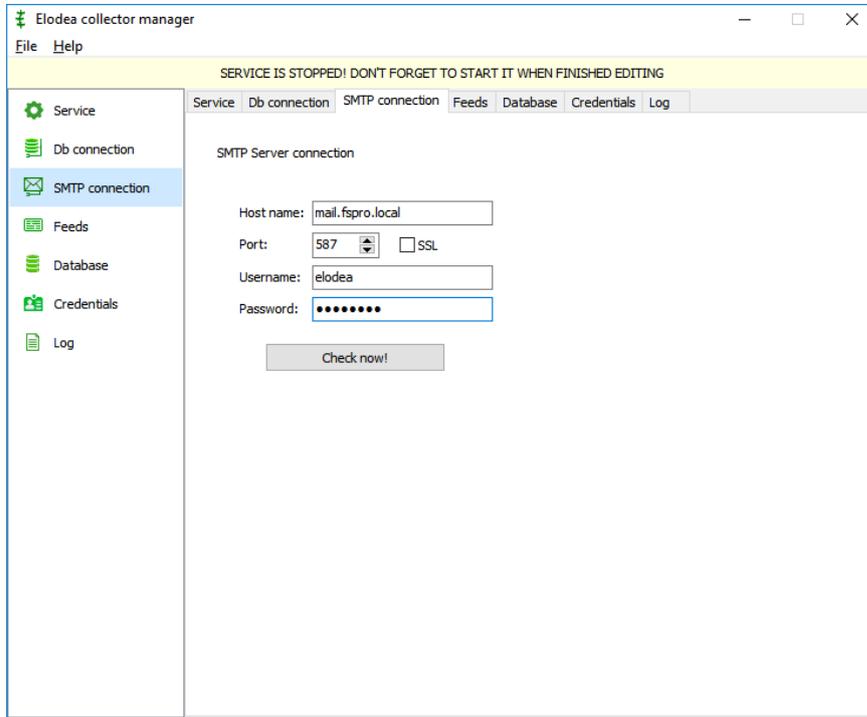
To grant a user DBO privileges, browse to Security->Logins, Double click on the required login name, select User mapping. In Then select the Elodea database, tick **Map** column and change **Default schema** to dbo. Then in **Database role membership** tick db\_owner and press OK.

Depending on your Service Logon as settings and Database authentication mode, you may need to configure different SQL server users.

Database authentication Service Logon as	Windows authentication	SQL Server authentication
User account	<p>The service will connect the database using the specified user account.</p> <p>You should add this user account to the list of SQL Server Logins (if it is not listed yet) and grant <b>db_owner</b> role to this account and map this user account to the <b>dbo</b> role for your database.</p> <p>If you start the Application from the user account that doesn't match your service user account, you should also grant your current user account the same db permissions that you granted to the service account.</p>	<p>Regardless on the Service logon settings, both the Service and the Application will connect the database using specified SQL Server username/password. The specified user must be the database owner. If you use your database server instance solely for Elodea database, you may use <b>sa</b> username.</p> <p><b>This is the easiest way to setup Elodea, but on some MSSQL Server configurations, SQL Server authentication may be disabled. In this case you should either enable SQL Server authentication (SQL Server and Windows Authentication mode) or use Windows authentication in Elodea. To enable SQL Server and Windows Authentication mode, please refer to this article:</b></p>
Local system account	<p>The service will connect the database using Local System account.</p> <p>You should add NT_AUTHORITY\SYSTEM to the list of SQL Server Logins and grant <b>db_owner</b> role to NT_AUTHORITY\SYSTEM and map NT_AUTHORITY\SYSTEM to the <b>dbo</b> Schema for your database.</p> <p><b>This will work only if the service is installed on the same machine with the database server!</b></p>	<p><b>enable SQL Server and Windows Authentication mode, please refer to this article:</b></p> <p><a href="https://docs.microsoft.com/sql/databa-se-engine/configure-windows/change-server-authentication-mode">https://docs.microsoft.com/sql/databa-se-engine/configure-windows/change-server-authentication-mode</a></p>

### 6.3. Configuring SMTP Connection

You should configure SMTP connection only if you want Elodea to send notifications about specific events by email. If you don't plan to receive email alerts, you can skip this step. Click on **SMTP connection** tab to configure it.



**Host name** is a name of the SMTP server. This can be either as a host name or as an IP address.

**Port** is a TCP port of the SMTP server.

**SSL** should be enabled for secure connection to the SMTP server.

**Username** is a user name for mail server.

**Password** – is a user password for mail server.

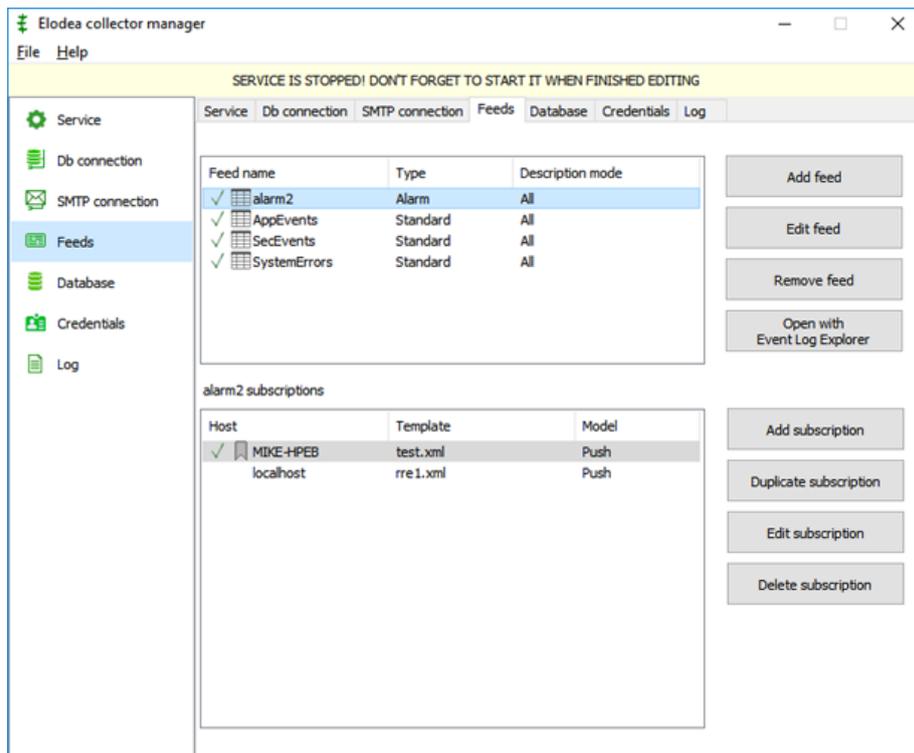
Click **Check Now** to verify connectivity to the SMTP server.

## 6.4. Configuring feeds and subscriptions

Before starting configuring feeds and subscription, make sure that you read chapter **Understanding Elodea, Feeds and Subscriptions**.

To manage feeds and subscriptions click on **Feeds** tab. You can edit feeds and subscriptions only if you connected to the database (pressed **Connect now!** on the Db connection tab).

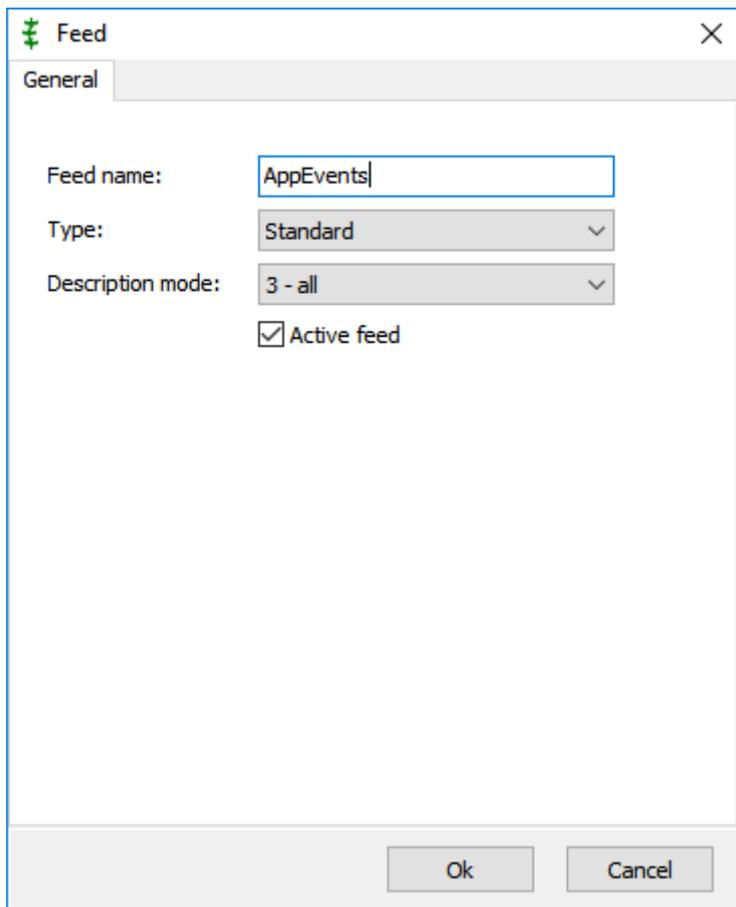
Feeds tab is divided into two areas.



The top area displays feeds and the bottom area displays event subscriptions that belong to the selected feed.

To create a new feed, press **Add feed** button.

In the Feed dialog, give name to the new Feed. The table name maximum length is 100 symbols and the following symbols are valid: English (Latin) characters, numbers, underscore. Spaces or special characters are not allowed!



Select **Standard** or **Alarm** feed in the Type field.

**Description mode** defines how Elodea will form event details (descriptions). Windows usually doesn't store event descriptions in the event logs. Instead it stores only description parameters in the log and renders the description when required based on message files, event id, event source and other parameters.

**Example:**

Application log, Source MSSQL\$SQLEXPRESS, Event 17137.

Windows Event Viewer (or Event Log Explorer) displays event description "Starting up database 'EventLogs'" But the event contains only 'EventLogs' text. A message file contains text message "Starting up database '%s'". And when displaying event description, Event Viewer finds this text message based on Event ID, source and some other data and substitutes %s with the description parameters.

Description mode can be set to one of the following values:

**0 – no description** – Elodea will not render description and forward empty description into the database. This can save the database size and increase performance. You can use this when you don't need textual event description or if you can render description on the fly, when reading events from the database.

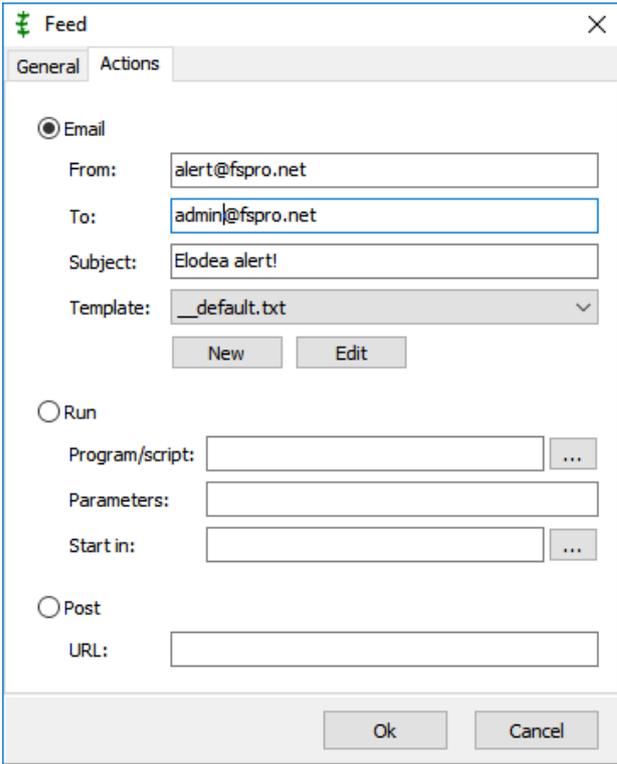
**1 – renderingInfo** – When you use Elodea along with Windows Event Forwarding, Windows Event Collector can render the description and save it under RenderingInfo key in ForwardedEvents log). In this case you can rely on this information and do not extract the description from event.

**2 – normal render** – When normal render is selected, Elodea will render event description itself based on the event data and message files.

**3 – all** – Elodea will try to use RenderingInfo key in the event, and if it doesn't exist, it will use normal rendering mode.

Enable **Active** checkbox to make this feed active. Active feeds are marked with a tick sign (✓) in the feed list.

If you select Alarm, a new tab **Actions** will appear in this dialog. Switch to this tab to setup alarm actions.



The screenshot shows a dialog box titled "Feed" with a close button (X) in the top right corner. It has two tabs: "General" and "Actions". The "Actions" tab is active. Under the "Actions" tab, there are three radio button options: "Email" (selected), "Run", and "Post".

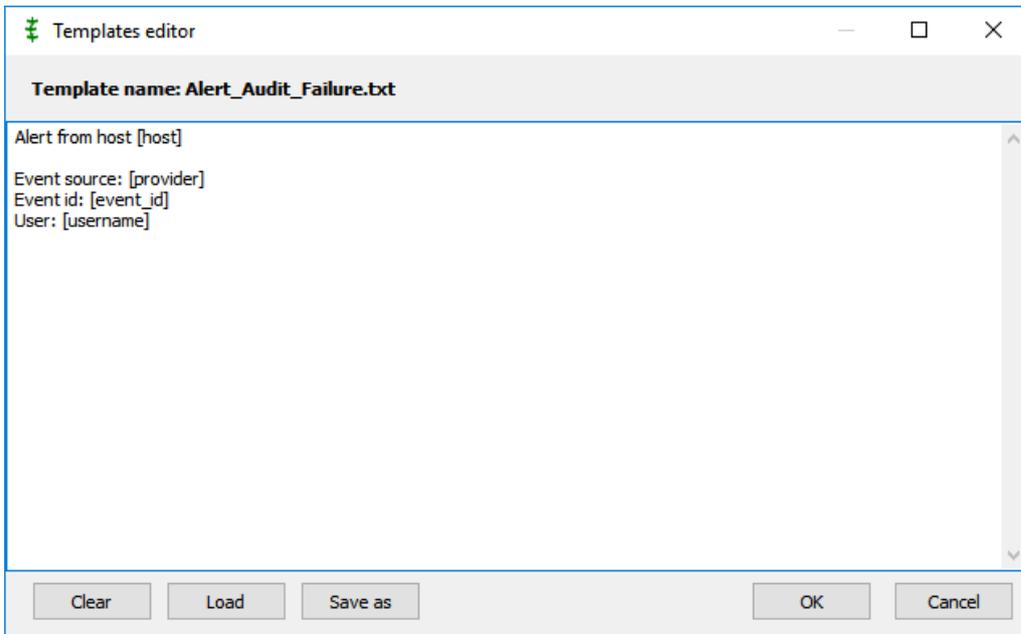
- Email:** Includes fields for "From:" (alert@fspro.net), "To:" (admin@fspro.net), "Subject:" (Elodea alert!), and "Template:" (a dropdown menu showing \_\_default.txt). Below the template dropdown are "New" and "Edit" buttons.
- Run:** Includes fields for "Program/script:" (with a browse button "..."), "Parameters:" (empty), and "Start in:" (with a browse button "...").
- Post:** Includes a "URL:" field.

At the bottom of the dialog are "Ok" and "Cancel" buttons.

You can choose between sending Email, running an application (or a script) or Posting an HTTP request.

For **Email** action, input **From**, **To** and **Subject** fields and make sure that you entered a correct email address in To field. Select a mail **template** from the list of predefined email templates or create a new one, by pressing **New** button. Email templates are the text files stored in MTemplates folder.

**New** or **Edit** buttons in the email section, let you create or modify your own email templates in the Templates editor.



Use **Load** button to load an existing template and use Save as button to save the template. Do not overwrite predefined templates (their name starts from double underscore – ( \_\_ )because it will be overwritten if you reinstall Elodea.

For **Run** action, enter a full path to the program to execute on action in **Program/Script** field. Enter command line parameters in **Parameters** field. Input the startup folder in **Start in** field.

For **Post** action, enter a URL that will receive the event information.

You can use substitution parameters in email subject, email body and run command line parameters. You can use the following substitution parameters:

Parameter name	Meaning
[e_level]	Event level
[keywords]	Event keyword mask
[provider]	Event provider (source)
[event_id]	Event id
[task]	Task category
[username]	User name
[computer]	Computer name
[channel]	Event channel (log name)
[descr]	Event description
[event_xml]	XML representation of the event
[TimeStamp]	Event date and time
[TimeStampUTC]	Event date and time in UTC
[host]	Host from which the events were received
[CRLF]	CRLF (carriage return, line feed)
[[ xpath ]]	Xpath expression to get extra data from event_xml

The http server receives 2 POST parameters in UTF8 charset:

Parameter name	Meaning
event_xml	XML representation of the event
Descr	Event description

After saving a new feed, it contains no subscriptions. To create a new subscription in feed, make sure that the required feed is selected in the list feeds and click **Add subscription** button.

A subscription edit window will appear.

The **Subscription id** field is autofilled with a new unique identifier of the subscription. You should not edit this identifier unless you know what you do.

Enter the computer name from which Elodea will collect events into the **Host** field. If you Press **L** button near this field, it will be changed to localhost.

Select the **subscription model** between Push and Pull models.

Specify an **XML query** file which requests the required events from event logs. The default location of XML query files is STemplates folder. You can create a new XML query by pressing **New** button or edit an existing query file by pressing **Edit** button.

You don't need to write XML query manually, instead you can press **Query Builder** button to build XML query with GUI. More information about Query Builder is available in **Building XML Queries** chapter.

Press **Test** button to verify XML query syntax.

**Start work at** defines whether Elodea will consume events that already stored in the event logs or not. When **future events** option is selected Elodea will ignore events stored in the event log and will collect and dispatch only new events appearing in the event logs.

When **oldest events** is selected, Elodea will start collecting events from the very first event recorded in the event log.



***If a subscription that collects old events belongs to Alarm feed, Elodea will fire notifications for these old subscription events.***

Enable **Active** checkbox to make this subscription active.

Do not confuse active subscriptions with working subscriptions. When a subscription is active, Event Collector will monitor the host for events specified in the XML query. Active subscriptions are marked with a tick sign (✓) in the subscription list. Working subscriptions are marked with a bookmark sign (🔖). This means that Event Collector has already received events for this subscription and bookmarked the latest event.

When feeds and subscriptions are created, you can modify or remove them by pressing **Edit** or **Delete** buttons.

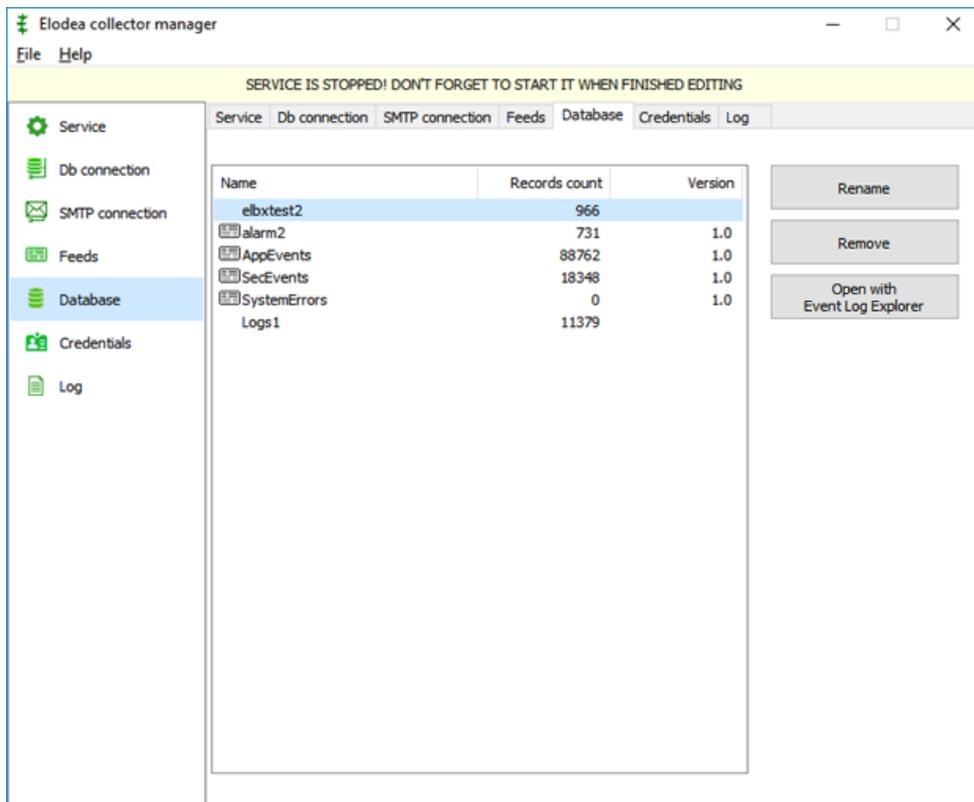


***You cannot change working subscription name, host name, or modify XML query in such a way that it will access other event logs rather than specified initially. If you need to change these parameters in the working subscription, you should create a new subscription instead.***

Deleting a feed automatically deletes all subscriptions that belong to the feed and **the database table of this feed!** Renaming the feed automatically renames the database table of the feed.

## 6.5. Managing database objects

To manage database objects, click on **Database** tab. This tab is available only if you connected to the database (pressed **Connect now!** on the Db connection tab).



The Database tab lists all tables in the database except Elodea service tables. It displays them without **fd\_** prefix for user convenience.

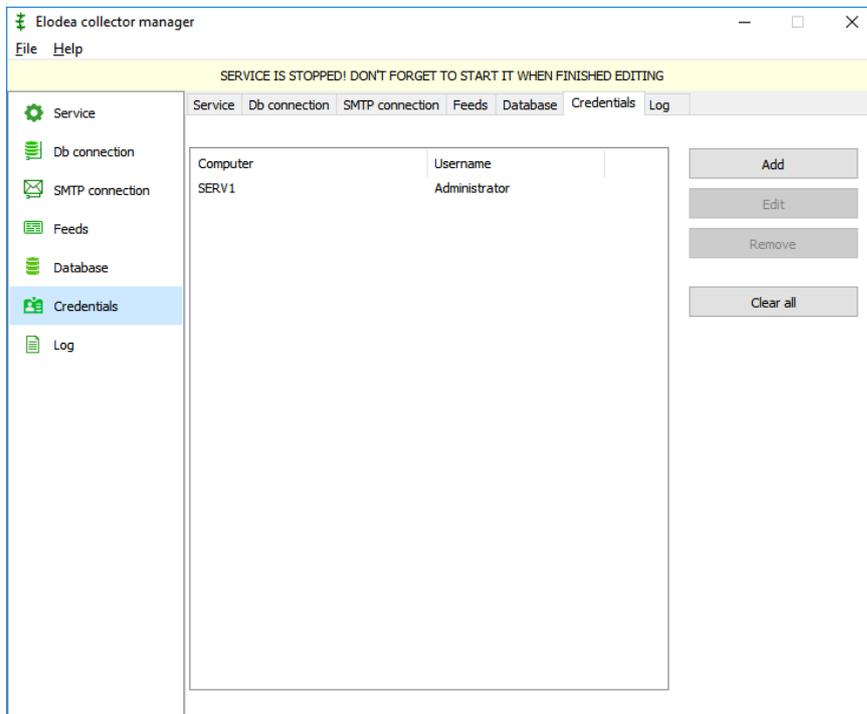
You can **Rename** or **Remove** selected table. When performing these operations, the application will prompt you to rename or remove a corresponding feed. If you don't confirm feed renaming/removing, Elodea will recreate feed table when you start Event Collector Service.

To view table contents, you can press **Open with Event Log Explorer**. If Event Log Explorer is installed on your computer, Elodea will start it (if it is not running) and display the table as Event Log Explorer log view.

## 6.6. Managing credentials

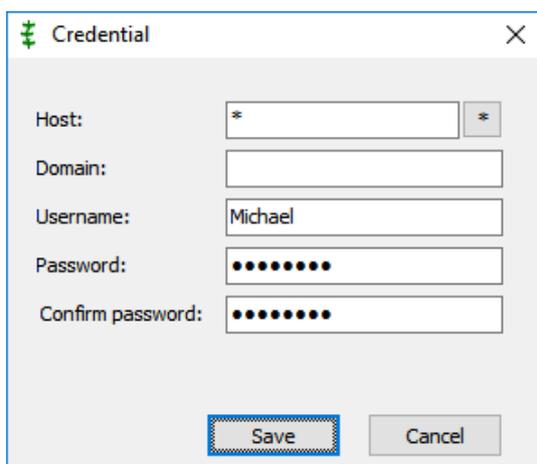
If Elodea Event Collector service runs under Local System account, it can access only local event logs by default. Even if the service runs under a user account, this user's permissions could be not enough to access some event logs.

Elodea lets you specify username and password when accessing remote event logs. To manage credentials, click on **Credentials** tab.



You can **add**, **edit**, **remove** credential or remove all credentials by pressing the corresponding buttons.

Pressing add or edit buttons open credential edit dialog.

The screenshot shows a 'Credential' dialog box with the following fields: 'Host:' with a text box containing an asterisk (\*) and a small asterisk icon to its right; 'Domain:' with an empty text box; 'Username:' with a text box containing 'Michael'; 'Password:' with a text box containing ten black dots; and 'Confirm password:' with a text box containing ten black dots. At the bottom, there are 'Save' and 'Cancel' buttons. The 'Save' button is highlighted with a blue dashed border.

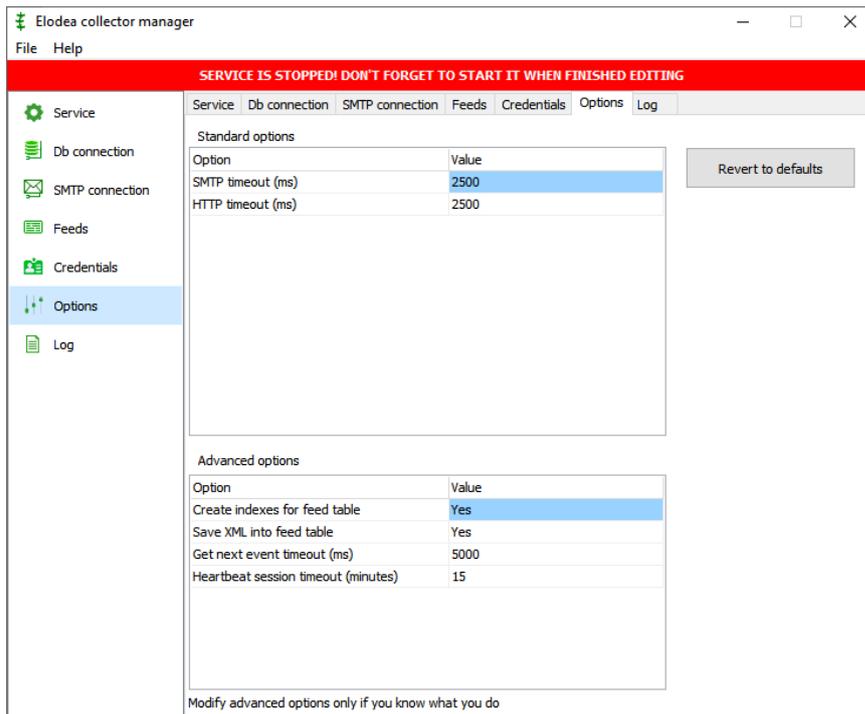
**Host** field defines that the specified username/password will be applied when Elodea connects this host. If you input \* (asterisk) into this field, Elodea will use the specified username/password for all hosts except the localhost and the other hosts listed in the credentials.

**Domain** defines the domain name of the Host. Can be empty for default domain or workgroup environment.

**Username, password** and **password confirmation** fields define a user credential which will be used to connect the Host.

Note that the application doesn't verify credential validity at this stage.

## 6.7. Extra options



You can finetune Elodea using **Options** tab.

There are 2 group of options Here

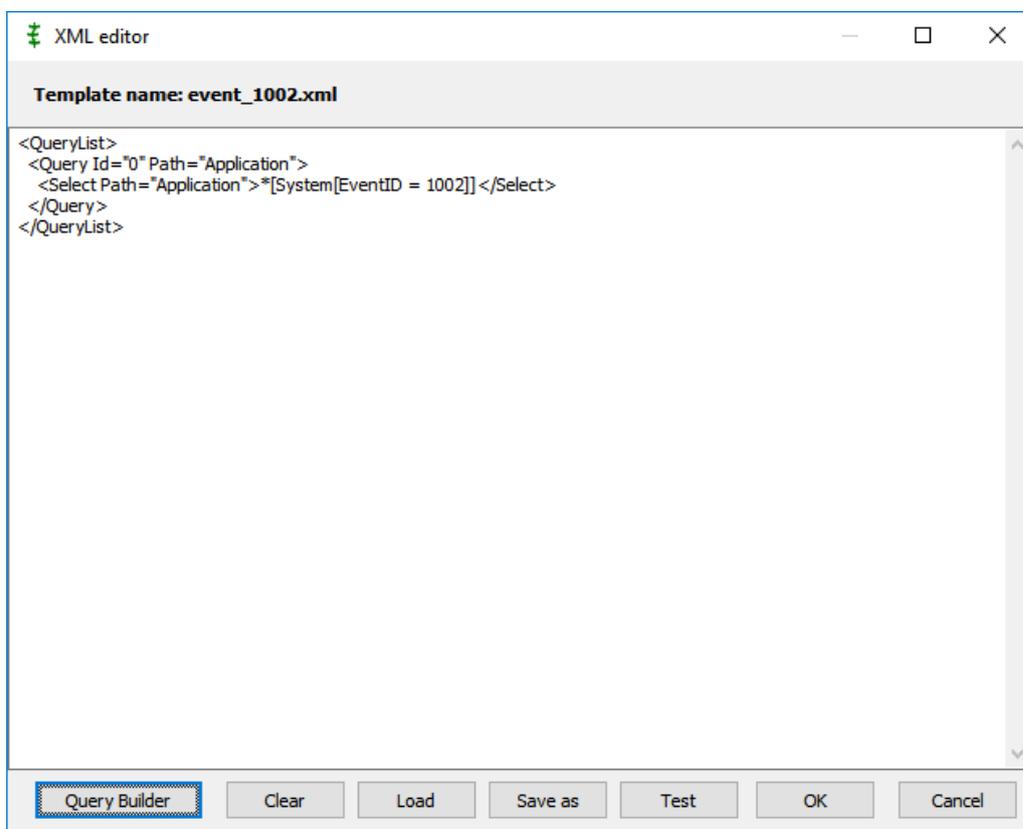
<b>Standard options</b>	
SMTP timeout (ms)	Defines a timeout (in milliseconds) when Event Collector tries to send notification email.
HTTP timeout (ms)	Defines a timeout (in milliseconds) when Event Collector tries to notify about alert via HTTP.
<b>Advanced options</b>	
Create indexes for feed table	If Yes, Elodea will create indexes on most fields in the feed table. Filter and sort operation with the feed table will be performed extremely fast, but indexes consume extra storage in the database. If No, Elodea won't create indexes and drop them if they exist.
Save XML into feed table	If Yes, Elodea will save an XML representation of the event into the database. The database will store the detailed data on the event. If No, event_XML field of the Feed table will be set to NULL value which mat dramatically reduce storage consumption.
Get next event timeout (ms)	Defines a timeout (in milliseconds) for querying an event from a log.
Heartbeat session timeout (minutes)	If no events come from the subscription during this period of time, Elodea will restart the subscription.

## 7. Building XML queries

Elodea uses structured XML queries to retrieve events from event logs. These queries are based on XPath 1.0 syntax. They are stored in XML files and can be edited or created in any text editor or directly in the Collector Settings application. Elodea comes with several predefined query files saved in STemplate folder. Refer to **Files and Folders** chapter for the list of the files and their purpose. When you create your own XML query, you can edit the existing file and save it as a new file. Do not overwrite the predefined XML files since they will be overwritten when you reinstall Elodea.

To create a new XML query file when you are editing a subscription, click **New** button in the Subscription dialog. Give a name to the new file and press **OK**.

**XML Editor** window will appear.



You can type your query in the text editor or load an existing query using **Load** button. To make the editing process easy, Elodea comes with XML Query Builder.

Click on **Query Builder** button to open **XML Query Builder** window.

Query Builder

Event log

Levels and keywords

All levels       All keywords

Information       Audit Failure       Response Time

Warning       Audit Success       Sqm

Error       Correlation Hint       Wdi Context

Critical       Correlation Hint 2       Wdi Diagnostic

Verbose       EventLog Classic

Event Source(s)   Exclude

Event ID(s)

Enter ID numbers and/or ID ranges, separated by commas, use exclamation mark to exclude criteria (e.g. 1-19,100,250-450!10,255)

```
<QueryList>
<Query Id="0">
<Select Path="Application"*</Select>
<Select Path="System"*</Select>
</Query>
</QueryList>
```

Clear      OK      Cancel

**Event log** defines from which event logs (event channels) the events will be collected. You have to specify at least one event log in this field, and if required, you can select several event logs in one query.

**Levels and keywords** define which event levels and keyword masks will be collected.

**Event source(s)** defines the event sources (providers). You can select several sources in one query. You can tick **Exclude** checkbox next to the Event source(s) to select all sources except the selected.

**Event ID(s)** – Elodea will collect events that match the specified Event IDs. If you want to specify multiple IDs, use coma as a delimiter. To specify a range of IDs, use "-".

You can use "!" to specify the exception list of events. All events and event ranges following "!" will be considered as exceptions. E.g. 10,100-1000,2000-5000!250,500-600,3000-3200 will be equal to 10, 100-249,251-499,601-1000, 2000-2999,3201-5000.

Note that this syntax is slightly different from the syntax that Windows Event Viewer uses in Event Viewer Filter window.

In the lower area of the Query Builder dialog window, the application displays current XML query. If you change any field in this window, the application will immediately reflect this change in the XML query, so you can always see your changes in XML form. You cannot edit the XML in this area, but when you press **OK** button, XML Query Builder window will be closed, and your XML will appear in the XML Editor. If required, you can edit the XML in XML Editor

window. To verify the XML syntax, press **Test** button in the XML editor. If the syntax is correct, you will get an Ok message.

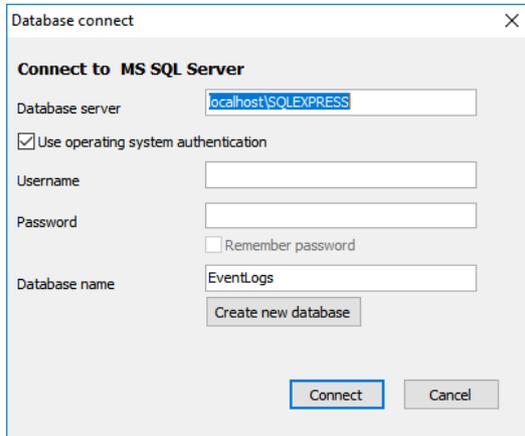
Press **OK** in the XML Editor window to close the window and save your XML.

More information about the structured XML queries and XPath expressions available at [https://msdn.microsoft.com/en-us/library/windows/desktop/dd996910\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/dd996910(v=vs.85).aspx)

## 8. Viewing event tables

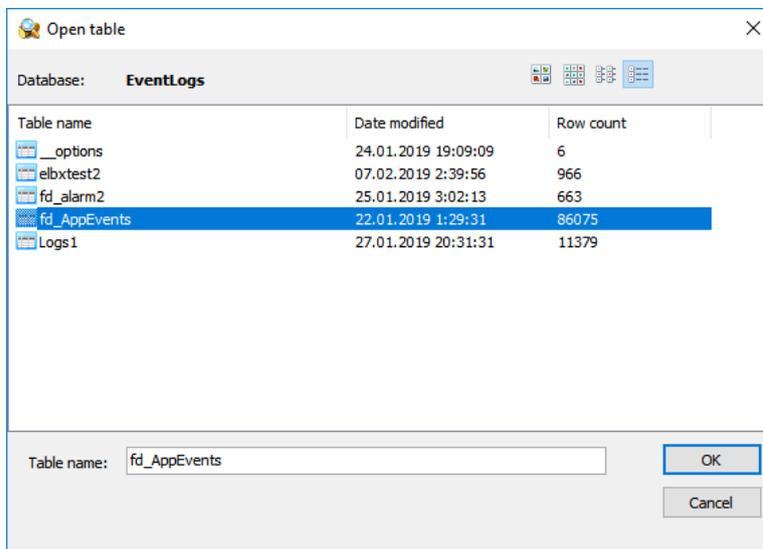
Elodea dispatches events into SQL Server tables. You can open these tables using our Event Log Explorer software available at <https://eventlogxp.com>.

To explore event tables with Event Log Explorer, start Event Log Explorer, select **Database -> Connect** from the main menu.



Type SQL server instance name, your user name and password (for SQL server authentication) and the database name, then click **Connect** button.

The select **Database -> Load table** and select the required table name in Open table dialog and click **Open** button.



Note that feed table names start with fd\_ prefix.

You can also start Event Log Explorer and display tables directly from Feeds or Database tabs.

More information is available in Event Log Explorer documentation.

## 9. Feed table format

It is not necessary to have Event Log Explorer to work with Elodea. You can also use other tools to work with the Elodea tables. Below is the design of Elodea feed table:

Column name	Data type	Column description
id	bigint	Record identifier
sub_id	uniqueidentifier	Subscription identifier
host	nvarchar(255)	Host name.
event_xml	XML	XML representation of the event
time_stamp	datetime	Event timestamp in UTC
e_level	tinyint	Event level
keywords	binary(8)	Keywords mask
provider	nvarchar(255)	Event provider (source)
event_id	Int	Event Id
task	nvarchar(255)	Task category
username	nvarchar(255)	User name
computer	nvarchar(255)	Computer name
channel	nvarchar(255)	Channel (event log name)
descr	nvarchar(MAX)	Event description
elx_type	int	Event Log Explorer compatible type
bin_data	varbinary(MAX)	Event binary data
notify_status	Int	Action error code for alarm feeds
kind	smallint	Reserved. Must be 1.

Elx\_type is calculated as follows:

For **Security** log:

If keyword mask contains Audit Failure, Elx\_type = 13 (Audit Failure)

If keyword mask contains Audit Success, Elx\_type = 12 (Audit Success)

If keyword mask contains no Audit Success or Audit Failure, the rules for other logs will be applied.

For other (non-Security) logs:

If event level = 0, Elx\_type = 2 (Information)

If event level = 1, Elx\_type = 5 (Critical)

If event level = 2, Elx\_type = 4 (Error)

If event level = 3, Elx\_type = 3 (Warning)

If event level = 4, Elx\_type = 2 (Information)

If event level = 5, Elx\_type = 1 (Verbose)

## 10. Files and folders

After installation, Elodea setup creates several folders in your system:

**C:\Program Files (x86)\Elodea** (or "C:\Program Files\Elodea" on a 32-bit system) – Elodea application folder. Contains Elodea program files, documentation and the license agreement.

**C:\ProgramData\Event Log Explorer\Default** – location of the default configuration files.

**C:\ProgramData\Event Log Explorer\Globals** – location of global settings like user credentials.

**C:\ProgramData\Event Log Explorer\STemplates** – default location of XML queries to access Windows event logs.

**C:\ProgramData\Event Log Explorer\MTemplates** – default location of email alert templates.

**C:\ProgramData\Event Log Explorer\Logs** – location of Elodea Event Collector logs and crash dumps.

Elodea program files:

EventCollector.exe – Elodea Event Collector Service.

CollectorSettings.exe – Elodea configuration tool.

eldbx.exe – an optional utility to export events to the database.

### Elodea predefined template files

#### Email template files

File name	description
__default.txt	Default email template file. Lists most of event fields in the message. You can create your own email templates based on this file.

#### XML query files

File name	description
__application_all.xml	All application events
__system_all.xml	All system events
__security_all.xml	All security events
__application_system_all.xml	All application and system events
__app_sys_sec_all.xml	All application, system and security events

__application_system_warning+.xml	Warning, error and critical events of application and system logs
__application_system_error.xml	Error and critical events of application and system logs
__security_audit_failure.xml	Audit failure events of security log
__security_logon_failed.xml	Account failed to log on events
__security_known_attack.xml	Windows detected a known attack (4649 – replay attack detected, 5148 – DoS attack detected)

## 11. Elodea Event Collector Log

Elodea logs issues into its own event log. The default location of Event Collector log is Logs\ec\_default.log. This log is a text file, so you can use any text viewer to read events from the log. When troubleshooting, always check Elodea log to find possible reasons of the problem.

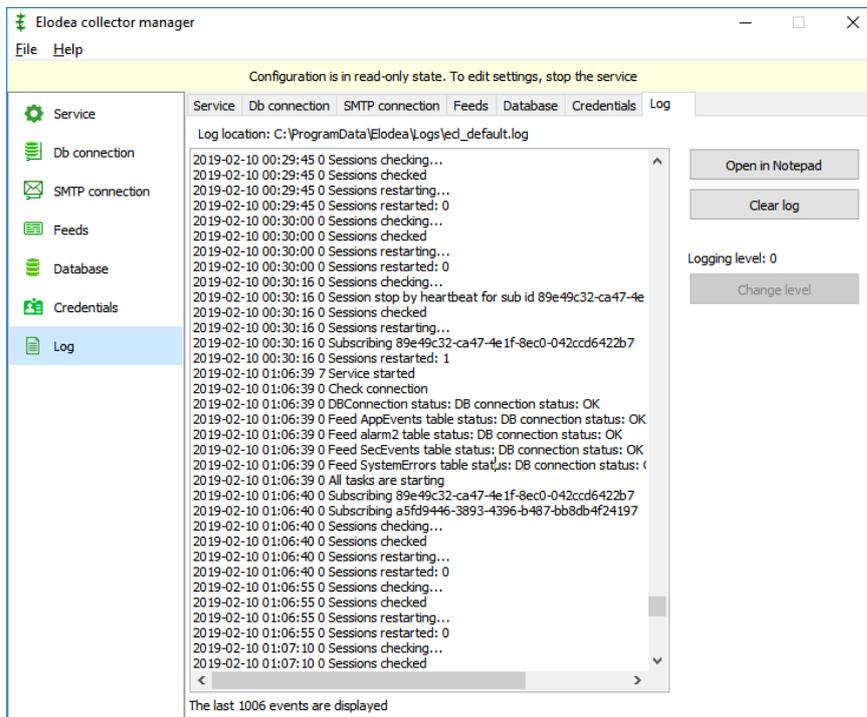
Elodea log record consist of 4 columns separated by space.

Column	Format	Note
Date	yyyy-mm-dd	Local date
Time	hh:mm:ss	Local time
Level	Number	0 – information 1 – event message rendering issues. E.g. it is impossible to get message description or category name 2 – warning 3 – error 7 – service control commands (e.g. start/stop)
Message	Text	Information about the issue

Example:

```
2019-01-12 10:59:21 3 EvtSubscribe 89e49c32-ca47-4e1f-8ec0-042ccd6422b7: Access is denied
```

You can view event collector log by switching to Log tab in the application.



To log only specific events, change logging level by pressing **Change level** button. Select a new level number from the menu. Elodea will log events of this and higher levels. E.g. if you select 2, it will log warnings, errors and service control commands.