

Introduction.

The Cybelin integrated solution includes a minimalistic ASP.NET Core middleware application designed to capture and log details of HTTP requests and responses and to block the HTTP request received from malicious IPs.

The middleware captures information such as the HTTP method, headers, query string, request body, HTTP version, and client IP, as well as the response headers, status code, response size, and response time. All this information is stored in a SQL Server database using Entity Framework Core.

The integrated solution includes also monitoring software that can be configured to blacklist IPs that the middleware will block. The monitor software can monitor different servers and includes functionality to replicate the blacklisted IPs in all the servers.

When the monitor software is used to supervise a server, it can trigger 12 types of alerts:

1. **Outbound Traffic per Client IP in Bytes per Minute**
Monitors the volume of data being sent out by each client IP every minute. A sudden increase may indicate data exfiltration or potential data leaks.
2. **Number of Requests per Client IP per Minute**
Tracks the number of requests made by each client IP per minute. A high request rate could signify suspicious activity, such as attempts to overwhelm the server or unauthorized scraping.
3. **Number of 4XX Responses per Client IP per Minute**
Alerts when a client IP receives multiple 4XX (client-side error) responses within a minute. This pattern might indicate automated tools trying to access restricted resources or a fuzzing attack.
4. **Number of Requests Without Response per Client IP per Minute**
Detects when requests from a specific client IP do not receive a response within a minute. This could indicate network issues or a client trying to overload the system with requests.
5. **Outbound Traffic per Endpoint in Bytes per Minute**
Measures the data volume being sent out per endpoint every minute. High traffic to certain endpoints may signal data extraction attempts or unintended data exposure.
6. **Number of Requests per Endpoint per Minute**
Monitors the number of requests to each endpoint per minute. A sudden spike could indicate an attempted attack on a particular API function.
7. **Number of 4XX Responses per Endpoint per Minute**
Tracks the number of 4XX responses for each endpoint every minute. Many errors on an endpoint might indicate automated tools trying to access restricted resources or a fuzzing attack.

8. **Number of Requests Without Response per Endpoint per Minute**
Detects endpoints that frequently fail to respond to requests. This might indicate performance issues or that an endpoint is being targeted in an attack.
9. **Outbound Traffic per Server in Bytes per Minute**
Measures data volume being sent out by the server every minute. Sudden spikes can reveal unauthorized data access or other anomalies.
10. **Number of Requests per Server per Minute**
Tracks the total request count per server each minute, identifying unusually high traffic that could point to a DDoS attack or heavy load.
11. **Number of 4XX Responses per Server per Minute**
Alerts on multiple 4XX errors server-wide per minute, helping to identify issues affecting multiple clients or endpoints.
12. **Number of Requests Without Response per Server per Minute**
Detects when the server fails to respond to multiple requests within a minute, indicating potential overload, system faults, or targeted disruption attempts.

The Cybelin middleware is easy to integrate with your APIs. Currently, it supports APIs developed with ASP.NET Core, and we are developing middleware for APIs written in Python, Java, JavaScript, and Go. For more information on new middleware please visit our website www.cybelin.com.

NuGet Packages

Several NuGet packages are used in the middleware to provide essential functionality, such as database management, API documentation, and design-time services.

Microsoft.EntityFrameworkCore.SqlServer: This package enables the application to work with SQL Server databases. It allows the application to store and retrieve data from a SQL Server database using Entity Framework Core.

Microsoft.EntityFrameworkCore.Tools: This package provides tools for working with Entity Framework Core migrations, allowing the developer to update the database schema as the application evolves.

Microsoft.EntityFrameworkCore.Design: This package provides design-time services required for scaffolding and managing migrations in Entity Framework Core.

Swashbuckle.AspNetCore: Swashbuckle integrates Swagger with ASP.NET Core, enabling automatic generation of API documentation. It provides an interactive UI (Swagger UI) for exploring and testing API endpoints.

Installation.

The Cybelin Monitor software works in conjunction with the logging and malicious IP blocking middleware.

To run the software, you need to download the source code for both projects:

- **Data Loss Prevention Middleware Project:** This contains the source code for logging and malicious IP blocking middleware. It's an ASP.NET Core application that includes middleware and some test endpoints you can run with Swagger to check how the software operates.
- **Cybelin Monitor Project:** This is the monitoring software used to detect alerts through the middleware.

To get started, you must create two databases used by the software:

1. **Create the Cybelin database:** This is the main database used by the monitoring software. To set it up, create a new database in SQL Server named Cybelin and then run the SQL script "Create Cybelin Database.sql."
2. **Create the CybelinServer database:** This database is used by the example API and middleware to monitor logs and block blacklisted IPs. In SQL Server, create a database named CybelinServer and then run the SQL script "Create CybelinServer Database.sql."

Next, open the Cybelin Monitor project in Visual Studio 2022 and update the connection string in the monitoring software within the DataContext class in the OnConfiguring event. Ensure the connection string points to the Cybelin database correctly.

To verify that the Cybelin database is set up correctly and the connection string is valid, go to the "Manage Monitor" menu and select "Manage Monitor Configurations." If the database and connection string are correct, the "Manage Monitor Configurations" window should display several records allowing you to configure an email account to send alert notifications.

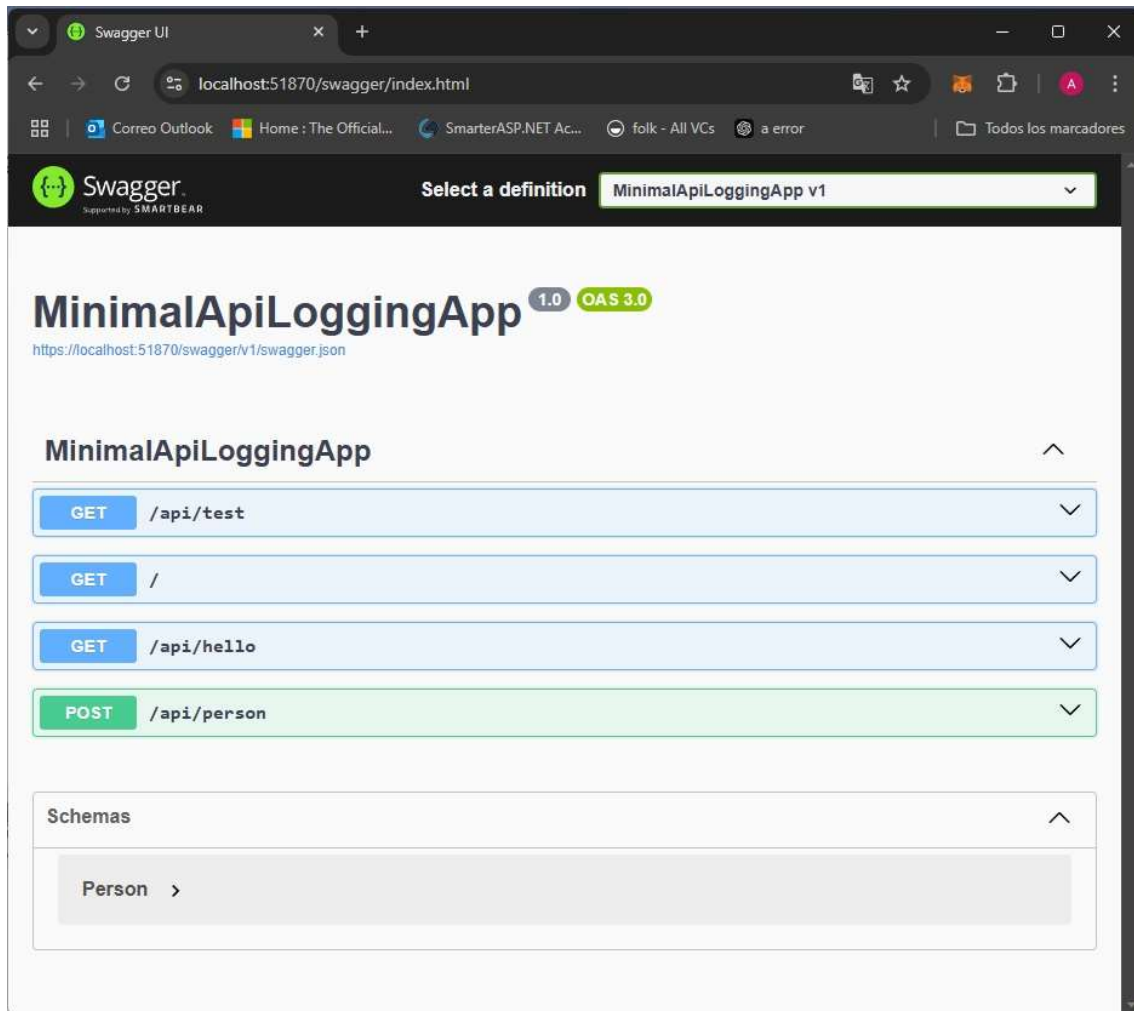
Run the monitoring software, then go to the "Manage Servers" menu and select "Manage Servers." Add a server with the name "CybelinServer" and the correct connection string for accessing the CybelinServer database. **IMPORTANT:** The connection string should be entered without starting and ending quotation marks.

To verify that the CybelinServer database is set up correctly and the connection string is valid, go to the "Manage Servers" menu and select "Manage Servers Configurations." A window named Server Configurations will open with a combobox. Select the server "CybelinServer" from the combobox, and you should see a configuration record in the grid named "MaliciousIpCheckIntervallnSeconds" with a value of 55. If this record appears correctly, it indicates that the CybelinServer database and connection string are set up successfully.

Once both databases are correctly created and the Cybelin Monitor software is running correctly, open the Data Loss Prevention Middleware project in Visual Studio 2022, and update the connection string in the appsettings.json file with the correct connection details for the CybelinServer database.

First steps.

You can then run the middleware project in Visual Studio, and it will open a browser displaying the message “Welcome to the Minimal API!” Modify the URL in the browser to add “/swagger,” and you’ll access the Swagger interface to test the middleware.

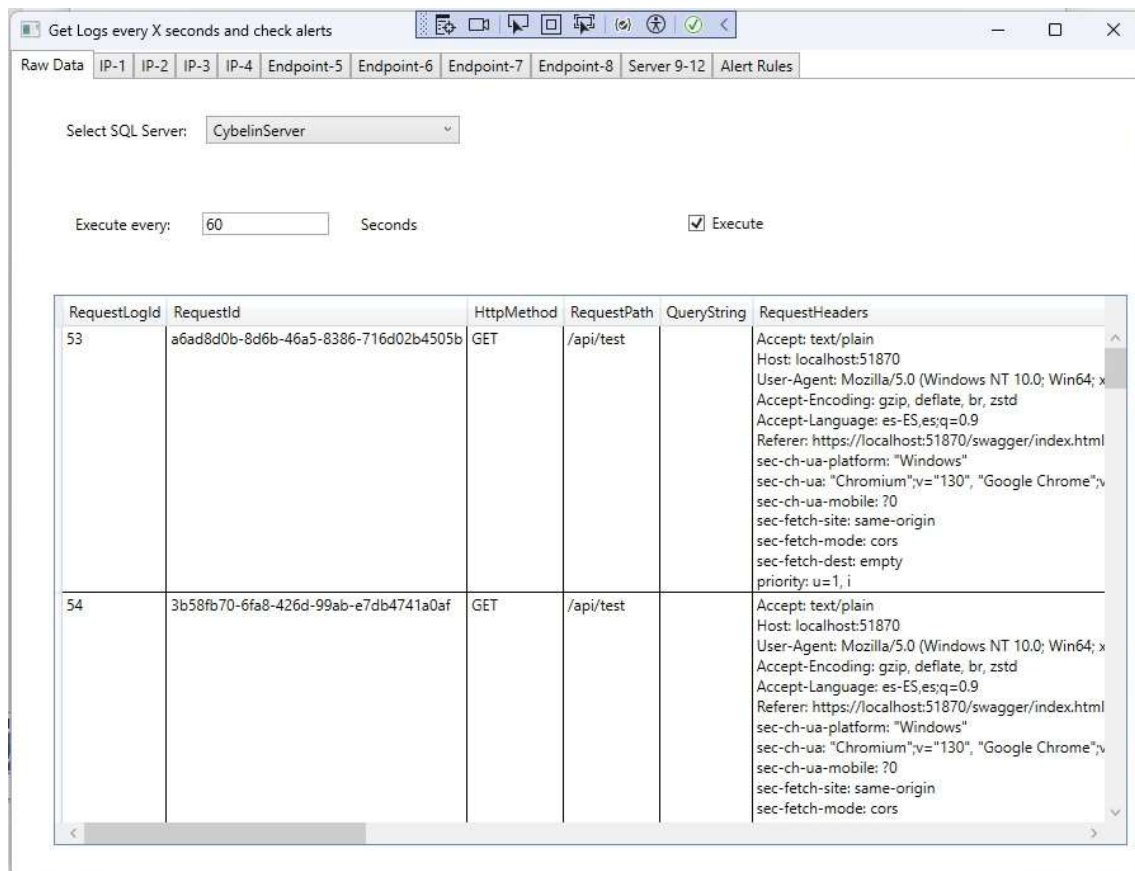


You can execute the different endpoints. Each time an endpoint is executed, a record with the HTTP request data is added to the RequestLogs table, and a record with the HTTP response data, except for the body, is added to the ResponseLogs table. The response body is not added to the ResponseLogs table because it may contain sensitive information. You can verify in SQL Server Management Studio that records are being added to these tables in the CybelinServer database.

These endpoints have been added to the middleware software so you can test it before integrating it into your own APIs.

To capture middleware data from the Cybelin Monitor application, run this application, go to the Logs menu, and open the window "Get Logs Every X Seconds And Check Alerts." Select the server “CybelinServer” from the combobox and activate the “Execute”

checkbox. Execute some endpoints so that the middleware captures the logs, and after 60 seconds, the captured data will appear in the Cybelin Monitor.



The screenshot shows the Cybelin Monitor window with the title "Get Logs every X seconds and check alerts". The interface includes a tabbed menu at the top with options: Raw Data, IP-1, IP-2, IP-3, IP-4, Endpoint-5, Endpoint-6, Endpoint-7, Endpoint-8, Server 9-12, and Alert Rules. Below the tabs, there is a "Select SQL Server:" dropdown menu set to "CybelinServer". Below that, there is a "Execute every:" input field with the value "60" and a "Seconds" label, followed by a checked "Execute" checkbox.

The main area displays a table with the following columns: RequestLogId, RequestId, HttpMethod, RequestPath, QueryString, and RequestHeaders. The table contains two rows of data:

RequestLogId	RequestId	HttpMethod	RequestPath	QueryString	RequestHeaders
53	a6ad8d0b-8d6b-46a5-8386-716d02b4505b	GET	/api/test		Accept: text/plain Host: localhost:51870 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x Accept-Encoding: gzip, deflate, br, zstd Accept-Language: es-ES,es;q=0.9 Referer: https://localhost:51870/swagger/index.html sec-ch-ua-platform: "Windows" sec-ch-ua: "Chromium";v="130", "Google Chrome";v sec-ch-ua-mobile: ?0 sec-fetch-site: same-origin sec-fetch-mode: cors sec-fetch-dest: empty priority: u=1, i
54	3b58fb70-6fa8-426d-99ab-e7db4741a0af	GET	/api/test		Accept: text/plain Host: localhost:51870 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x Accept-Encoding: gzip, deflate, br, zstd Accept-Language: es-ES,es;q=0.9 Referer: https://localhost:51870/swagger/index.html sec-ch-ua-platform: "Windows" sec-ch-ua: "Chromium";v="130", "Google Chrome";v sec-ch-ua-mobile: ?0 sec-fetch-site: same-origin sec-fetch-mode: cors

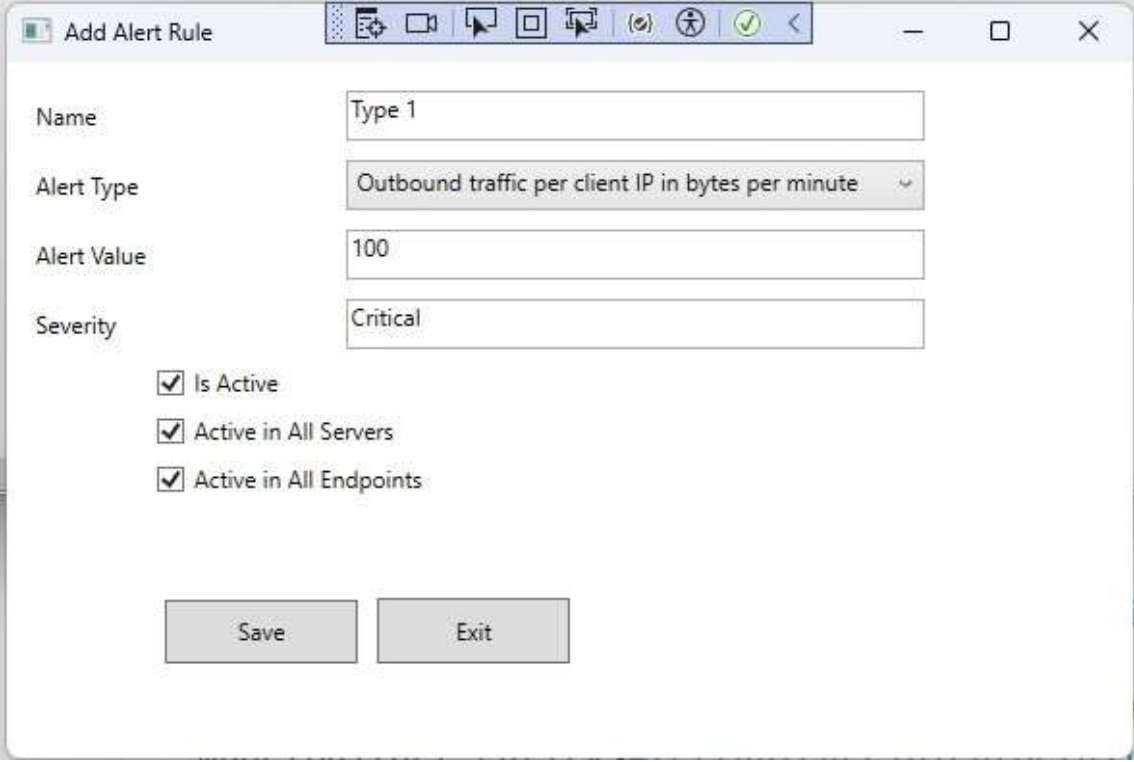
Each row in the grid displays data from an HTTP request and its corresponding HTTP response. To view the response data, move the grid's horizontal scrollbar to the right.

This window has several tabs that allow you to see the data used to generate the 12 types of alerts. You can open the different tabs in the window to see how the data corresponds to the various alert types:

- Tabs **IP-1 to IP-4** show the data used to trigger alerts of types 1 to 4. These alerts depend on the client IP address executing the endpoints.
- Tabs **Endpoint-5 to Endpoint-8** show the data used to trigger alerts of types 5 to 8. These alerts depend on the endpoints.
- The **Server 9-12** tab displays the data used to trigger alerts of types 9 to 12. These alerts depend on the server.
- The **Alert Rules** tab shows the active alerts for this server. Currently, none are displayed because they are not yet configured.

To configure an alert, go to the **Alert Management** menu, open the **Manage Alert Rules** window, and click the **Add** button. A window called **Add Alert Rule** will appear where you must complete all fields. Follow these steps:

- Choose a name for the Alert Rule, such as "Type 1."
- In the combobox, select the first alert type, named "Outbound traffic per client IP in bytes per minute."
- Set a low value in **Alert Value** for testing, such as 100. This will trigger an alert when a client receives responses equal to or greater than 100 bytes in size.
- Assign a severity level, for example, "Critical."
- Activate all three checkboxes: **Is Active**, **Active in All Servers**, and **Active in All Endpoints**.
- Click the **Save** button.



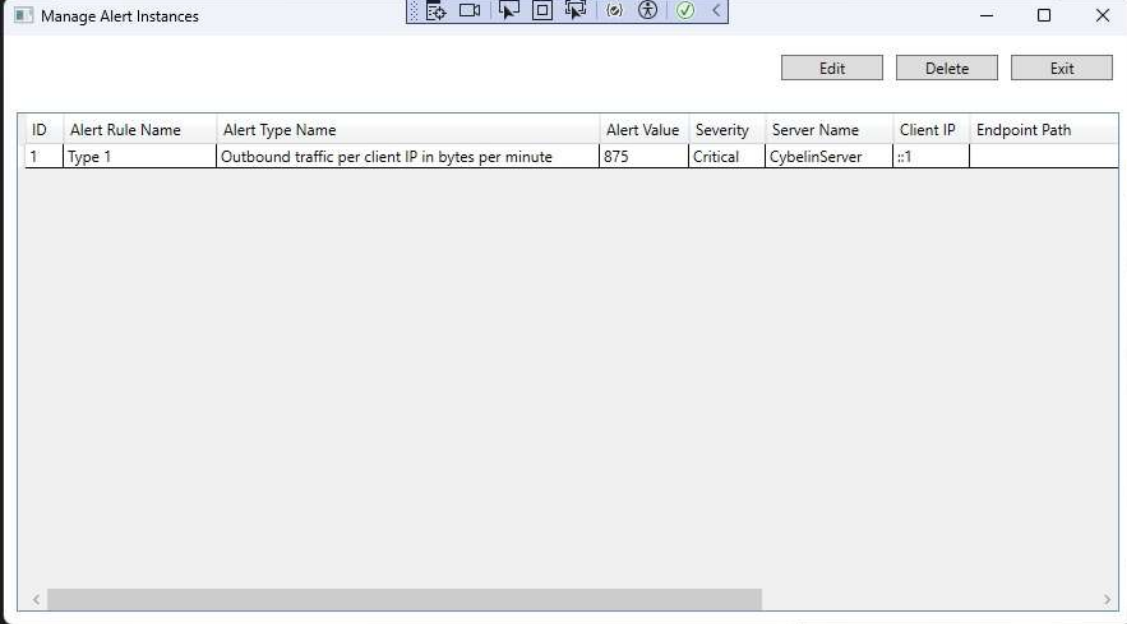
The screenshot shows the "Add Alert Rule" dialog box. The "Name" field is set to "Type 1". The "Alert Type" dropdown is set to "Outbound traffic per client IP in bytes per minute". The "Alert Value" field is set to "100". The "Severity" dropdown is set to "Critical". The checkboxes for "Is Active", "Active in All Servers", and "Active in All Endpoints" are all checked. The "Save" and "Exit" buttons are at the bottom.

This way, you have set up a test alert. To activate the alert, go to the "Get Logs Every X Seconds And Check Alerts" window, select the server "CybelinServer," and check the "Execute" checkbox.

Next, run an endpoint multiple times to ensure the volume of transmitted data exceeds 100 bytes. For instance, you can trigger several endpoints 10 times.

After 60 seconds, you will see the logging data in the first tab, "Raw Data."

In the IP-1 tab, you can check the total traffic generated from the different endpoints. If the value exceeds 100 bytes, an alert will have been triggered and stored in the “AlertInstances” table. To verify this, go to the **Alert Management** menu and open the window called “Manage Alert Instances.” A record should appear on the grid with the details of the generated alert.



The screenshot shows a window titled "Manage Alert Instances" with a toolbar at the top containing icons for various actions. Below the toolbar are three buttons: "Edit", "Delete", and "Exit". The main area of the window contains a table with the following data:

ID	Alert Rule Name	Alert Type Name	Alert Value	Severity	Server Name	Client IP	Endpoint Path
1	Type 1	Outbound traffic per client IP in bytes per minute	875	Critical	CybelinServer	::1	

The table has a horizontal scrollbar at the bottom, indicating that the data is wider than the visible area.

To view all fields of the Alert Instance, you can move the grid's horizontal scrollbar to the right or click the “Edit” button, which will display all fields of the Alert Instance.

Alert Rule Name:
Type 1

Alert Type Name:
Outbound traffic per client IP in bytes per minute

Alert Value:
875

Severity:
Critical

Server Name:
CybelinServer

Client IP:
::1

Endpoint Path:

Triggered At (UTC):
2024-11-01 05:18:02

Resolved At (UTC):

Status:
Firing

Save Exit

Here, and in all windows of the Cybelin Monitor application, date and time data are displayed in reference to UTC (Coordinated Universal Time). This allows you to have servers in different time zones while handling consistent datetime data across various time zones.

This window displays all data for the Alert Instance, and the “Status” combobox allows you to change its status. There are three possible statuses:

- **Firing** (when the Alert Instance is created).
- **Work in Progress.**
- **Resolved.**

When an alert of a certain type is in the Firing state, another alert of the same type is not generated to avoid creating hundreds or thousands of alerts of the same type (later, we’ll see that for each Alert Instance, the monitoring program allows sending a notification by email. If there were hundreds or thousands of alerts of the same type, this would lead to hundreds or thousands of identical emails).

More exactly when a new alert is triggered the Cybelin monitor software verifies if exists an alert with the same Server identifier, the same Alert Rule identifier, the same Alert Type

identifier and the same Status. In case it exists the software does not create an Alert Instance.

API Endpoints

The application exposes three main API endpoints, each demonstrating basic functionality.

GET /

This endpoint returns a simple welcome message when accessed.

GET /api/hello

This endpoint accepts two query parameters (`name` and `age`) and returns a personalized greeting message. Example request: ` /api/hello?name=John&age=30` .

POST /api/person

This endpoint accepts a JSON object representing a `Person` in the body of the request. It expects the following structure:

```
{  
  "name": "John",  
  "age": 30  
}
```

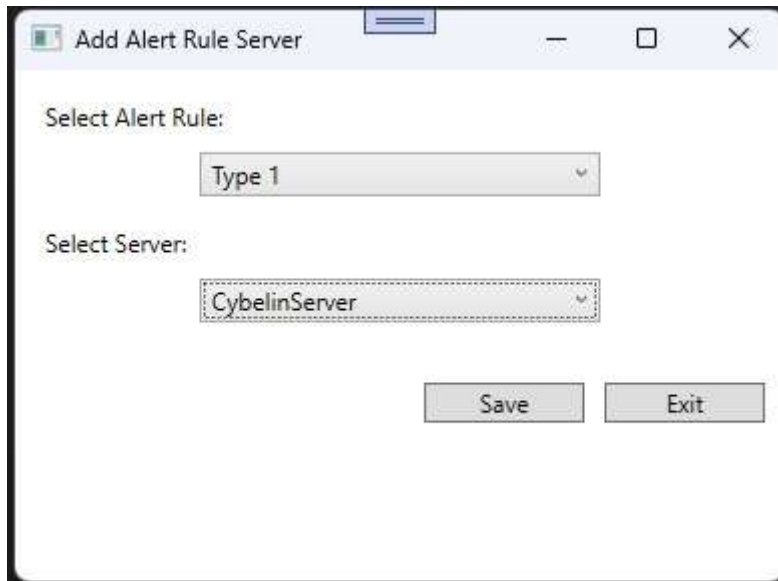
The endpoint returns a confirmation message with the received data.

Alert Rules Servers

When creating an Alert Rule, you can specify whether it applies to one or more specific servers or to all servers.

To trigger an Alert Rule only for a specific server, uncheck the “Active In All Servers” checkbox when creating the Alert Rule and add the specific server through the Alert Rules Servers window.

For example, you can add a record to the AlertRulesServers table by clicking the “Add” button in the Manage Alert Rules Servers window. Then, select the Alert Rule named “Type 1” that you created earlier from the “Select Alert Rule” combobox and choose the server named “CybelinServer” from the “Select Server” combobox.

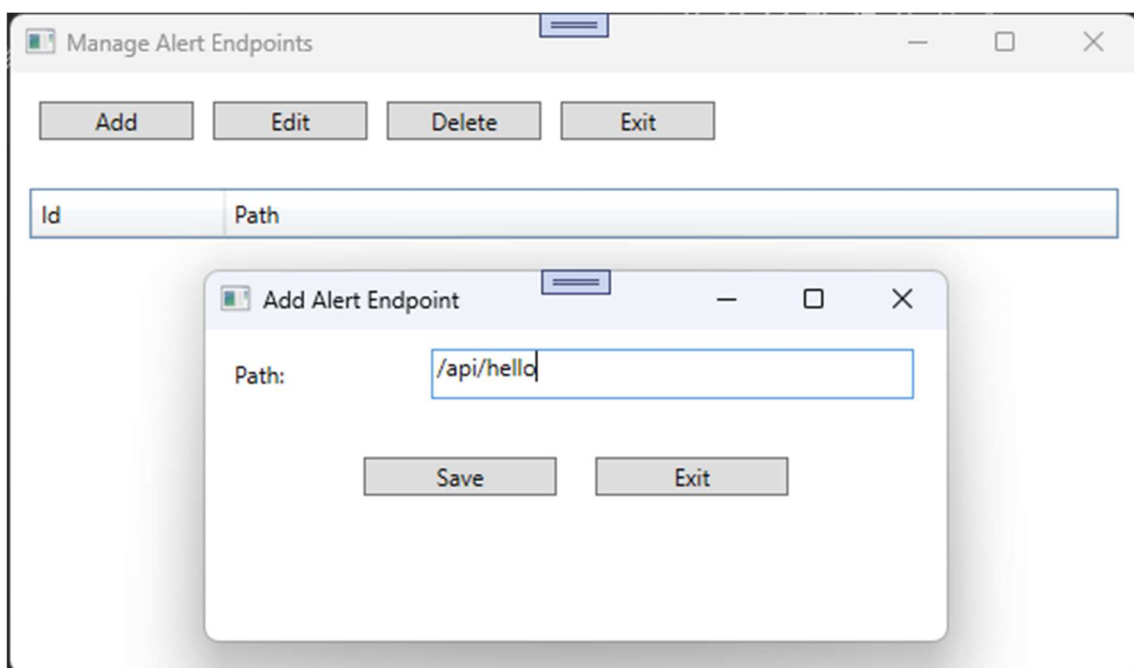


In this way, you can create Alert Rules that are specific to each server.

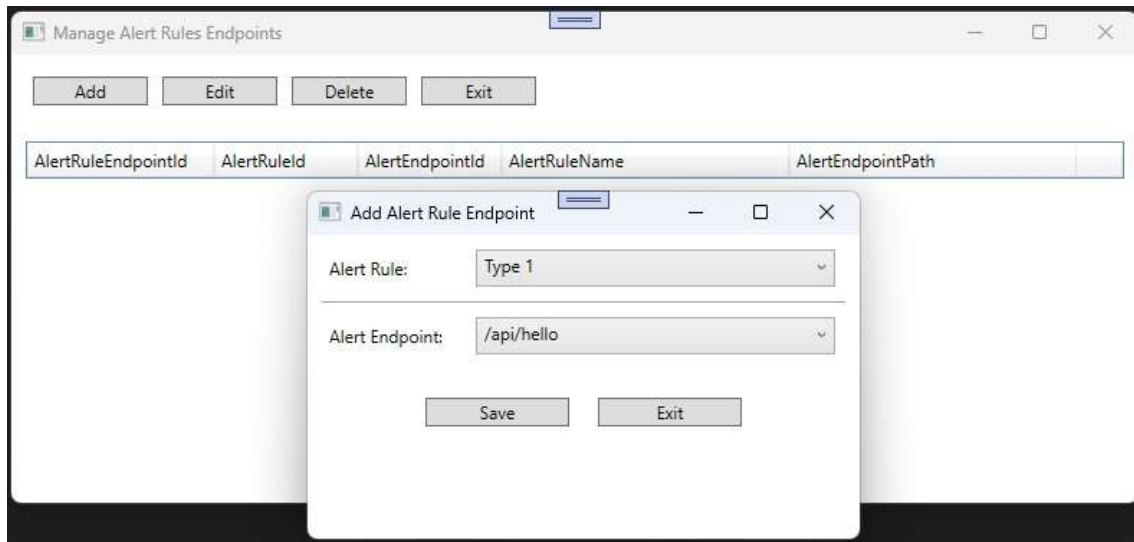
Alert Rules Endpoints.

Similarly, when creating an Alert Rule, you can specify whether it applies to all endpoints or only to one or more selected endpoints. If you want an Alert Rule to apply only to certain endpoints, uncheck the “Active in All Endpoints” checkbox when creating the Alert Rule.

To create an Alert Rule that depends on one or more endpoints, first add the endpoints you wish to monitor using the Manage Alert Endpoints window.

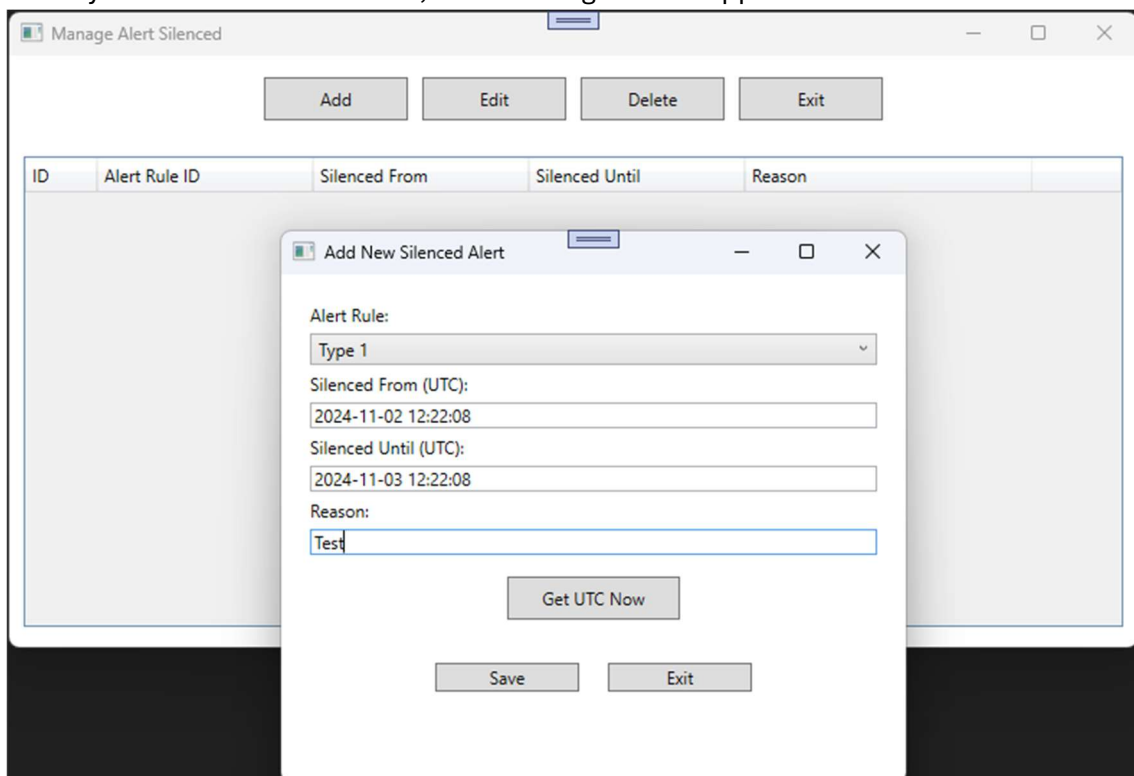


Once the endpoint has been created, you need to add a record to the AlertRulesEndpoints table using the Manage Alert Rules Endpoints window by selecting the values in the Alert Rule and Alert Endpoint comboboxes.



Alert Silenced.

The Cybelin Monitor program allows you to define a time period during which a specific Alert Rule will not be triggered. This is done using the Manage Alert Silenced window. When you click the “Add” button, the following window appears:

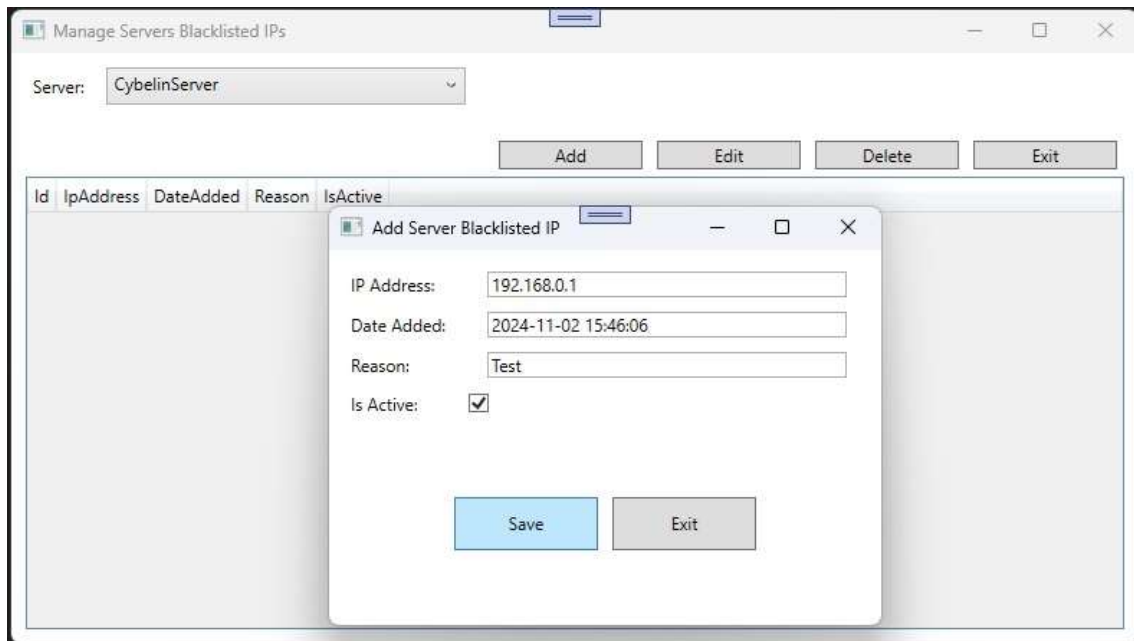


During the time period between the datetime value in the “Silenced From (UTC)” field and the value in the “Silenced Until (UTC)” field, the alert selected in the Alert Rule combobox will not be triggered.

Blacklisted IPs

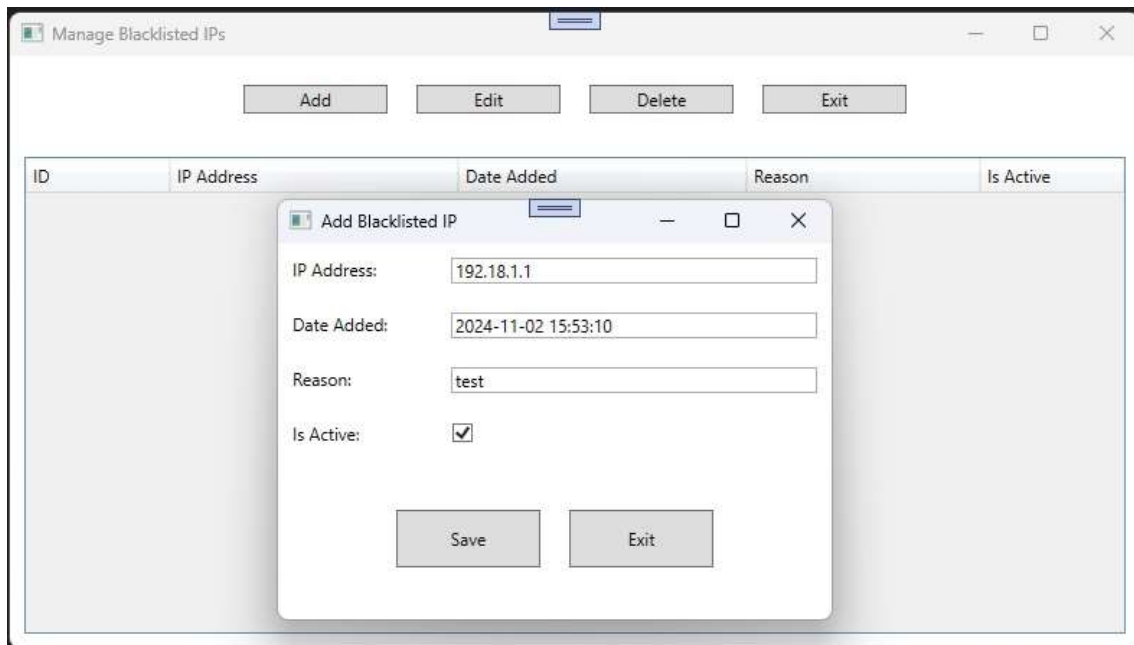
The Cybelin Monitor software allows you to create a list of blacklisted IPs. When a server receives an HTTP request from an IP that is on its blacklisted IPs list, the server responds with an “Error 403 Forbidden” message.

The blacklisted IPs list can be configured on servers in two ways. The first way is by selecting the “Manage Servers” menu and opening the “Manage Servers Blacklisted IPs” window, which allows you to add IP addresses directly to the server’s “BlacklistedIps” table. To do this, click the “Add” button, and a window will appear where you can select the server via a combobox, enter an IPv4 or IPv6 address, and add it to the server’s list.

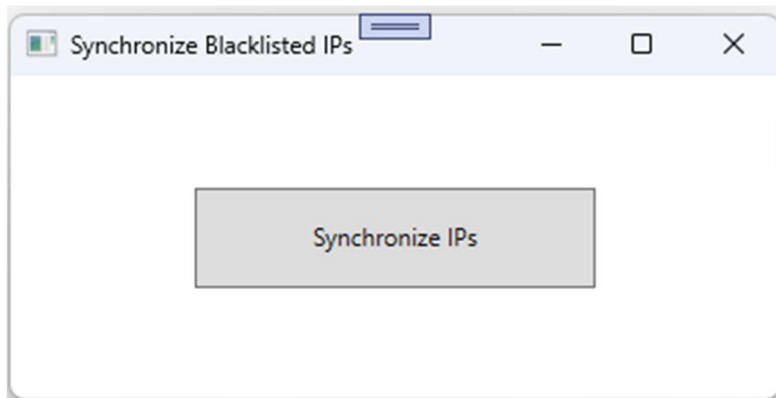


The second way to create a blacklisted IPs list is to set it up in the Cybelin Monitor database and replicate the list across all server databases.

To add blacklisted IP addresses to the Cybelin Monitor database, click on the “Manage Monitor” menu and open the “Manage Monitor Blacklisted IPs” window. The following window will appear, allowing you to add IP addresses to the Cybelin Monitor database:

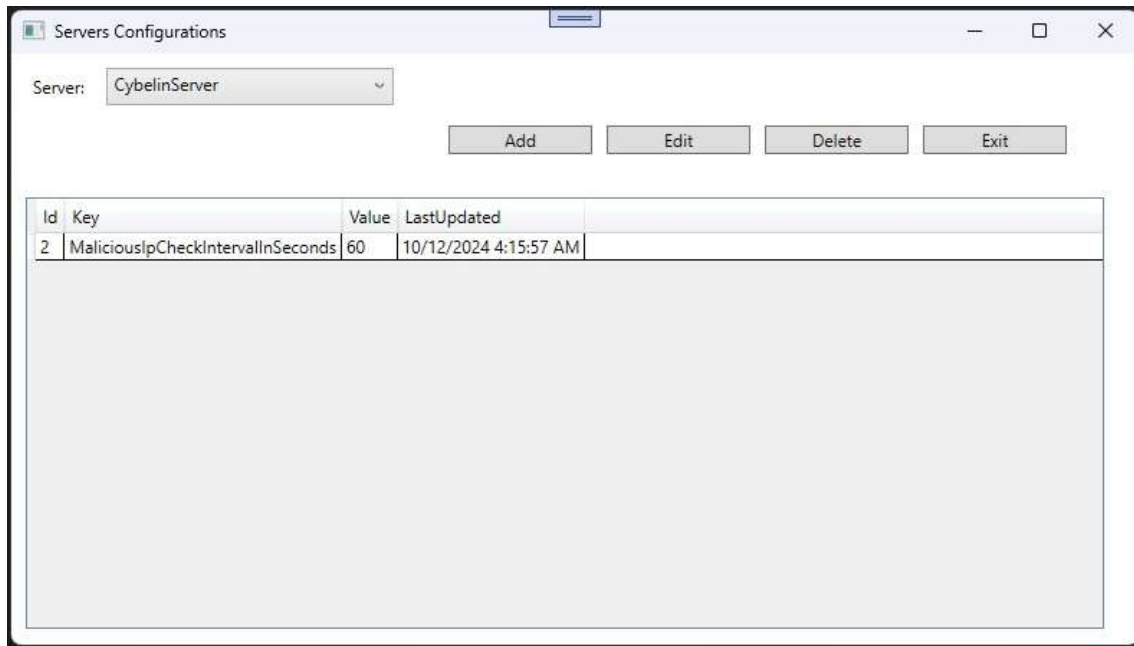


Once you have created the list of blacklisted IP addresses in the Cybelin Monitor database, you can replicate this list across all servers. To do this, select the “Manage Servers” menu, open the “Synchronize Servers Blacklisted IPs” window, and click the “Synchronize IPs” button.



In this way, the list defined in the Cybelin Monitor database is copied to the “BlacklistedIps” tables of all servers, allowing centralized management of blacklisted IPs.

The middleware for blocking malicious IPs on the servers reads data from the “BlacklistedIps” table every 60 seconds to keep an updated in-memory list of blacklisted IPs. This time interval can be configured from the “Manage Servers” menu by opening the “Manage Servers Configurations” window.



Whitelisted IPs

The whitelisted IPs table is used to prevent alerts from being triggered by HTTP requests originating from trusted IP addresses.

To add an IP address to this table, go to the “Manage Monitor” menu and open the “Manage Monitor Whitelisted IPs” window. Click the “Add” button, and the following window will appear, allowing you to enter whitelisted IPs

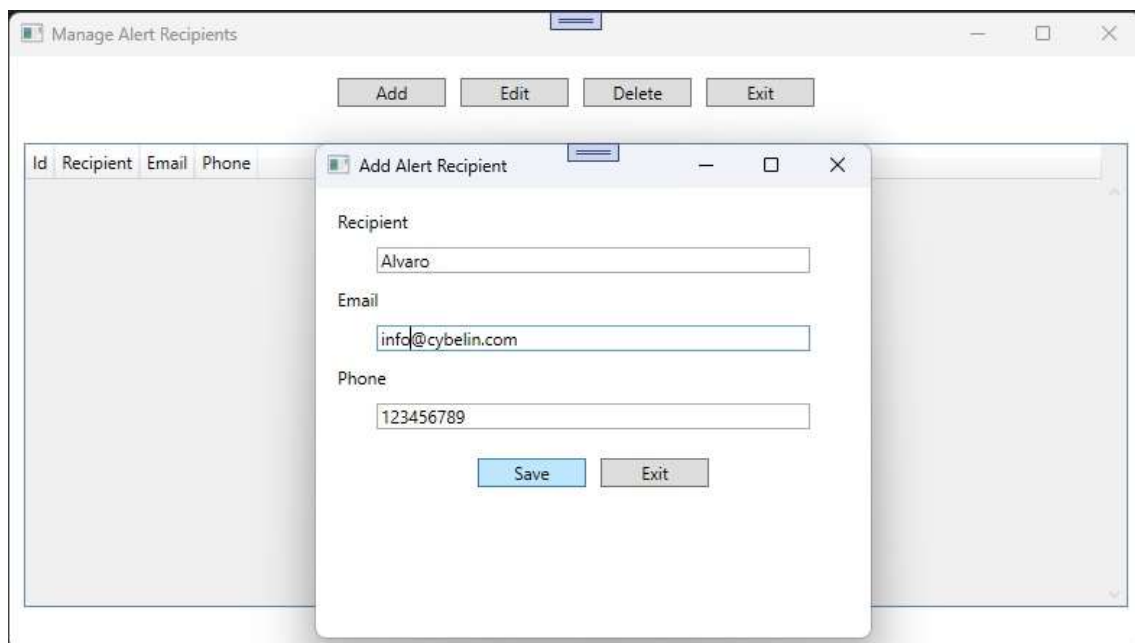


Notifications

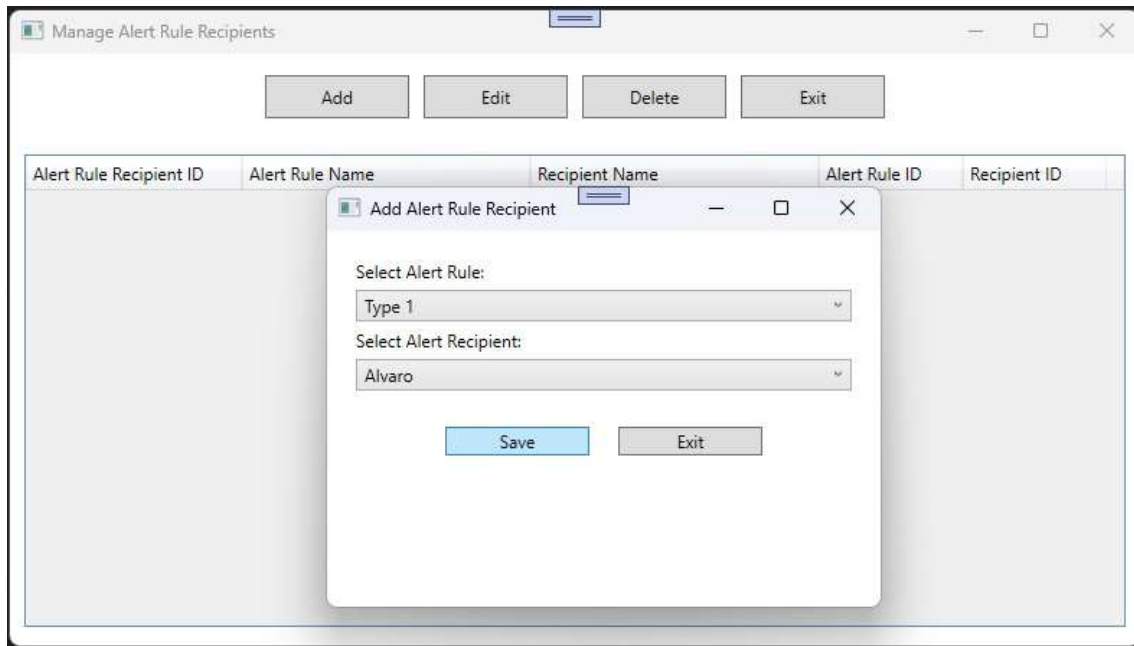
The Cybelin Monitor software allows you to generate notifications and send them by email when an alert is triggered. To do this, follow these steps:

1. Create one or more records in the Recipients table for the person or people who will receive notifications.
2. Create a record in the Alert Rule Recipients table for each person who will receive a notification for each Alert Rule.
3. Configure the SMPT client in the configurations table.

To add a record to the Recipients table, go to the “Alert Management” menu and open the “Manage Alert Recipients” window. Click the “Add” button, and the following window will appear:

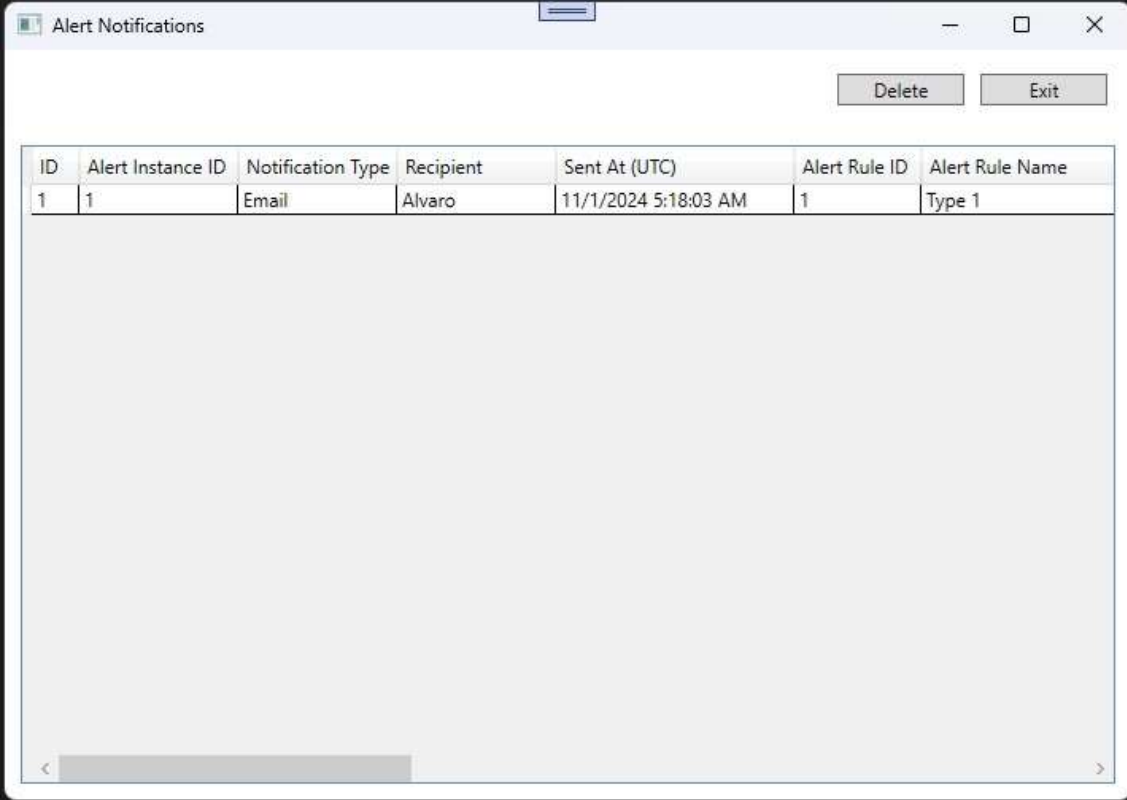


Once the record is created in the Recipients table, you can add records to the “Alert Rule Recipients” table. To do this, go to the “Alert Management” menu and open the “Manage Alert Rules Recipients” window. Press the “Add” button, and the following window will appear:



Mediante el primer combobox puede seleccionar la Alert Rule y mediante el segundo combobox puede seleccionar el Recipient. De esta forma puede añadir los recipients que desee y Cybelin monitor generará una notificación para cada uno de ellos cuando se cree la alerta seleccionada.

Cuando se genera una alerta que tiene un Recipient asociado, Cybelin monitor crea una notificación que se almacena en la tabla "AlertNotifications". Puede consultar y borrar los registros de la tabla "AlertNotifications" desde el menú "Alert Management" y abriendo la ventana "Manage Alert Notifications":



ID	Alert Instance ID	Notification Type	Recipient	Sent At (UTC)	Alert Rule ID	Alert Rule Name
1	1	Email	Alvaro	11/1/2024 5:18:03 AM	1	Type 1

As you can see, this window does not allow you to add or modify records because the records in the “AlertNotifications” table are generated automatically by Cybelin Monitor when an alert is triggered.

To ensure that alert notifications are sent to recipients by email, you need to configure your SMTP client. To do this, go to the “Manage Monitor” menu and open the “Manage Monitor Configurations” window. You should update the “Value” field of the email configuration records to work with your own email setup.

Manage Monitor Configurations			
<div> Add Edit Delete Exit </div>			
ID	Key	Value	Last Updated
3	Email SmtpClient	smtp.office365.com	10/22/2024 7:04:47 AM
4	Email Port	587	10/22/2024 7:05:38 AM
5	Email UseDefaultCredentials	false	10/22/2024 7:06:22 AM
6	Email Credential Name	Your email	11/3/2024 6:57:45 AM
7	Email Credential Password	Your password	11/3/2024 6:58:08 AM
8	Email EnableSsl	true	10/22/2024 7:08:16 AM
9	Email From	Your email	11/3/2024 6:58:26 AM
10	Email IsBodyHtml	false	10/22/2024 7:09:06 AM

Get Logs from one IP.

You can view the HTTP request and HTTP response data for a specific IP address. To do this, go to the Logs menu and open the “Get Logs From One Client IP” window. The following window will appear:

Get Logs From One Client IP

Query IP Logs

Server: CybelinServer

UTC Datetime: 2024-10-03T07:04:51Z

UTC Now

Client IP: ::1

Get Data

RequestLogId	RequestId	HttpMethod	RequestPath	QueryString	RequestHeaders
1	4bd6f1e0-802b-48a9-877c-e1eb24f273ec	GET	/		Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 Host: localhost:51870 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/115.0 Accept-Encoding: gzip, deflate, br Accept-Language: es-ES, es;q=0.9, en-US;q=0.8, en;q=0.7 Upgrade-Insecure-Requests: 1 sec-ch-ua: "Chromium", "Chromium";v=120, Chrome;v=120 sec-ch-ua-mobile: ?0 sec-ch-ua-platform: "Windows" sec-fetch-site: none sec-fetch-mode: navigate sec-fetch-user: ?1 sec-fetch-dest: document priority: u=0, i

In this window, you can use the combobox to select the server from which you want to retrieve logs and enter the UTC date and time in the UTC Datetime field to get log records from that date and time onward. The “UTC Now” button helps you by entering the current UTC date and time, and the “Get Data” button retrieves the logs from the selected server's database.

Conclusion

The Cybelin integrated solution provides a comprehensive solution for logging HTTP request and response details in ASP.NET Core applications and for blocking HTTP requests from malicious IPs. It captures valuable information for auditing, monitoring, and debugging, while using a minimalistic design.

The monitor software is easy to use and can trigger different alerts to detect any data loss. The integration with Entity Framework Core allows for easy storage and retrieval of data, and Swagger enables easy API exploration.