



O-Insights™ IoT Drivers– 4.1 User Guide

Contents

Contents.....	2
Introduction	5
Introduction to O-Insights.....	5
Features	6
Technical Overview	7
About MQTT	7
About Webhooks.....	9
About OPC UA.....	10
Prerequisites.....	12
Drivers	13
IoT Configurator Tool	13
IoT Service.....	14
Configuring IoT Service	14
MQTT Service.....	20
Configuring MQTT Driver.....	20
OPC UA Service.....	31
Configuring OPC UA Driver	31
Driver List	31
OPC UA Configuration.....	32
Basic Config.....	32
Server Config	33
Client Config	35
Adding OPC Servers.....	35
OPC UA Service as a Server	39
Camera Presets	40
Activating Presets By calling “Goto” method.....	40
Activating Presets By calling “Goto” method.....	40
PTZ-Position.....	41

Get Absolute position of PTZ Camera	41
Changing Position	41
Streaming Properties.....	42
Camera Cache Update	43
User Defined Events.....	43
Triggering Events.....	43
<i>By calling TriggerByEventName method</i>	43
<i>By Trigger method</i>	44
<i>By changing value</i>	44
Data Manager Service.....	44
Configuring Data Manager Service.....	45
IoT Config Tool	49
Setting up the Config Tool	49
Starting the Config Tool.....	49
Creating and Organizing Folders	49
Points and Histories.....	49
Alarm Configuration.....	60
Configuring the IoT Config Tool.....	63
Configurations.....	63
DB Stats.....	63
Audit Trail	64
O-Insights Publish Topic Plugin	65
Publish Topics.....	65
Creating Publish Topics.....	66
Creating Custom Publish Topics.....	67
Leveraging the IoT Data	69
O-Insights for VMS.....	69
IoT Data Widget	69
IoT Data List Widget.....	69
IoT Custom Data Widget	70

IoT Donut Chart	70
IoT Line/Bar Chart	70
IoT Heatmap Chart	70
IoT Comparison Chart Widget	70
IoT Gauge Chart.....	71
O-Insights Reporting	72
IoT Analytics Report.....	72
O-Insights Web.....	73
IoT Single Value Widget	73
IoT Data List Widget.....	73
O-Insights Maps Plugin	74
Configuration of the Maps Plugin	74
Smart Client Views for Adding Sensor Values to Maps.....	75
FAQs.....	77

Introduction

Introduction to O-Insights

O-Insights™ is a suite of applications for XProtect that monitors system health, ensures compliance, enhances security, boosts operational efficiency, and streamlines VMS management. It features an Agile Workspace where users can create and customize their workspaces as needed. This solution offers exceptional control over video management operations by incorporating advanced visualization tools and detailed reporting capabilities. Key features include customizable dashboards, real-time monitoring, historical data analysis, and advanced reporting options.

The O-Insights for IoT umbrella comprises the following applications:

- **O-Insights IoT Drivers:** Enables the integration of various IoT devices, such as cameras and sensors, to collect real-time data.
- **O-Insights Config Tool:** Streamlines the configuration and management of points and alarms within the XProtect Smart Client user interface.
- **O-Insights Maps Plugin:** Visualizes IoT data on XProtect maps, providing an intuitive way to monitor and manage connected devices in real-time.
- **O-Insights Publish Topics Plugin:** Handles notification of XProtect alarms and events via MQTT.
- **IoT Driver Configurator:** Offers detailed driver information and configuration options.
- **Query/Reporting Engine for O-Insights:** Delivers powerful data querying capabilities to support advanced analytics.
- **O-Insights Plugin for VMS:** Seamlessly integrates with Milestone XProtect, creating a unified platform for video management and metadata analysis.

- **O-Insights Web:** Allows users to create and review dashboards and reports easily and directly from a web browser.

Features

- **Data Manager Service:** Responsible for storing historical data related to points, camera analytics, and IoT sensor values.
- **IoT Config Tool:** A new XProtect plugin for configuring points and sensors and creating alarms in XProtect. This tool enhances efficiency by streamlining the setup of data points.
- **Widgets for IoT Data Visualization:** Allows users to combine multiple widgets to create comprehensive dashboards for monitoring and analysis.
- **MQTT Broker:** O-Insights IoT Driver can now function as an MQTT broker, facilitating data exchange between clients.
- **Webhooks:** O-Insights IoT Driver now supports webhooks, enabling real-time data transfer to the IoT driver.
- **OPC UA:** Provides a standardized and secure communication framework, complementing features like the MQTT Broker and Webhooks for efficient data exchange and real-time IoT integration in XProtect environments.

Technical Overview

About MQTT

MQTT (Message Queuing Telemetry Transport) is a lightweight, publish-subscribe messaging protocol designed for constrained devices and low-bandwidth, high-latency networks. It is widely used in IoT (Internet of Things) environments to facilitate communication between devices and a central server or cloud platform. MQTT is efficient in its use of network resources, making it ideal for real-time data transfer, such as sensor readings or status updates.

MQTT operates by allowing devices to publish messages to specific topics, which other devices can subscribe to, enabling seamless data exchange in a scalable and flexible manner.

Key Advantages of Using MQTT Over Direct Metadata Retrieval from Cameras/Sensors/IoT Devices:

- **Scalability:** MQTT is highly effective in managing numerous connected devices and applications. The publish-subscribe model efficiently distributes data from various publishers (such as sensors and cameras) to a broad subscriber base without overloading the original data sources. This decentralized approach enhances system reliability and scalability.
- **Clarity:** MQTT uses JSON for efficient data transfer. The human-readable format of JSON simplifies data parsing and interpretation, thereby improving overall system efficiency.
- **Lightweight:** MQTT optimizes resource usage by decoupling data production from data consumption. Cameras, sensors, and IoT devices publish metadata to a central broker, which then distributes it to multiple clients without placing excessive strain on the devices themselves.

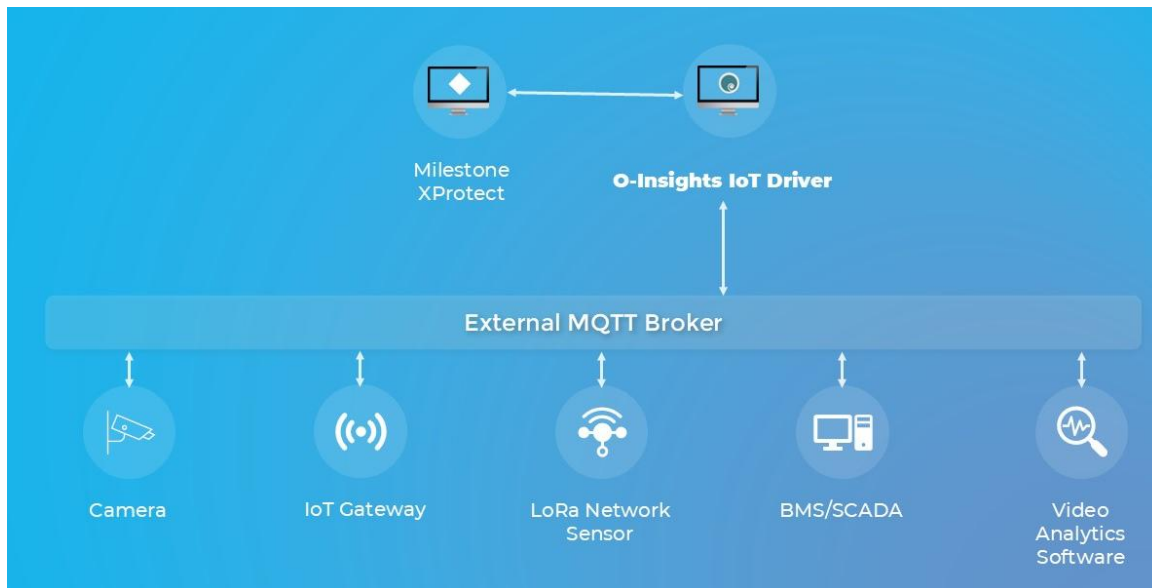
Key Components:

- **Clients:** These include IoT devices, sensors, gateways, SCADA systems, and cameras that transmit data using the MQTT protocol.

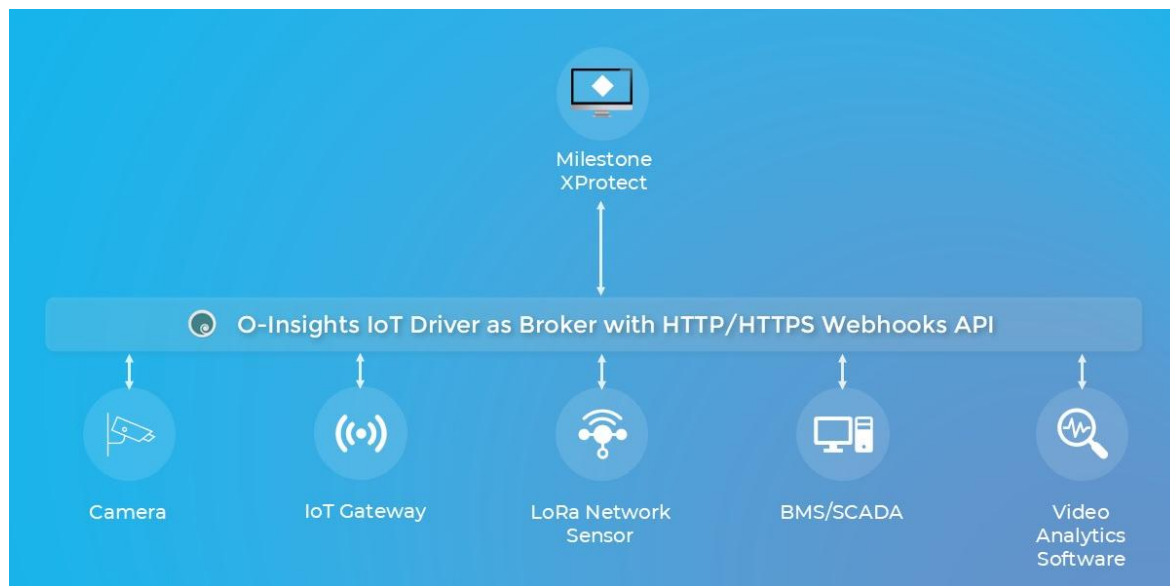
- **MQTT Broker:** This is the software that acts as a server or hub, distributing MQTT messages to subscribing clients and receiving MQTT messages from cameras.

Scenarios:

1. **External Broker:** An external broker is used for the MQTT network.
2. **O-Insights as Broker:** O-Insights IoT Driver itself serves as the broker for the MQTT network.



O-Insights as MQTT Broker



O-Insights IoT Driver can also function as an MQTT broker, managing point data exchange between clients within the O-Insights MQTT broker configuration.

The Client-Broker architecture implemented in O-Insights is illustrated above.

The O-Insights MQTT Publish Topics plugin, available within the XProtect Management Client, is used to create topics for XProtect Events and Alarms which will be sent over MQTT.

About Webhooks

- **Webhooks** are automated HTTP/HTTPS callbacks that are triggered by specific application events, such as data updates, system status changes, or user actions. When an event occurs, a predefined URL is notified, often including relevant data payloads. This mechanism is essential for enabling real-time updates and ensuring seamless integration across various IoT systems, allowing different components to communicate and dynamically respond to changes.
- **In the context of O-Insights**, webhooks are crucial for managing data flow and ensuring that all relevant systems stay up to date without manual intervention. Whenever a data point captures or generates new information, a webhook is automatically triggered. This webhook then

sends the updated information to a designated endpoint, such as O-Insights IoT driver.

- This process ensures that the IoT driver receives the latest data in real-time, enabling it to take immediate action based on the incoming information. For instance, if a sensor detects a change in environmental conditions, the webhook will instantly transmit this data to the IoT driver, which can then adjust operations, trigger alerts, or update the system dashboard as needed.

About OPC UA

OPC UA (Open Platform Communications Unified Architecture)

Overview

Open Platform Communications (OPC) is a set of standards and protocols designed to enable interoperability between industrial automation devices and systems, facilitating reliable data exchange. OPC was originally developed to standardize communication between industrial hardware and software from different manufacturers, allowing them to share real-time data seamlessly. This protocol is widely adopted in various industries, including manufacturing, energy, and building automation, where it ensures efficient and secure data integration. The O-Insights IoT driver provides robust and bidirectional communication enabling secure data exchange between OPC UA-enabled devices and Milestone XProtect. It ensures that data can be both sent and received seamlessly, making it suitable for dynamic IoT ecosystems.

Key Components

OPC Server: Acts as the central data source, collecting information from devices like sensors, controllers, and other data points. The OPC server standardizes this data and makes it accessible to authorized clients.

OPC Client: Any application or system that requires access to the OPC server data. Clients can read from or write data to the server, enabling actions based on real-time information.

OPC Standard

OPC Unified Architecture (UA): The more advanced OPC UA standard provides a platform-independent, scalable, and secure framework. It supports

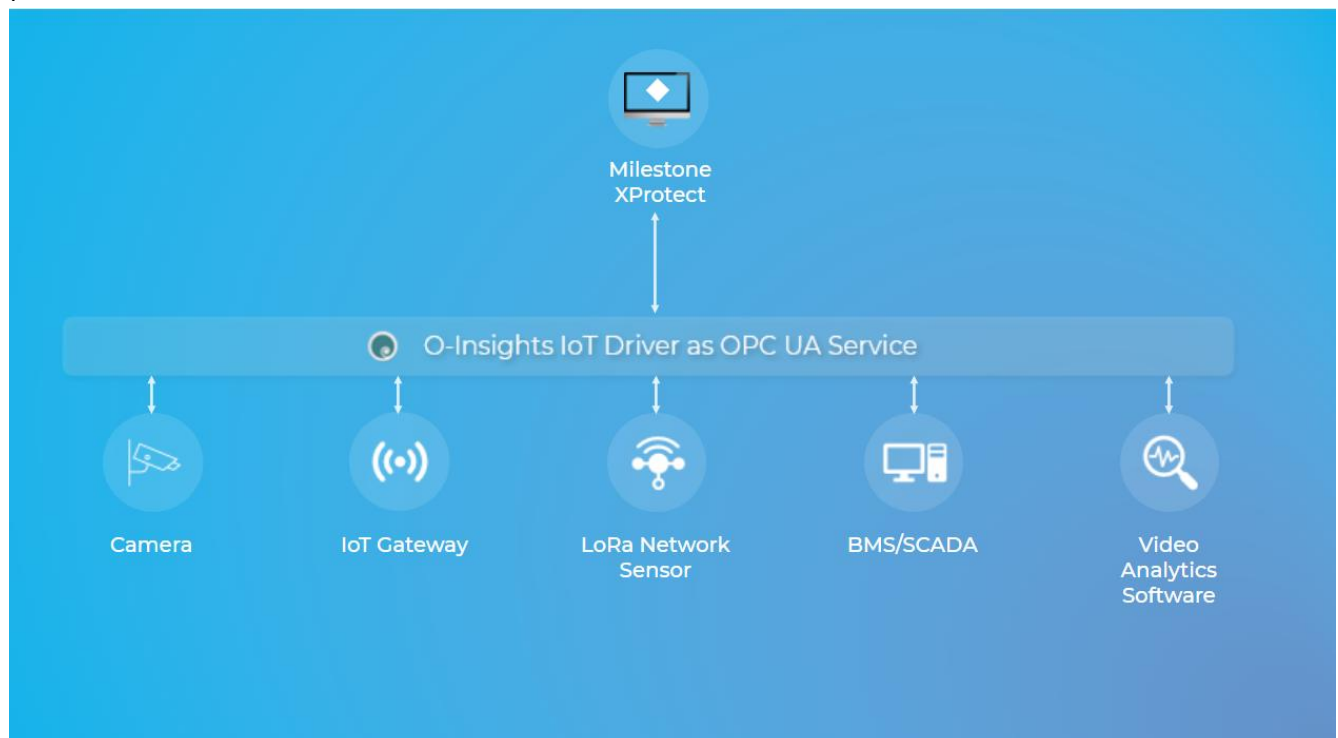
complex data structures, cross-platform connectivity, and enhanced security features, making it suitable for modern industrial applications and IoT environments.

Advantages of OPC

- **Interoperability:** OPC enables seamless communication between different devices and systems, reducing integration costs and improving operational efficiency.
- **Scalability:** Particularly with OPC UA, it can accommodate a wide range of applications, from small device networks to large industrial systems.
- **Security:** OPC UA incorporates encryption, authentication, and user authorization features, ensuring secure data transmission.

In context of O-Insights

In the context of O-Insights, 'OPC' refers to the **Open Platform Communications** protocol, specifically the OPC UA (Unified Architecture) standard. O-Insights leverages OPC UA to facilitate seamless, bidirectional communication between Milestone XProtect Video Management Systems (VMS) and various Building Management Systems (BMS), Supervisory Control and Data Acquisition (SCADA) systems, and other industrial automation platforms.



Prerequisites

- **Microsoft Visual C++ Redistributable 2019 Update 9 (Version 14.27.29903) or later**
- **.NET Runtime 8 or later**
- **ASP.NET Core Runtime 8 or later**
- **.NET Framework 4.8 or later**
- **Milestone XProtect 2023 R2 or later**
- **O-Insights Query Engine v5.1** (required for IoT widgets and reports)
- **Required Ports:** 8090, 8091, 8094, 8093, 8095, and 48030 must be opened


Port Number	Service
8090	O-Insights IoT driver
8091	O-Insights OPC UA Service
8093	Webhook Endpoint
8094	O-Insights MQTT driver
8095	O-Insights Data manager service
9011	O-Insights Query engine service
48030	OPC Port (Default Port for OPC)
1883,1884 (MQTTS)	MQTT Broker of O-Insights MQTT driver

Note: The installation package includes MongoDB v7.0.12, simplifying the setup process.

Note: O-Insights IoT Driver has broadened compatibility by now supporting Milestone basic user accounts in addition to Windows-based accounts to connect to the XProtect Management Server

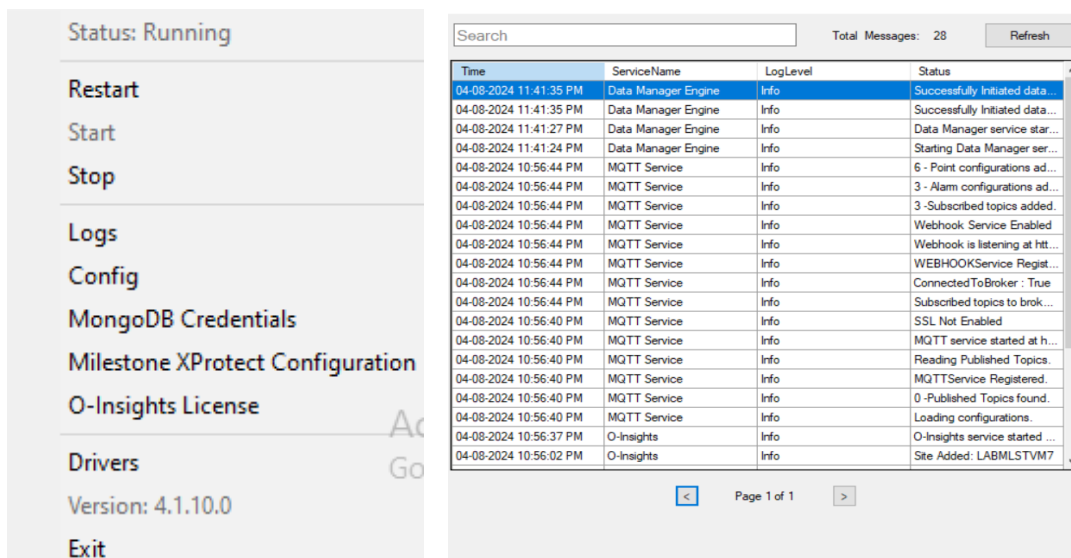
Drivers

IoT Configurator Tool

The IoT Configurator Tool provides a centralized interface for managing and configuring the O-Insights IoT Driver. Accessible from the system tray () , it allows users to monitor the status of the Data Manager Service, manage service operations (such as starting, stopping, and restarting the service), and view event logs. The tool also facilitates configuring essential components like MongoDB credentials and Milestone XProtect settings, ensuring proper communication between the IoT service and Milestone XProtect. Additionally, users can view service versions, check the IoT Driver license, and manage connected drivers—all from a single, convenient location.

Default path of the Configurator Tray Tool: C:\Program Files\O-Insights IoT Driver\IoTDriverConfigurator.exe

The available options are explained in the *Data Manager Service* section.



The screenshot displays the IoT Configurator Tool interface. On the left, a sidebar shows the status as 'Running' and lists several actions: Restart, Start, Stop, Logs, Config, MongoDB Credentials, Milestone XProtect Configuration, O-Insights License, Drivers, Version: 4.1.10.0, and Exit. The main area on the right features a search bar, a 'Total Messages: 28' indicator, and a 'Refresh' button. Below these is a table with four columns: Time, ServiceName, LogLevel, and Status. The table contains 28 entries, with the first row highlighted in blue. The last row shows 'Site Added: LABMLSTM7'.

Time	ServiceName	LogLevel	Status
04-08-2024 11:41:35 PM	Data Manager Engine	Info	Successfully Initiated data...
04-08-2024 11:41:27 PM	Data Manager Engine	Info	Successfully Initiated data...
04-08-2024 11:41:24 PM	Data Manager Engine	Info	Data Manager service star...
04-08-2024 10:56:44 PM	Data Manager Engine	Info	Starting Data Manager ser...
04-08-2024 10:56:44 PM	MQTT Service	Info	6 - Point configurations ad...
04-08-2024 10:56:44 PM	MQTT Service	Info	3 - Alarm configurations ad...
04-08-2024 10:56:44 PM	MQTT Service	Info	3 - Subscribed topics added.
04-08-2024 10:56:44 PM	MQTT Service	Info	Webhook Service Enabled
04-08-2024 10:56:44 PM	MQTT Service	Info	Webhook is listening at htt...
04-08-2024 10:56:44 PM	MQTT Service	Info	WEBHOOKService Regist...
04-08-2024 10:56:44 PM	MQTT Service	Info	ConnectedToBroker : True
04-08-2024 10:56:44 PM	MQTT Service	Info	Subscribed topics to brok...
04-08-2024 10:56:40 PM	MQTT Service	Info	SSL Not Enabled
04-08-2024 10:56:40 PM	MQTT Service	Info	MQTT service started at h...
04-08-2024 10:56:40 PM	MQTT Service	Info	Reading Published Topics.
04-08-2024 10:56:40 PM	MQTT Service	Info	MQTTService Registered.
04-08-2024 10:56:40 PM	MQTT Service	Info	0 -Published Topics found.
04-08-2024 10:56:40 PM	MQTT Service	Info	Loading configurations.
04-08-2024 10:56:37 PM	O-Insights	Info	O-Insights service started ...
04-08-2024 10:56:02 PM	O-Insights	Info	Site Added: LABMLSTM7

IoT Service

The **IoT Service** is the core service responsible for facilitating communication between IoT devices and the XProtect. This service acts as a bridge, ensuring seamless data exchange between connected devices and the XProtect.

Note: O-Insights IoT license is required for the complete usage of services. Your License file will be located at *<Installed directory>\O-Insights IoT Driver\License*

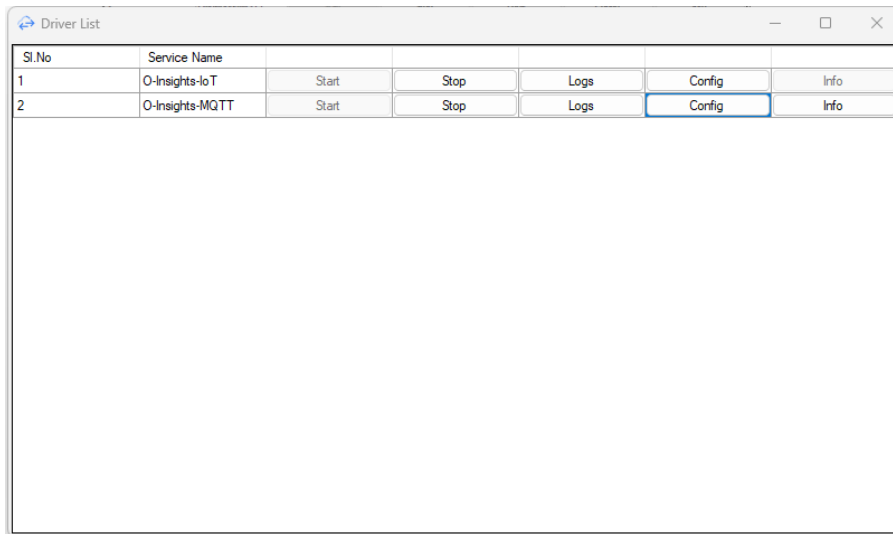
Note: O-Insights IoT license is based on number of cameras and license for type of Driver.

Configuring IoT Service

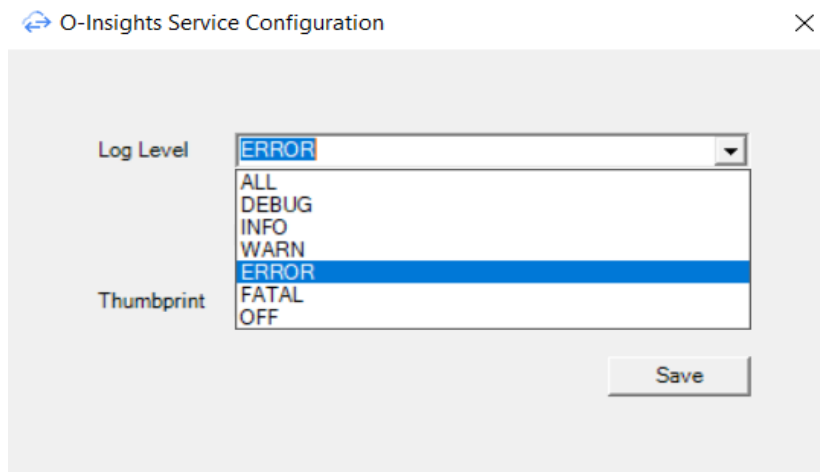
To ensure optimal performance and seamless integration with the XProtect, the O-Insight IoT Service requires proper configuration. Users can configure this service through the Configurator Tool's user-friendly interface, which offers straightforward options for setting up basic parameters. For more advanced customization, users can also directly modify the configuration file, allowing fine-tuning of the service to meet specific operational needs. The following sections detail the configuration settings available both through the UI and within the advanced settings of the config file.

Configurator Tool

- **From the driver service list, users can update and configure the O-Insights service.**
- **To configure the O-Insights service:**



- *Start*: Initializes the O-Insights service.
- *Stop*: Stops the service.
- *Logs*: Opens the log file for the O-Insights service. Log files provide a detailed record of system activities.
- *Config*: Opens the config window as shown below:



- **Log Level**: Log levels categorize the content of the log files. By default, it is set to *ERROR*, meaning every error of the O-Insights Service will be recorded in the log file. This setting can be adjusted to capture additional details for troubleshooting purposes.
 - Other log levels include:
 - *ALL*
 - *DEBUG*
 - *INFO*
 - *WARN*

- *ERROR*
- *FATAL*
- *OFF*
- To enable encryption, check the *Enable Encryption* box and input the corresponding certificate's thumbprint in the designated text field.
- To apply changes, save your configuration and then restart the service. This ensures that the new settings take effect.

Config File (Advanced)

Location to the config file: <Installation Directory>\O-Insights IoT Driver\O-Insights Service\OInsights.exe.config

Key	Description	Default Value
OInsightsServerPort	Port used by O-Insights IoT driver	8090
EnableSSL	To enable encryption for O-Insights IoT driver. To Enable Encryption set to 1 else 0	0
SslCertificateThumbprint	Thumbprint of SSL certificate	
VMSServer	XProtect Management server address	http://localhost
SecureOnly	Connection to VMS secure or not	false
VMSLoginType	User type to login to XProtect	LogonUser/Basic User

RetryCount	Number of retries for login	5
RetryDelay	Delay for each retry in seconds	30
ConnectionTimeout	Timeout in seconds for XProtect connection during login	30
IncludeChildSites	Whether need to include child sites	false
MLSTBasicUsername	BasicUser name, if login type is BasicUser	
MLSTPassword	Encrypted password, if login type is BasicUser	
EventListener	Whether need to listen to events	true
AlarmListener	Whether need to listen to Alarms	true
CacheUpdateCronTime	Cron expression for updating camera cache	0 0 11/1 * ? * (At 01:00 AM Everyday)
IgnoreEvents	Events to be ignored (Comma separated names)	
IgnoreAlarms	Alarms to be ignored (Comma separated names)	

TriggerAlarms	If this option is set to True, then the IoT driver will create an Alarm in XProtect	false
AlarmClass	Alarm class for generating alarm	
AlarmType	Alarm type for generating alarm	
DefaultAlarmMessage	Alarm message while generating alarm	
AlarmName	Alarm name when generating alarm	
AlarmPriority	Priority of alarm when generating alarm	1
AlarmPriorityName	Priority name of alarm when generating alarm	High
AlarmStateName	Alarm state name when generating alarm	New
AlarmState	Alarm State when generating alarm	1
TriggerUserDefinedEventsBasedOn	For auto trigger events in XProtect, this key will trigger a user defined event based on a match on Event, Source or Source_Event (Used in OPC UA driver)	

DataVisible	To disable authentication for point value API	false
--------------------	---	-------

MQTT Service

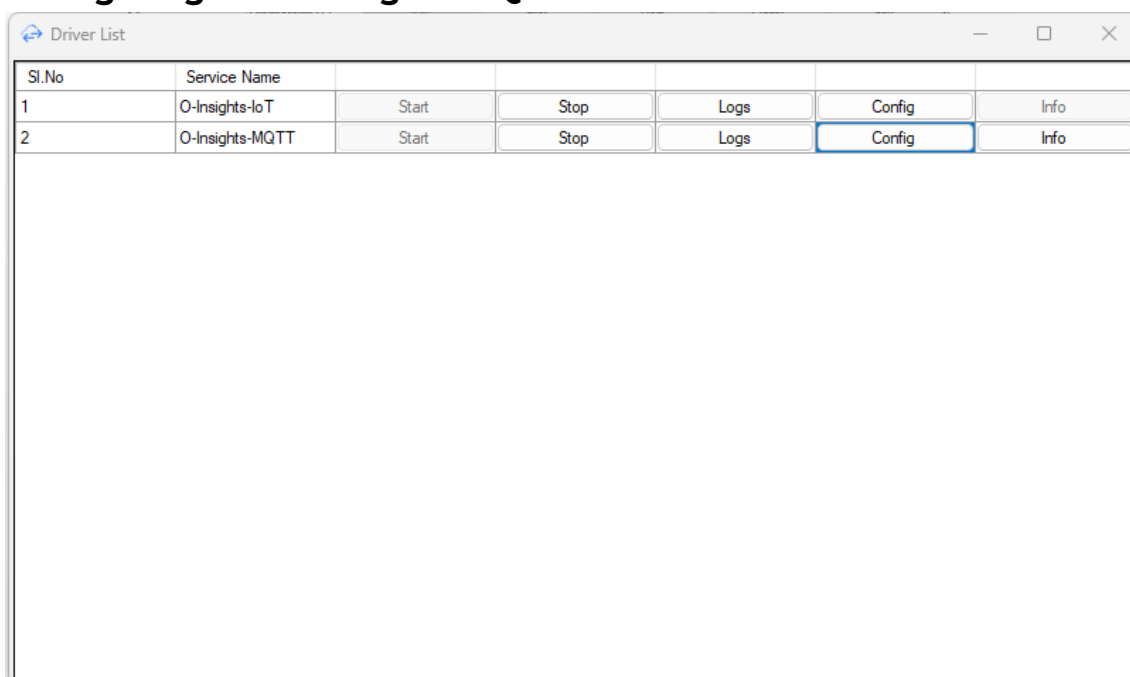
The **MQTT Service** serves as the communication layer between the O-Insights IoT Driver and MQTT-enabled devices or data sources. This service is responsible for managing the exchange of MQTT data, ensuring that data flows efficiently between IoT devices and the IoT Driver.

Configuring MQTT Driver

To ensure optimal performance and seamless integration with the XProtect, the O-Insights MQTT driver requires proper configuration. Users can configure this driver through the Configurator Tool's user-friendly interface, which offers straightforward options for setting up basic parameters. For more advanced customization, users can also directly modify the configuration file, allowing for fine-tuning of the service to meet specific operational needs. The following sections detail the configuration settings available both through the UI and within the advanced settings of the config file.

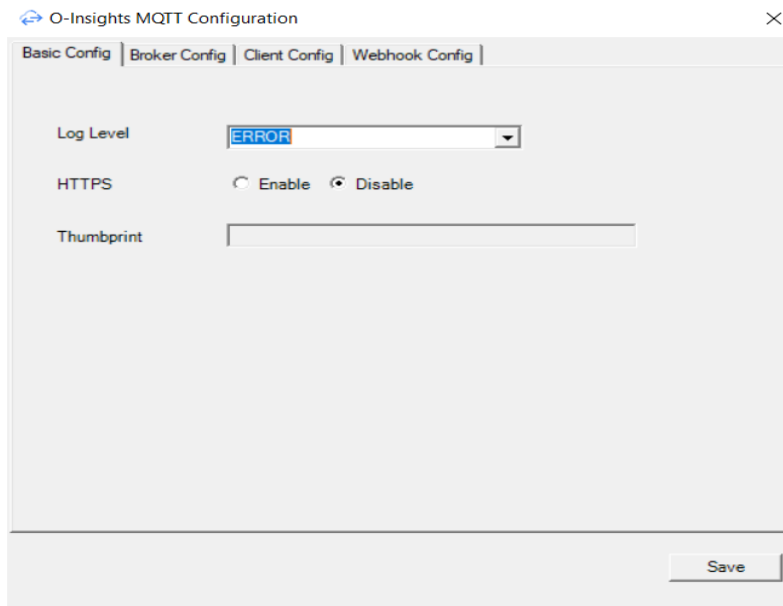
Configurator Tool

- **From the driver list, users can update and configure the O-Insights-MQTT driver.**
- **Configuring the O-Insights MQTT Driver**



- *Start*: Initializes the O-Insights MQTT driver.
- *Stop*: Stops the driver.
- *Logs*: Opens the log file for the O-Insights MQTT driver. Log files provide a detailed record of system activities.
- *Config*: Opens the config window.

- **Basic Config**



- In the basic config tab:
 - *Log Level*: Log levels categorize the content of the log files. By default, it is set to *ERROR*, meaning every error of the O-Insights-MQTT driver will be recorded in the log file. This setting can be adjusted to capture more detailed troubleshooting information.
 - Other log levels include:
 - *ALL*
 - *DEBUG*
 - *INFO*
 - *WARN*
 - *ERROR*
 - *FATAL*
 - *OFF*
 - To enable HTTPS, click on *Enable* and input the corresponding certificate's thumbprint in the designated text field. This enables encryption for both MQTT Driver and Webhook.

- **Broker Config**

The screenshot shows the 'O-Insights MQTT Configuration' window with the 'Broker Config' tab selected. The window has four tabs: 'Basic Config', 'Broker Config', 'Client Config', and 'Webhook Config'. The 'Broker Config' tab contains the following settings:

- MQTT Broker:** ☒ Enable, ☐ Disable
- MQTT Host Address:** Text field containing 'localhost'
- MQTT Port:** Text field containing '1883'
- MQTTS:** ☐ Enable, ☒ Disable
- Thumbprint:** Empty text field
- User Authentication:** ☐ Enable, ☒ Disable
- Username:** Empty text field
- Password:** Empty text field
- ☐ Show Password
- * MQTT Client ID is mandatory to connect to MQTT broker
- Save** button

- In the broker config tab:
 - To enable the MQTT Broker, click on *enable*.
 - The MQTT host address should specify the location of the MQTT broker, such as an IP address, a domain name, or a hostname.
 - In the MQTT Port field, input the corresponding port number of the MQTT broker.
 - To enable MQTTS, click on *Enable*.
 - MQTTS is the secure version of the MQTT protocol, encrypting data transmission using SSL/TLS.
 - In the thumbprint text field, input the corresponding certificate's thumbprint.
 - If MQTTS is enabled, the default port will be 1884.
 - To enable user authentication, click on *enable*.
 - This verifies the identity of a connecting client before granting access to the broker.
 - Input the corresponding username and password in the text fields.
 - *Note:* A client ID is mandatory to connect to the MQTT broker.

- **Client Config**

The screenshot shows the 'O-Insights MQTT Configuration' window with the 'Client Config' tab selected. The window has four tabs: 'Basic Config', 'Broker Config', 'Client Config', and 'Webhook Config'. The 'Client Config' tab contains the following fields and controls:

- MQTT Host Address:** A text field containing 'localhost'.
- MQTT Port:** A text field containing '1883'.
- MQTTS:** Radio buttons for 'Enable' and 'Disable'. The 'Disable' button is selected.
- User Authentication:** A section containing:
 - Username:** An empty text field.
 - Password:** An empty text field.
 - Show Password:** A checkbox that is currently unchecked.
- Test Connection:** A button located at the bottom right of the configuration area.
- Save:** A button located at the bottom right of the window.

- The MQTT host address specifies the network location of the MQTT broker. This combined with the port number, determines where clients should connect to send and receive messages.
 - The MQTT Port field will be pre-filled with the default port for MQTT.
 - 1883: For unencrypted connections.
 - 1884: For encrypted connections (MQTTS over TLS)
- User authentication:
 - Enter the corresponding client username and password in the text fields.
 - Proper configuration of user authentication is crucial for securing MQTT communication and preventing unauthorized access.
- Test Connection: The *Test Connection* button can be used to verify the Broker's Credentials.

- **Webhook Config**

The screenshot shows a window titled "O-Insights MQTT Configuration" with a close button (X) in the top right corner. The window has four tabs: "Basic Config", "Broker Config", "Client Config", and "Webhook Config", with the "Webhook Config" tab currently selected. Inside the "Webhook Config" tab, there is a "Webhook" section with two radio buttons: "Enable" (which is selected) and "Disable". Below this, there are three text input fields: "URL" (containing "https://192.168.0.27:8093/webhook/receive"), "Auth Key" (containing "AuthKey"), and "Auth Value" (containing "JVhwJrH578A3uM03"). To the right of the "URL" field is a "Copy URL" button. At the bottom right of the window is a "Save" button.

- To enable webhooks, click on *Enable*.
- URL specifies the target URL for data transmission upon event occurrence. Edit the URL if required.
- *Copy URL*: Copies the entire URL to the system clipboard for subsequent use.
- *Auth Key*: An authentication key is required to validate webhook requests.
- *Auth Value*: An Auth Value is a secret key or token used to validate the authenticity of a webhook request. Input the Auth Value in the text field.
- Example of calling webhook using Postman:

POST ▼ | https://192.168.0.27:8093/webhook/receive

Params **Authorization** ● Headers (9) ● Body ● Pre-request Script Tests Settings ●

Type

API Key ▼

The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)

Key

AuthKey

Value

JVhwJrH578A3uMO3

Add to

Header ▼

After all required configurations have been performed, click on *Save*. Restart the MQTT driver to apply changes.

- **Info Tab**

- Provides comprehensive details about the MQTT driver, including Clients Connected, MQTT Messages, and Webhook Messages.

MQTT Details

Clients Connected to Broker 1 Refresh

Clients Connected MQTT Messages Webhook Messages

Sr.No.	Client ID
1	O-Insights-Client-774120338

- *Clients Connected*: Shows all the connected clients to the MQTT broker.

MQTT Details

Clients Connected to Broker: 1

Refresh

Clients Connected | MQTT Messages | Webhook Messages

Search

Time	Topic	Payload	QoS	Retain
21-08-2024 18:51:27	DESKTOP-MID9F07/Came...	{"TotalCameras":8,"CamerasOnlineCount":4,"CamerasO...	0	False
21-08-2024 18:51:27	DESKTOP-MID9F07/Came...	{ "Time": "2024-08-21T18:51:27.017Z", "CameraDetail...	0	False

Page 1 of 1

- *MQTT Messages*: Displays every topic received by the MQTT broker, containing information sent by MQTT clients.

MQTT Details

Clients Connected to Broker: 1

Refresh

Clients Connected | MQTT Messages | Webhook Messages

Time	Payload
21-08-24 06:57:34 PM	{ "temperature": 45, "humidity": 49, "object": { "rxInfo": { { "altitude": 0, "rssi": 55, "latitude": 0, "n...

- *Webhook Messages*: Shows webhook messages received by the IoT driver.

Config File (Advanced)

Location to the config file: <Installation Directory>\O-Insights IoT Driver\O-Insights MQTT\MQTTService.exe.config

Key	Description	Default Value
MQTTServerPort	Port used by O-Insights MQTT driver	8094
SendEventOnlyIfNotAlreadyInAlarm	This is set to True to prevent repeated triggering of user defined events when the sensor value is already in alarm	true
SetCulture	Set to true if want to change the default localisation culture of the IoT driver. Only for internal use	false
CultureString	Localisation culture value	en-US
TopicsDataVisible	If set to true, the APIs for getting point values can be accessed without authentication.	false
EnableSSL	To enable encryption for MQTT/Webhook driver. To enable encryption set to 1 else 0	0
SslCertificateThumbprint	Thumbprint of SSL certificate	
OInsightsServer	O-Insights IoT driver hostname/ IP address. If not present will default to local machine.	
DataManagerServer	Data manager service hostname/ IP address. If not present will default to local machine	
EnableWebhookService	To enable Webhook	true/false

WebhookServicePort	Port on which Webhook service listens	8093
WebhookAuthKey	Authentication key value	
WebhookMessageQueueSize	Number of messages received by Webhook to be displayed in Configurator tool	10
EnableBroker	To enable MQTT broker for the O-Insights MQTT driver	true/false
BrokerIP	IP of MQTT broker to be enabled	localhost
BrokerPort	Port for MQTT broker	1883
EnableAuth	Set true to enable authentication for the MQTT broker	
BrokerUserName	If EnableAuth is true, username for connection	
BrokerPassword	If EnableAuth is true, password for connection	
MQTTEnableSSL	Enable SSL for MQTT broker	false
MQTTCertThumbprint	Thumbprint of SSL certificate	
MQTTBrokerIP	Hostname/ IP address of MQTT Broker to connect	localhost

MQTTBrokerPort	Port of MQTT Broker to connect	1883
MQTTBrokerUserName	Username to connect to MQTT Broker	
MQTTBrokerPassword	Password to connect to MQTT Broker	
MQTTBrokerClientID	Default client ID	
ValidateCertificate	MQTT client requires validation of SSL Certificate	true
PublishTopicPrefix	When sending alarms and events from XProtect over MQTT, adds a prefix to every topic before publishing to MQTT Broker. Ensure topic ends with a "/"	
UserTopic	When sending alarms and events from XProtect over MQTT, a common topic will be used for all events and alarms when publishing to MQTT Broker	
MaxCameraPerPayloadCameraStatusPublish	Number of cameras in each payload when sending status. Details send include Camera Name, Model, Status, recording Server and Site Name	10
EnableCameraStatusPublish	To enable publishing camera status over MQTT	false

CameraStatusPublishIntervalInMins	Interval for camera status published in minutes	30
--	---	----

OPC UA Service

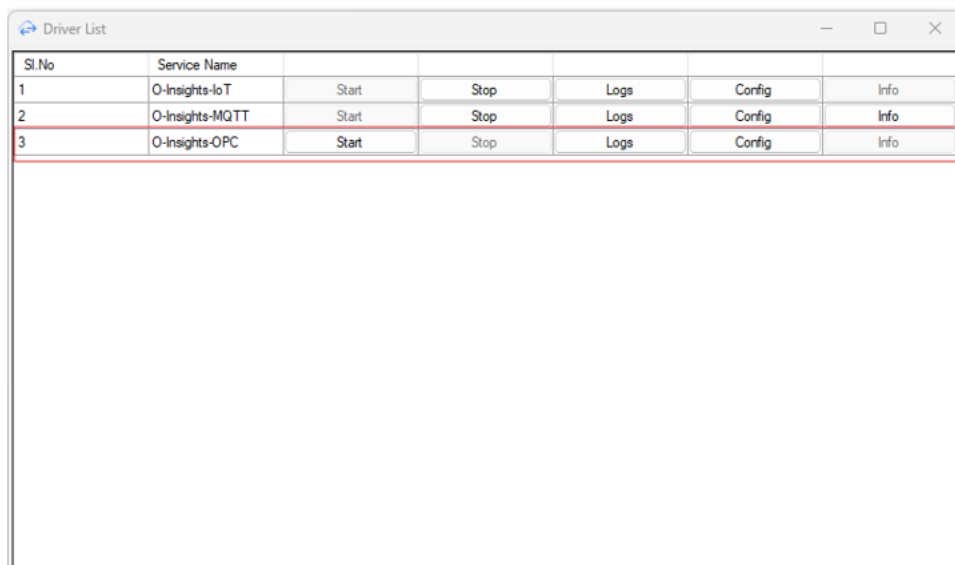
The OPC UA Service provides robust and bidirectional communication, enabling secure data exchange between the O-Insights IoT Driver and OPC UA-enabled devices or systems. It ensures that data can be both sent and received seamlessly, making it suitable for dynamic IoT ecosystems.

Configuring OPC UA Driver

The O-Insights IoT Config Tool now supports a new driver service, OPC UA, complementing the existing MQTT and Webhook drivers. The OPC UA driver extends the platform's capabilities, enabling users to configure OPC UA points and manage them seamlessly within the IoT Config Tool's interface. Once the three driver services—O-Insights IoT, O-Insights MQTT, and O-Insights OPC—are installed, the Driver List in the IoT Configurator Tool will display all available drivers for streamlined management.

Note: The OPC service in O-Insights is compatible only with OPC UA and does not support other versions of OPC.

Driver List



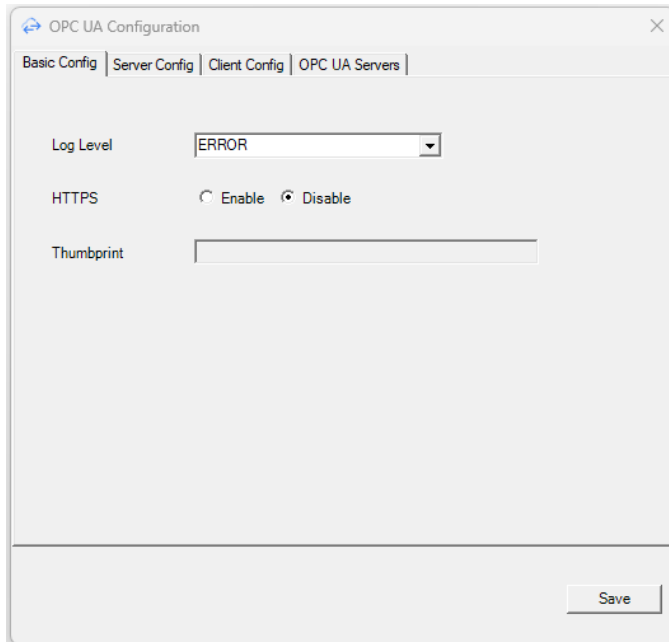
Sl.No	Service Name	Start	Stop	Logs	Config	Info
1	O-Insights-IoT	Start	Stop	Logs	Config	Info
2	O-Insights-MQTT	Start	Stop	Logs	Config	Info
3	O-Insights-OPC	Start	Stop	Logs	Config	Info

The **Driver List** displays the three available drivers and allows users to manage them through the following options:

- **Start:** Initializes the selected service.
- **Stop:** Stops the selected service.

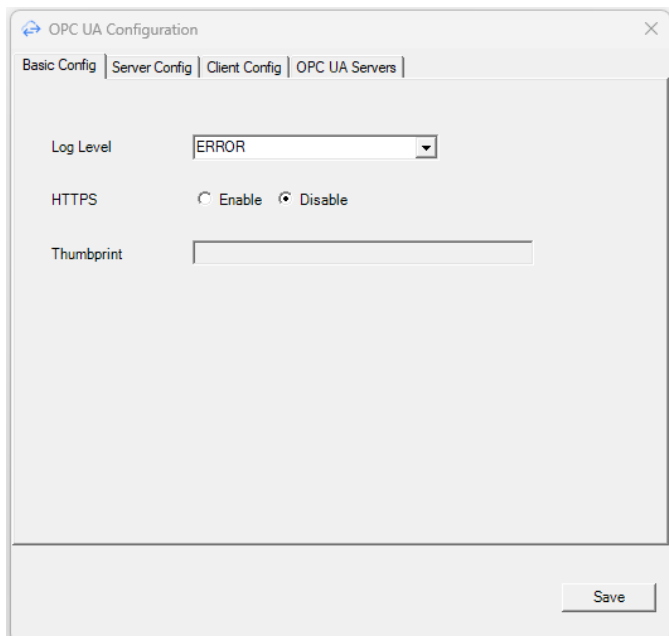
- **Logs:** Opens the log file for the selected service, providing a detailed record of system activities.

OPC UA Configuration



To configure the OPC UA driver, navigate to the *OPC UA Configuration* window in the IoT Driver Configurator. This interface consists of multiple tabs for setting up and managing the OPC UA service.

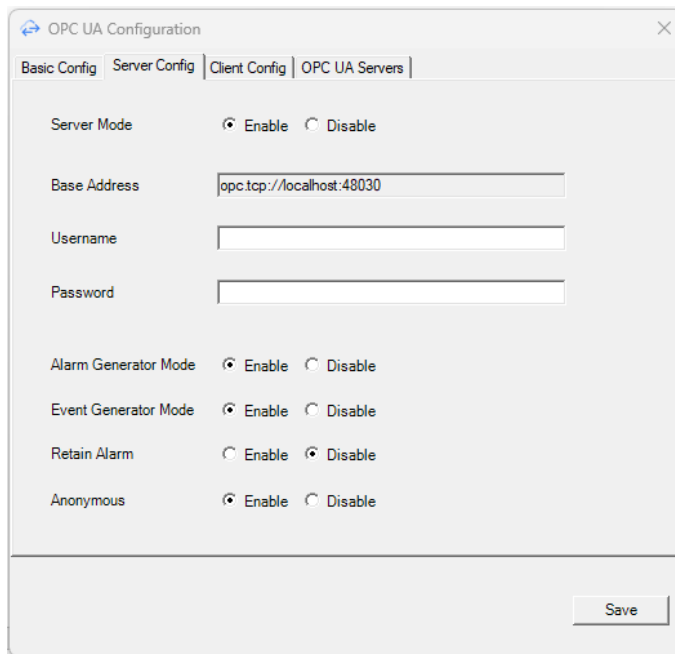
Basic Config



The *Basic Config* tab opens by default and includes the following fields:

- **Log Level:** Defines the level of detail for logging information generated by the OPC Service. Options include DEBUG, INFO, WARN, ERROR, etc., similar to the MQTT logging setup.
- **HTTPS:** Enables or disables SSL encryption for the OPC UA Service. To enable encryption, select *Enable* and enter the SSL certificate's *thumbprint* in the designated field. To enable HTTPS, click on *Enable* and input the corresponding certificate's thumbprint in the designated text field. This enables encryption for both MQTT Driver and Webhook.

Server Config



The screenshot shows the 'OPC UA Configuration' window with the 'Server Config' tab selected. The window has four tabs: 'Basic Config', 'Server Config', 'Client Config', and 'OPC UA Servers'. The 'Server Config' tab contains the following settings:

- Server Mode:** Radio buttons for 'Enable' (selected) and 'Disable'.
- Base Address:** Text field containing 'opc.tcp://localhost:48030'.
- Username:** Empty text field.
- Password:** Empty text field.
- Alarm Generator Mode:** Radio buttons for 'Enable' (selected) and 'Disable'.
- Event Generator Mode:** Radio buttons for 'Enable' (selected) and 'Disable'.
- Retain Alarm:** Radio buttons for 'Enable' and 'Disable' (selected).
- Anonymous:** Radio buttons for 'Enable' (selected) and 'Disable'.

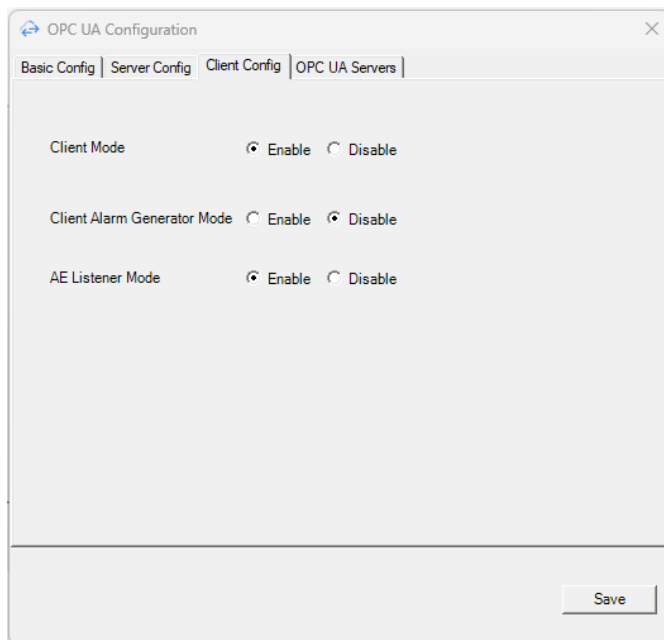
A 'Save' button is located at the bottom right of the window.

The *Server Config* tab allows the OPC service to operate in either *Server* or *Client* mode:

- **OPC Server Mode:** When enabled, the OPC service acts as an OPC UA server. The server's base address is fixed at `opc.tcp://[MACHINE_NAME]:48030`
- Milestone Alarms can be sent out as OPC UA notifications by enabling Alarm Generator mode in the driver. To enable the Alarm Generator mode, toggle the Alarm Generator mode.
- To transfer Milestone Events as OPC UA notifications, enable Event Generation by toggling the Event Generator mode.

- To prevent Alarm entries from auto closing when an Alarm is closed this option needs to be enabled. They can be retained by enabling the *Retain Alarm* option.
- The OPC UA driver supports the following Authentication Modes.
 - Anonymous
 - Allow OPC UA Clients to connect anonymously to the server.
 - Username/Password
 - Allows the OPC UA Clients to connect using username and password configured in the server.
- The Username/Password mode allows the user to decide whether authentication is required. If the User wants to enable authentication, then the User should enter the Username and Password in the Server Mode. The service will then use these credentials to authenticate the user. If no username and password are entered, the service will operate without requiring authentication.
- To save the above options, click on the “Save” button at the top-right corner. A success notification will be shown once the settings are saved.
- Now restart the O-Insights OPC UA Service, followed by the O-Insights Service.

Client Config

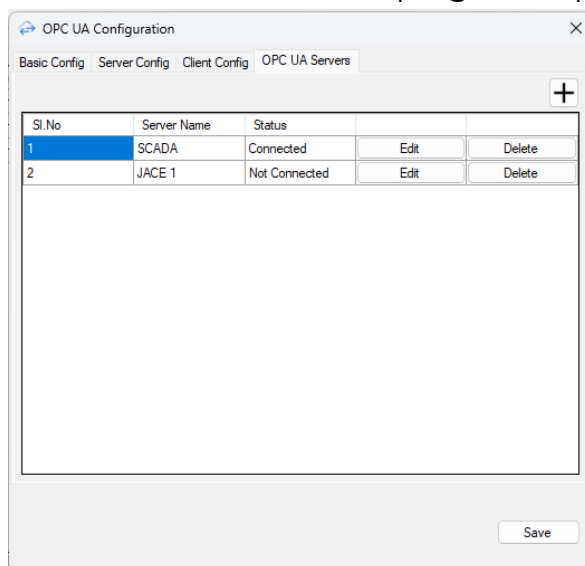


The *Client Config* tab configures the OPC service to act as a *Client* connecting to external OPC UA servers. When in client mode, the OPC service can read values from connected servers and trigger events based on alarm conditions.

Adding OPC Servers

To connect to an external OPC server, go to the *OPC Servers* tab:

1. Click the **+** icon at the top right to open the *Add Server* window.



2. Fill in the following details:

- Server Endpoint
 - The base address of the server.
- Namespace Uri
 - Namespace for nodes defined in the server.
- Security Policy
 - To view the Security Policies in the drop down, the Hostname of the OPC server must be added to the hosts file which is in C:\Windows\System32\drivers\etc folder.
 - The Security Policy to use when securing messages.
 - Supports three policies:
 - Basic256Sha256
 - Aes128_Sha256_RsaOaep
 - None
- Message Security Mode
 - The type of security to apply to the messages.
 - Supports three modes:
 - SignAndEncrypt
 - Sign
 - None
- Authentication Mode
 - Two modes:
 - Anonymous: Allows anonymously to connect to the server.
 - Username/Password: Accepts username and password for the connection.
- Send Inactive
 - Milestone by default will show active alarm entries. But enabling the *Send Inactive* option will allow Milestone to show inactive/normalized alarm entries also. To get the normalized Alarm entries in Milestone, toggle the *Send Inactive* option.

After entering the necessary details, use the *Test Connection* button to verify connectivity. Once confirmed, click *Add* to save the server configuration. The added server will now be listed as an option in the *OPC Server* dropdown within the *Add Point* window.

Note: Map the Host Name of OPC UA Server to the Host file of machine running O-Insights OPC UA Driver.

Config File (Advanced)

Location to the config file: <Installation Directory>\O-Insights IoT Driver\O-Insights OPC\OInsightsOPC.exe.config

Key	Description	Default Value
OPCWebServerPort	Port used by O-Insights OPC UA driver	8091
ServerMode	To enable ServerMode in OPC Service	true/false

ClientMode	To enable ClientMode in OPC Service	true/false
AlarmGeneratorMode	Alarm Generator (Dependent on ServerMode)	true/false
EventGeneratorMode	Event Generator (Dependant on ServerMode)	true/false
ClientAlarmGeneratorMode	Alarm Generator (Dependant on Client Mode)	true/false
AlarmListenerMode	Event/Alarm Listener (Dependent on Client Mode)	true/false
OInsightsServer	O-Insights IoT driver hostname/ IP address. If not present will default to local machine	
DataManagerServer	Data manager service hostname/ IP address. If not present will default to local machine.	
OPCWebServerPort	OPCWebServerPort	8091
Retry	Retry calling above APIs	1
WaitInterval	Wait Interval for Retry (in milliseconds)	1000
RetainAlarm	Retain Normalized Alarms in the Alarm View.	true/false
ExportNodeDetails	Settings for exporting the exposed node details	true/false

PointsDataVisible	If set to true, the APIs for getting point values can be accessed without authentication.	true/false
EnableSSL	To enable encryption for MQTT/Webhook driver. To enable encryption set to 1 else 0	0/1
SslCertificateThumbprint	Thumbprint of SSL certificate	
OPCServerUsername	Username for authentication	
OpcServerPassword	Password for authentication	

- In Client mode, the Driver can also be configured to Auto Trigger Alarms in Milestone XProtect in addition to an Event. The configuration is part of O-Insights Service\OInsights.exe.config
- To Turn on Alarm creation:
`<add key="TriggerAlarms" value="True"/>`
- Alarm Properties that can be configured:
`<add key="AlarmType" value="MLST"/>`
`<add key="DefaultAlarmMessage" value="MLST Alarm"/>`
`<add key="AlarmName" value=""/>`
`<add key="AlarmPriority" value="1"/>`
`<add key="AlarmPriorityName" value="High"/>`
`<add key="AlarmStateName" value="New"/>`
`<add key="AlarmState" value="1"/>`

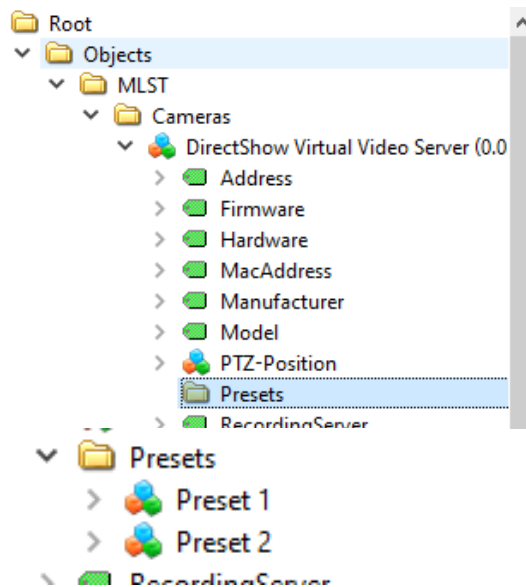
OPC UA Service as a Server

Our OPC UA Service can act as a Server. Camera/recording sever status and other properties are exposed along with Camera/server facts (total count, online count, offline count). User defined events configured in the XProtect system are exposed when running in server mode which can be triggered from the OPC UA Client. All the alarms /events occurring in the XProtect

system are also exposed in server mode which can be read by any OPC UA client. The SCADA / OPC UA client needs to listen to all alarms and events exposed by the OPC UA Server (XProtect). Server mode also allows control of camera presets, zoom level, and position in case of PTZ cameras.

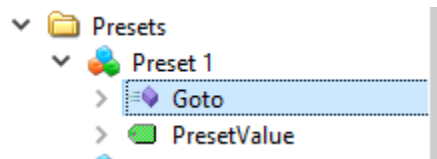
Camera Presets

All camera Presets can be viewed under the **Preset** folder.



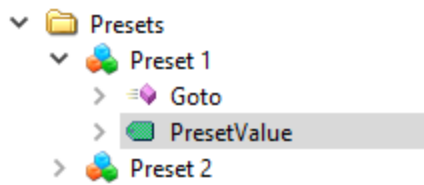
Activating Presets By calling “Goto” method.

Under each preset **Goto** method is there. By calling that method, that preset will be activated.



Activating Presets By calling “Goto” method.

Under each preset, there is a “**PresetValue**”.



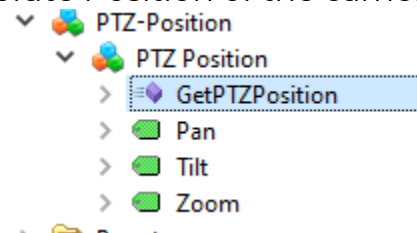
This can be viewed in the data access view, and by changing its value to “1” we can activate that preset.

Data Access View					
#	Server	Node Id	Display Name	Value	Datatype
1	CTS OlnsightsO...	NS2 String d776...	PresetValue	0	Double

PTZ-Position

Get Absolute position of PTZ Camera

Under the PTZ-Position object, the “**GetPTZPosition**” method to get the Absolute Position of the camera.



While calling this method, we will get its position in Pan, Tilt, and Zoom variables in Data Access View.

Changing Position

Values of Pan, Tilt, and Zoom variables can be changed in Data View

Data Access View					
#	Server	Node Id	Display Name	Value	Datatype
1	CTS OlnsightsO...	NS2 String d776...	Pan	0	Double
2	CTS OlnsightsO...	NS2 String d776...	Tilt	0	Double
3	CTS OlnsightsO...	NS2 String d776...	Zoom	0	Double


















Pan: Acceptable values range from -1 to 1.

Tilt: Acceptable values range from -1 to 1.

Zoon: Acceptable values range from m 0 to 1.

Streaming Properties

The Default Streaming properties of each camera can be seen under “**StreamingProperties**” folder. Values can be viewed in Datta Access View

- ▼  StreamingProperties
 - >  - Media profile
 - >  Codec
 - >  Edge storage support
 - >  Frames per second
 - >  Keep Alive type
 - >  Max. frames between keyframes
 - >  Max. frames between keyframes mode
 - >  Maximum bit rate (kbit/s)
 - >  Multicast address
 - >  Multicast force PIM-SSM
 - >  Multicast port
 - >  Multicast time to live
 - >  Quality
 - >  Resolution
 - >  Stream reference ID
 - >  Streaming method

Camera Cache Update

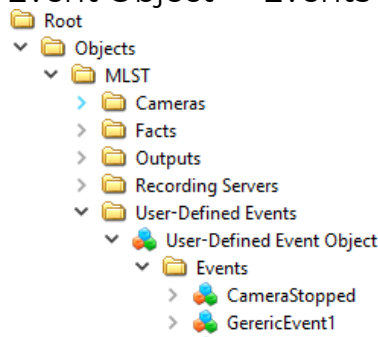
Camera updates will update all variable details like Address, resolution, FPS, etc. Cache update interval can be configured in “**OInsights.exe.config**” file.

```
<add key="CacheUpdateCroneTime" value="0 0 1 1/1 * ? *" />
```

Time value is a **CRON Expression**. This CRON Expression runs at 1:00 AM everyday

User Defined Events

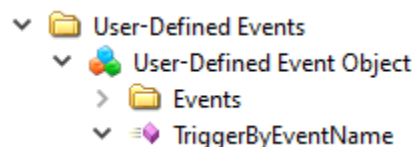
All user defined events are Listed under User-Defined Events -> User-Defined Event Object -> Events



Triggering Events

*By calling **TriggerByEventName** method*

“User-Defined Event Object” which contains the method to be called, namely “**TriggerByEventName**”.



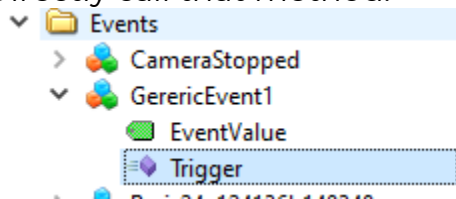
The method “**TriggerByEventName**” accepts a parameter named “*EventName*” of the type “String”.

Input Arguments			
Name	Value		Data Type Description
EventName	<input type="text"/>	... Load file...	String

Result

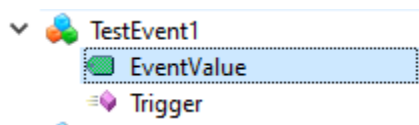
By Trigger method

Under each event, there is a method “**Trigger**” to call that event. We can Directly call that method.



By changing value

Under each User Defined Event, there is a variable “**EventValue**”.



This can be viewed in the data access view, and by changing its value to “1” we can trigger that event.

Data Access View Event View					
#	Server	Node Id	Display Name	Value	Datatype
1	CTS OInsightsO...	NS2[String]TestEvent1.EventValue	EventValue	0	Double

Data Manager Service

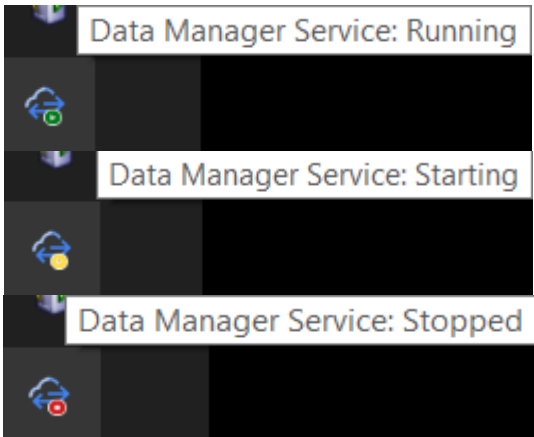
Data Manager Service is responsible for the efficient storage and retrieval of historical data. This service manages the long-term storage of data points with IoT sensor values, ensuring that historical information is readily accessible for analysis and reporting.

Configuring Data Manager Service

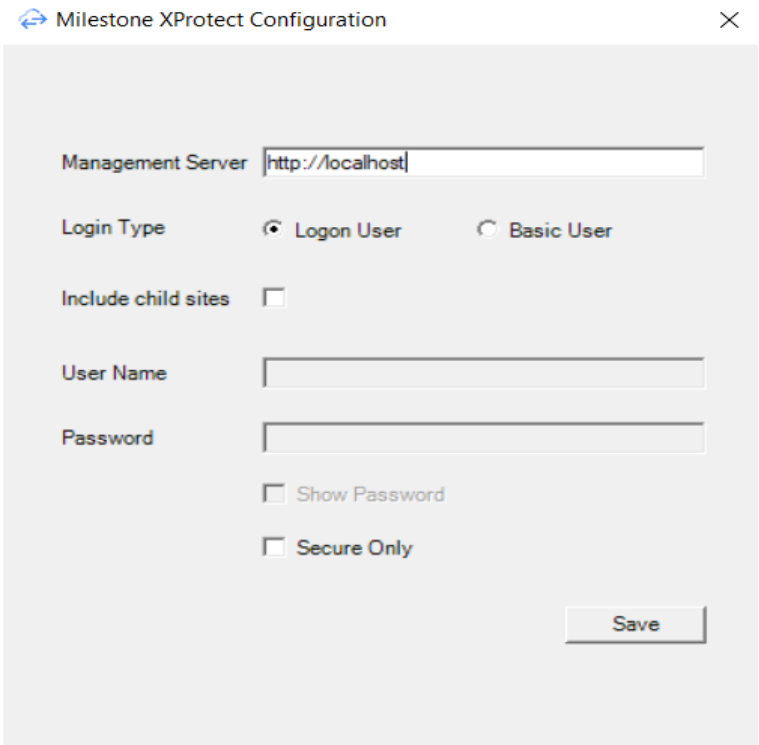
To ensure optimal performance and seamless integration with the XProtect, the O-Insights Data Manager Service requires proper configuration. Users can configure this service through the Configurator Tool's user-friendly interface, which offers straightforward options for setting up basic parameters. For more advanced customization, users can also directly modify the configuration file, allowing for fine-tuning of the service to meet specific operational needs. The following sections detail the configuration settings available both through the UI and within the advanced settings of the config file.

Configurator Tool

The IoT Configurator tool can be used to configure the Data Manager. The available options are explained in the table below.

Name	Description
Status: Indicator	<div></div> <p>Status of the Data Manager Service</p>
Restart	Restarts the Data Manager Service
Start	Starts the Data Manager Service
Stop	Stops the Data Manager Service
Logs	Shows event logs for the data manager service. Default Path<Installation Directory>\O-Insights IoT

	<i>Driver\ O-Insights Data Manager\Logs</i>
Config	<div><div><div>Log Level</div><div>ERROR</div><div></div></div><div><input type="checkbox"/> Enable Encryption</div><div>Thumbprint</div><div></div><div>Save</div></div> <div><div>Log Level</div><div>ERROR</div><div>ALL DEBUG INFO WARN ERROR FATAL OFF</div></div> <div>Thumbprint</div> <div></div> <div>Save</div>

Milestone XProtect Configuration	 <p>Milestone XProtect configuration for the O-Insights IoT service to communicate with Milestone XProtect. Add all the needed information and click Save.</p>
Version	It shows the version of the currently installed Data Manager service.

Note: If you cannot access the logs from the IoT configurator tool, do exit the tool and rerun as administrator

Config File (Advanced)

Location to the config file: <Installation Directory>\O-Insights IoT Driver\ O-Insights Data Manager\ DataManager.dll.config

Key	Description	Default Value
HostingPort	Port used by O-Insights Data Manager service	8095
EnableSSL	To enable encryption for Data Manager service. To Enable Encryption	0

	set to 1 else 0	
SSLCertificateThumbprint	Thumbprint of SSL certificate	
MongoUserName	Username for MongoDB connection	
MongoPassword	Encrypted password for MongoDB Connection	
MongoPort	Mongo DB port number	27017
MongoIpAddress	MongoDB address	localhost

IoT Config Tool

Setting up the Config Tool

This step-by-step guide will guide users in setting up the IoT Config Tool, including Point Configuration and Alarm Configuration. It is recommended to install the IoT Config Tool on Smart Clients where the IoT data is to be configured and managed.

Note: Any changes made in the IoT Config Tool will only take effect after the configuration is reloaded.

Starting the Config Tool

- **Launch Milestone XProtect Smart Client** and log in using your Milestone credentials.
- Navigate to the **IoT Config Tool** tab in the Smart Client.
- In the sidebar menu, navigate to the **IoT Config Tool** page and click on **Point Configuration**.
- You will now be in the **Point Configuration** interface.

Creating and Organizing Folders

- You will see a default folder named “config.” All new folders will be created under this config folder in a hierarchical tree structure, allowing users to organize points efficiently:
 - Click on the **+** icon to create new folders.
 - In the text field that appears, enter the name of the new folder.
Example: *Building, Floor, Area*

Points and Histories

Understanding Points and Histories

A point represents specific live/real-time data elements associated with cameras, sensors, or device events. It is a measurable value or status used for

monitoring, analysis, and automation. History is the recorded value of the points.

Examples of Points:

- **Video Analytics Data:** Object count, motion detection, intruder detection
- **Alarm Status:** Active/Inactive
- **Environmental Data:** Temperature, humidity, light level

Note: Ensure that the IoT Config Tool is installed on your XProtect Smart Client before performing point configuration.

Creating/Configuring Points for MQTT/Webhook

Add Point

Name *
Person_Out

Driver Type
MQTT

Point Type
NUMERIC

Topic Name *
Hanwa/Cam1

Value Key
Count

Precision
0

Unit

History Configuration

☐ Enable Sampling ☐ COV Enabled ☒ Enable History

Alarm Configuration

☐ Enable Alarm

Low Limit
0

High Limit
0

Event Name

Rules

Connector
AND +

Key	Operator	Value	Actions
ObjectType	=	Person	
Direction	=	OUT	

CANCEL ADD

- Navigate to the desired folder within the main config folder.
- Click the + icon to add a new point within that folder.

- You will now be in the **Add Point** interface:
 - Complete the required fields in the Add Point interface to define the new point. This typically includes:
 - **Name:** A descriptive label for the point.
 - Example: *People_In, Room Temperature*
 - **Driver:** There are two driver types supported by O-Insights:
 - **MQTT:** The point will be included in the payload sent using MQTT to the IoT Drivers.
 - **Webhook:** The point will be part of the payload sent by the Webhook, which will then be delivered to the IoT drivers.
 - **Point Type:** The category or type of data the point represents. O-Insights supports the following point types:
 - **Numeric Point Type:** Represents a quantitative measurement or quantity, such as temperature or person count.
 - **Boolean Point Type:** Represents a logical state with two possible values (true/false), often used to indicate conditions like intrusion or door status.
 - **String Point Type:** Represents textual data within a point, used for qualitative data such as Operating Mode or Fault Status.
 - The available configuration options will vary based on the selected point type:
 - **Numeric Point Type:**
 - **Precision:** Specifies the number of decimal places for the point.
 - **Unit:** Defines the measurement units (e.g., degrees Celsius) for clarity and consistency.

Creating/Configuring History for MQTT/Webhook

- **Point History:**
 - To access and analyse historical data associated with a specific point, utilize the point history feature. Tick the checkbox to enable point history.
 - Disabling this feature will only retain the current point value as the live value, and historical data will not be recorded.

- **Enable Sampling:**
 - Sampling is a method for collecting and recording data at predetermined time intervals.
 - For point values that generate continuous data streams, enabling sampling will omit data for the duration specified, optimizing system resources while maintaining essential information.
 - To enable sampling:
 - Click on the checkbox and enter the minimum sampling time in minutes.
- **COV (Change of Value) Enabled:**
 - Establish a COV to record data only when the new value differs from the previous reading.
 - Click on the checkbox to enable COV.
 - Sampling and Change of Value (COV) functionalities cannot be activated at the same time.

Topic Name

- A topic name is a string used to classify messages published to the broker. Every MQTT message is associated with a topic name, which subscribers use to filter and receive relevant messages.
 - Refer to OEM documentation for specific topic names related to sensors or cameras.
 - **Example:**
 - *Building/Floor1/Room1/temperature*
 - *VMS/Camera1/App/VMD_alarm*

ValueKey

- If the payload contains multiple data points within a JSON structure, you can selectively extract the desired value based on its corresponding key.
- **Example:** Consider the following JSON payload:


```
{
```

```
"temperature": 24,  
"humidity": 40,  
"AQI": 46  
}
```

- If you need to set a point for 'temperature,' enter 'temperature' into the ValueKey text field.
- The point value will now be associated with the key 'temperature'.

Point Alarm Configuration

- **Point Alarm configuration** allows you to trigger a user-defined event in XProtect when a point value is considered an alarm.
- **To enable alarm functionality:**
 - Check the **Enable Alarm** checkbox.
 - Alarm conditions vary based on the point data type:
 - **Numeric:**
 - Specify the High Limit and Low Limit values. An event in XProtect will be triggered whenever the numerical payload value exceeds these thresholds.
 - **String:**
 - The alarm is triggered when the incoming value exactly (or partially) matches a predefined value. Enter the desired string in the string value text field.
 - **Boolean:**
 - The alarm is triggered based on the state of a Boolean point, which can have only two values: true or false.
 - Additionally, define text messages to display when each alarm state is active in the **True Text** and **False Text** fields.
 - Alarms for the user-defined events will need to be configured in XProtect.
- **Event Configuration:**
 - Enter the corresponding event name. This action will initiate a user-defined event trigger within the XProtect. It is required to restart the O-Insights IoT service after adding new events in the Milestone which will be configured here.

Rules

Rules are conditional statements applied to incoming payload data. These criteria determine whether the data is accepted and processed or discarded based on specific parameters which are defined below.

- **Connector**

- AND Operator: All defined rules must be satisfied for the data to be recorded. If any rule fails, the data is discarded.
- OR Operator: At least one of the defined rules must be satisfied for the data to be recorded. If none of the rules match, the data is discarded.

- **Example:**

- Consider a camera that counts people and sends the following payload. The count is specific to an Object Type and Direction. To create a “People Count OUT” point, the rule to extract the value from the payload would be as follows:

```
{  
  "UtcTime": "2024-03-04T10:46:33.449Z",  
  "Source": {  
    "VideoSourceToken": "VideoSourceToken-0",  
    "CountingRuleIndex": "0",  
    "RuleName": "LineCount2"  
  },  
  "Data": {  
    "ReportType": "Punctual",  
    "ObjectType": "Person",  
    "Direction": "OUT",  
    "Count": "3"  
  }  
}
```

- Click on the **+** icon to open the add expression text fields.
- **Key:** Enter the value key from the payload.
- **Operator:**
 - Greater Than (>)
 - Less Than (<)
 - Greater Than or Equal To (>=)
 - Less Than or Equal To (<=)

- Equal To (=)
- Contains (string matching)
- These operators enable the creation of complex rules to accurately process incoming data based on specific criteria.
- **Value:** Represents the specific data point used for comparison against the incoming payload data.
- Finalize rule creation by clicking the **Add** button.

Key	Operator	Value	Actions
ObjectType	=	Person	
Direction	=	OUT	

CANCEL UPDATE

Creating/Configuring Points for OPC UA

Adding OPC Points

Add Point

Name *

OPC Server

Unit

Driver Type

- MQTT
- WEBHOOK
- OPC

Point Type

Precision

History Configuration

☐ Enable Sampling ☐ COV Enabled ☐ Enable History

Alarm Configuration

ADD CONDITION

CANCEL ADD

To add an OPC point, use the *Add Point* window and select *OPC* as the driver type. Here, users can set up point configurations:

- **Point Type:** Select the type of data point (e.g., Numeric, Boolean).

- **Node ID:** Choose or enter the Node ID for the OPC point. For example, if connecting to SCADA, select a node from the server's list of exposed nodes. NodeIDs will be filtered based on their PointType.

Creating/Configuring History for OPC UA

- **Point History:**
 - To access and analyse historical data associated with a specific point, utilize the point history feature. Tick the checkbox to enable point history.
 - Disabling this feature will only retain the current point value as the live value, and historical data will not be recorded.
- **Enable Sampling:**
 - Sampling is a method for collecting and recording data at predetermined time intervals.
 - For point values that generate continuous data streams, enabling sampling will omit data for the duration specified, optimizing system resources while maintaining essential information.
 - To enable sampling:

- Click on the checkbox and enter the minimum sampling time in minutes.
- **COV (Change of Value) Enabled:**
 - Establish a COV to record data only when the new value differs from the previous reading.
 - Click on the checkbox to enable COV.
 - Sampling and Change of Value (COV) functionalities cannot be activated at the same time.

Point Alarm Configuration for OPC UA


- **Point Alarm configuration** allows you to trigger a user-defined event in XProtect when a point value is considered an alarm. Restarting O-Insights IoT service is required after adding new events in the Milestone which will be configured here.
- **To enable alarm functionality:**
 - Check the **Enable Alarm** checkbox.
 - Alarm conditions vary based on the point data type:
 - **Numeric:**
 - If multiple conditions are added, an alarm will trigger when the value stays within the set limits. Specify the High Limit and Low Limit values. An event in XProtect will be triggered whenever the numerical payload value exceeds these thresholds.
 - **String:**
 - The alarm is triggered when the incoming value exactly (or partially) matches a predefined value. Enter the desired string in the string value text field.
 - **Boolean:**
 - The alarm is triggered based on the state of a Boolean point, which can have only two values: true or false.
 - Additionally, define text messages to display when each alarm state is active in the **True Text** and **False Text** fields.
 - Alarms for the user-defined events will need to be configured in XProtect.
- **Event Configuration:**

- Enter the corresponding event name. This action will initiate a user-defined event trigger within the XProtect. If you are configuring a newly added event, O-Insights IoT service must be restarted. Otherwise alarms for this event will not be generated.

Updating Points

- To modify a point, click the designated **edit** icon within the point interface.
- Adjust the point attributes as necessary.
- Confirm the changes by clicking the **Update** button.
- Click the **Cancel** button to close the window without saving changes.

View Point History

- Click on the history icon to access the point history.
- **Timestamps:** Timestamps provide the precise time when the point's value was recorded.
- **Status:** The status indicates whether an alarm condition is currently active. A value of "true" signifies an active alarm, while a value of "false" denotes a normal state.
- **Value:** Displays the recorded point value.
- To delete the history for that point, click the  icon. **This action is irreversible.**



Delete Point History

- Permanently remove a point from the system by clicking on the "" icon.



Refresh Point History

- Click on the " " icon on the top right to refresh all points. It is recommended to refresh the points after configuring a new point.

Copy and Paste Points:

- To duplicate a point configuration:
 - Select the point you want to copy and click the **copy**  icon.
 - Paste the copied configuration at the desired location using the **paste**  icon.

Import and Export Points:

- To efficiently manage point configurations, users can export and import point data.
 - To export a point configuration, click the **export**  icon.
 - To import point configurations, click the **import**  icon.
 - Select the desired .csv file, and the system will load the saved points.
 - Importing will skip the point if the container is not present.

Note: Exporting generates a .csv file containing the selected points within a specified folder. Exporting is based on Folders. Importing configurations involves selecting a .csv file, ensuring the target container exists, and loading the data. Importing would fail if the container specified in the import file is not present.

Alarm Configuration

Update Alarm

Name *
CROSS LINE ALARM BICYCLE

Driver Type
MQTT

Topic Name *
i-PRO/Cameral/App/AIVMD_alarm

Value Key *
AlarmMessage

Event Configuration

Event Name *
CROSS LINE ALARM BICYCLE

Rules

Payload Key	Operator	Value	Actions
AlarmMessage	Contains	CROSS LINE ALARM BICYCLE	

CANCEL UPDATE

To trigger user-defined events in XProtect upon receiving an alarm payload, follow these steps. It's important to first configure the corresponding alarm within the XProtect Management Client to ensure proper functionality.

Alarm points do not generate historical data. Their primary function is to indicate current or active alerts by triggering user defined events in XProtect.

Alarm Configuration

- Access the alarm configuration settings by navigating to the “Alarm Configuration” tab in the sidebar menu of the IoT Config Tool.
- Click the **+** icon to add a new alarm.
- In the “Add Alarm” interface:
 - **Name:** Enter a distinct label for the alarm.
 - **Driver Type:** Choose the appropriate driver type:
 - **MQTT:** The alarm will be sent using MQTT to the IoT Drivers.
 - **WebHook:** The sensor value will be sent via Webhook to the IoT Drivers (e.g., for LoRaWAN Network Server).

- **Value Key:** If the alarm payload contains multiple data points within a JSON structure, specify the key corresponding to the desired value in the provided text field.

Event Configuration

- **Event Name:** Specify the exact XProtect event name in the text field.

Rule Configuration

Conditional statements applied to incoming payload data determine whether the data triggers an alarm or should be disregarded.

- **AND/OR Operators:**
 - **AND:** All conditions must be met for the rule to be satisfied.
 - **OR:** At least one condition must be met for the rule to be satisfied.
- **Rule Creation:** Follow the instructions outlined in section 6.9.2 to fill in the text fields for creating rules

Example


Payload Key	Operator	Value	Actions
AlarmMessage	Contains	CROSS LINE ALARM BICYCLE	

```
{
  "CameraIPaddress": "192168000044",
  "CameraMACaddress": "d42dc51674f6",
  "Time": "20240813094650",
```



```
"TimeZone": "10530",  
"SummerTime": "0",  
"AlarmMessage": "INTRUDER ALARM VEHICLE 0101"  
}
```

This payload includes various data points such as the camera's IP address, MAC address, timestamp, time zone, and an alarm message indicating an intruder alarm related to a vehicle. You can use these data points to configure rules or alarms within the system based on the specific values provided.

Update Alarm

- To modify an existing alarm trigger, click the designated edit  icon within the alarm interface.
- Make the necessary adjustments to the alarm attributes, then confirm your changes by clicking the Update button.
- If you wish to discard the changes, click the cancel button to close the window without saving.

Copy and Paste Alarms

- To duplicate an alarm configuration, select the desired alarm config and click the " " icon.
- Paste the copied configuration to the desired location using the " " icon.

Configuring the IoT Config Tool

- To access the settings panel for the IoT Config tool, click on the settings icon located at the bottom left corner of the panel.
- The settings panel comprises of two sections: “Configurations” and “DB Stats”.

Configurations

Reload Configuration

- Reload configurations to apply any changes made to points and alarms. This step is crucial after modifying points to ensure the updates take effect.

Retention Settings

- Retention settings determine how long historical data is stored. Users can specify the desired data retention period based on storage constraints and analysis needs. **By default, the IoT Driver stores historical data for three months.**

DB Stats

DB Stats Overview

- This section provides insights into database storage usage for historical point values.

Point Search

- This functionality allows users to search for a number of data records associated with a specific point.

Audit Trail

The Audit Trail provides a comprehensive record of all actions performed in the O-Insights IoT Config Tool. To access the Audit Trail, select it from the sidebar.

Date Range Selection

- Click the date range picker dropdown at the top corner of the table area to expand and select the desired date for viewing audit trail records.

Audit Trail Columns

- **Timestamp:** Lists the timestamp of the audit trail event.
- **Operation:** Indicates the operation performed.
- **Target:** Shows where the operation was performed.
- **Old Value:** Displays the previous value before the change.
- **New Value:** Shows the updated value after the change.
- **User:** Identifies the user who performed the action.

Use the **Operations** dropdown to filter the table by specific operations.

O-Insights Publish Topic Plugin

The O-Insights MQTT Publish Topics Plugin is an essential tool within the XProtect Management Client that allows you to define and manage the topics used for publishing XProtect events over MQTT. This plugin provides flexibility in sending camera status updates, alarms, and custom events to the MQTT broker.

Before proceeding, ensure that the MQTT driver address is properly configured by navigating to the following path:

C:\Program Files\Milestone\MIPPlugins\O-Insights MQTT Publish Topics\O-InsightsMqtt.dll.config

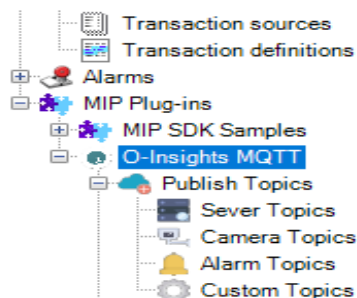
Publish Topics

- Open the XProtect Management Client and navigate to the O-Insights MQTT Publish Topics plugin.
- The plugin allows all XProtect events to be sent over MQTT.
- **Camera Online/Offline Status:** To send camera online/offline status, enable this setting in the MQTT driver configuration.

To enable set `EnableCameraStatusPublish` value to "true" /> in the following file.

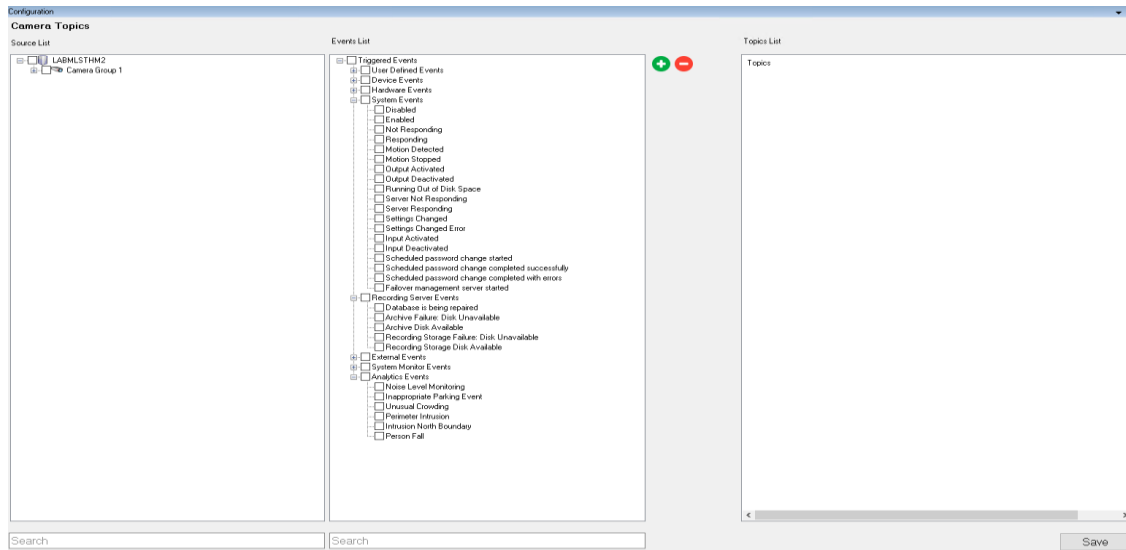
<Installation Directory>\O-Insights IoT Driver\O-Insights MQTT\MQTTSvc.exe.config

Can set frequency of sending camera status by setting value to *CameraStatusPublishIntervallInMins*.

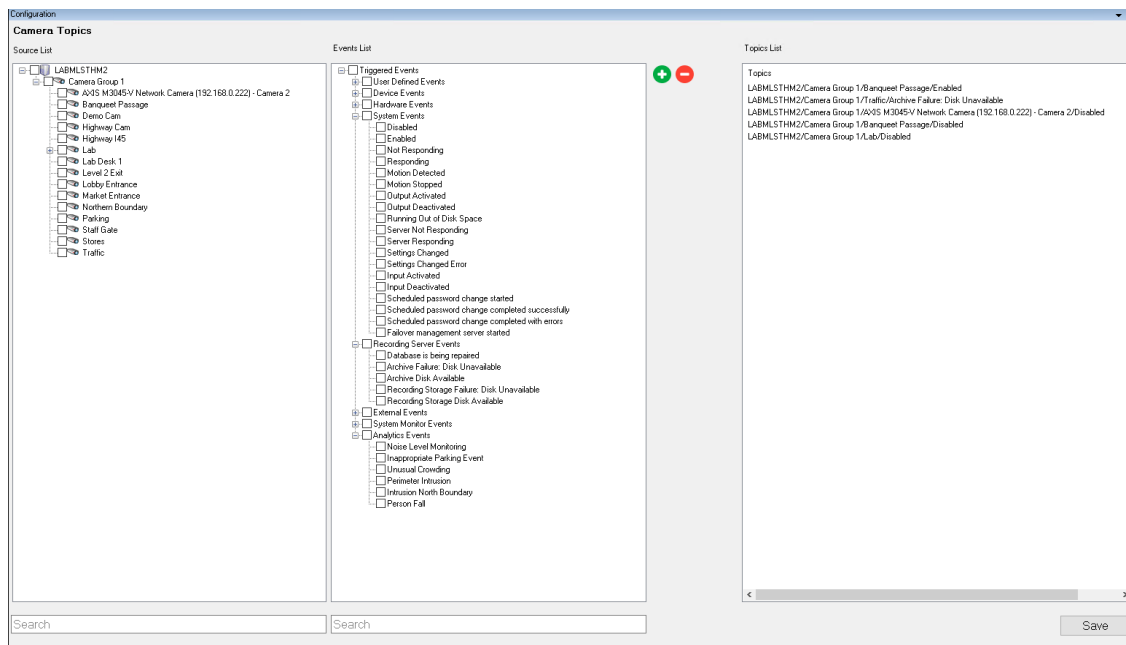


Creating Publish Topics

- Navigate to the **Publish Topics** section within the plugin.
- Select the desired items from the source list and the corresponding events from the events list.



- Add the selected items and save your configuration.



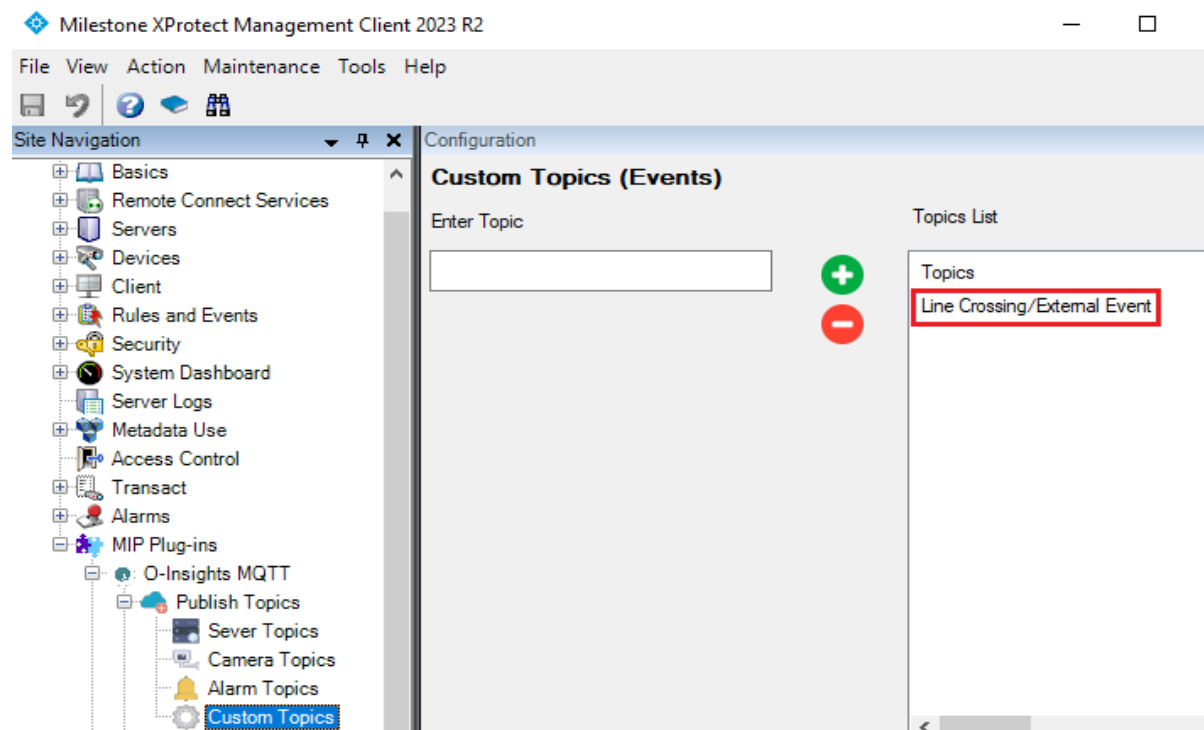
- Events and alarms received from XProtect will be forwarded to the MQTT broker according to the configured publish topics.

Creating Custom Publish Topics

The plugin also supports the creation of custom publish topics for user-defined events.

- Start by creating a user-defined event in XProtect. For example, you could create a custom event like “Line Crossing.”
- After creating the event, set up an associated alarm and verify its configuration in the Alarm Manager.
- Navigate to **MIP Plugin/O-Insights MQTT/Publish Topics/Custom Topics**.
- Add your custom event as an “eventname/External Event” in this section.
- Ensure that the MQTT server details are updated in the plugin configuration file.

By following these steps, you can effectively manage and publish both standard and custom XProtect events over MQTT, providing seamless integration with your IoT ecosystem.



Periodic Camera Status with Publish Topics

The driver can be configured to periodically send Camera Online/Offline status updates. The messages that will be sent include:

- **Camera Facts**

This message provides an overview of the total number of cameras, how many are online, and how many are offline.

- **Sample Payload:**

```
{  
  "TotalCameras": 5000,  
  "CamerasOnlineCount": 4900,  
  "CamerasOfflineCount": 100,  
  "Time": "19-08-2024 16:18:38"  
}
```

- **Camera Details**

This message gives specific details about individual cameras, including their name, model, status, and the server they are connected to.

- **Sample Payload:**

```
{  
  "Name": "Lobby Gate",  
  "Model": "FLEXIDOME indoor",  
  "Status": true,  
  "RecordingServer": "labmlsthm2.ctsdom.com",  
  "SiteName": "LABMLSTHM2"  
}
```

Leveraging the IoT Data

The data managed by the O-Insights IoT Tool can be leveraged across various O-Insights features, including Widgets for dynamic visualizations, the Agile Dashboard for interactive monitoring, the O-Insights Query Engine for advanced reporting, web-based dashboards, and the Maps Plugin for geographic data display on Milestone Maps.

To access the features of the new IoT Driver, ensure you configure the O-Insights Data Manager in the settings first by entering the address of the Data Manager Service.

O-Insights for VMS

The data from the IoT Plugin can be used across various O-Insights for VMS features to enhance monitoring and analysis capabilities.

The following widgets from O-Insights for VMS are currently available to visualize the data. Detailed information on configuration is available in the VMS Help file.

IoT Data Widget

The IoT Data Widget provides real-time visibility into live values from IoT data points, sourced from O-Insights IoT Drivers. It effectively visualizes data by categorizing values into predefined ranges. For instance, displaying air quality index (AQI) levels as "Good," "Moderate," "Unhealthy," "Very Unhealthy," and "Hazardous."

IoT Data List Widget

The IoT Data List Widget displays live values from multiple IoT data sources. For example, it can be configured to show the number of smart bins that are filled beyond 75%, providing a clear overview of bin status across different locations.

IoT Custom Data Widget

The IoT Custom Data Widget displays a single value or point data derived from calculations on historical data provided by the O-Insights IoT Drivers. For instance, it can show the total count of vehicles in a parking area for a specific day.

IoT Donut Chart

The IoT Donut Chart offers a dynamic, circular visual representation of data, where segments illustrate proportions of the total. It updates in real-time with data changes. For example, it can display the distribution of parking availability, used parking spaces, and total parking slots.

IoT Line/Bar Chart

The IoT Donut Chart enhances analytics by plotting historical data over a common date span. For example, it can visualize temperature and humidity levels for different areas alongside the operational status of an AC unit at hourly intervals throughout the day.

IoT Heatmap Chart

The IoT Heatmap Chart visualizes historical data using a color-coded heatmap to represent various data values or ranges. For example, it can highlight temperature hotspots in different areas, show occupancy distribution across spaces, or display CO2 concentration levels over the past 7 days.

IoT Comparison Chart Widget

The IoT Comparison Chart widget provides a side-by-side comparison of data points for two different periods. For instance, it can compare occupancy data

from this month with that from the last month, or vehicle counts in a parking area this week against the previous week, all on a single chart.

IoT Gauge Chart

The IoT Gauge Chart widget displays a single data point within a specified range, resembling a speedometer for quick visual interpretation. For instance, it is ideal for tracking progress toward a goal, monitoring performance metrics, or observing critical values like system usage.

O-Insights Reporting

The O-Insights Reporting enables advanced data analysis and report generation, providing deeper insights into historical and real-time information.

The following report from O-Insights Reporting is currently available. Detailed information on configuration is available in the Reporting Help file.

IoT Analytics Report

The IoT Analytics report type creates detailed reports by aggregating point values based on selected criteria, providing valuable insights. For instance, a temperature sensor might report temperature values, while a camera could provide people count data.

O-Insights Web

The O-Insights Web interface offers visualization of IoT data in web-based dashboards for convenient access and management.

The following widgets from O-Insights Web are currently available to visualize the data. Detailed information on configuration is available in the O-Insights Web Help file.

IoT Single Value Widget

The IoT Single Value Widget displays real-time values from IoT data points provided by the O-Insights IoT Driver, offering immediate visibility into the data being monitored.

IoT Data List Widget

The IoT Data List widget displays real-time values from multiple IoT data sources, providing a comprehensive view of various data points at once.

O-Insights Maps Plugin

The O-Insights map add-on enhances situational awareness by overlaying real-time IoT sensor data onto XProtect maps. By integrating icons onto Milestone XProtect Maps and Smart Maps, users can visualize sensor and camera locations along with their generated points and alarms, providing a comprehensive overview of the monitored environment.

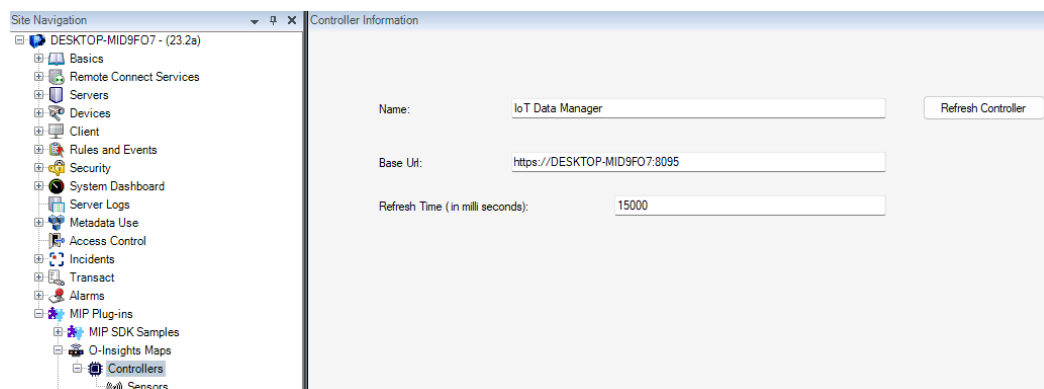
Configuration of the Maps Plugin

Access the O-Insights Maps Configuration

- Open the Milestone Management Client.
- Navigate to the “O-Insights Maps” section in the sidebar menu.
- Expand the menu to reveal the “Controllers” option.

Set Up the IoT Data Manager

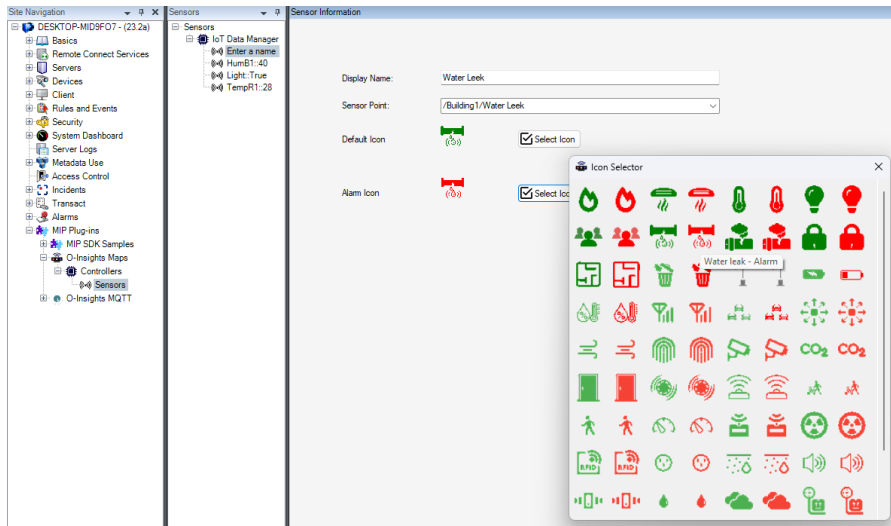
- The IoT Data Manager is automatically assigned as the default controller.
- Enter the base URL for the Data Manager in the Base URL field.
- Specify the data refresh interval in milliseconds in the Refresh Time field.
- After configuring the controller settings, click on the “Refresh Controller” button.



Add Sensors

- Navigate to the “Sensors” menu in the sidebar.
- Right-click and select “Add New” to add sensors to a new controller.

- Provide a Display Name for each sensor.
- Select the Sensor Point from the data manager configurations using the dropdown menu.
- Specify the Default Icon and Alarm Icon for the sensor. A diverse collection of pre-designed icons is available. Alarm icons are typically red, while Default icons are green.
- **After adding new sensors, restart the event server to apply the changes.**



Smart Client Views for Adding Sensor Values to Maps

Open Map View:

- In Smart Client, open the Map view layout and click on the “Puzzle” icon in the toolbar.
- Select “O-Insights Maps” from the dropdown menu.
- Choose the specified controller associated with your map.

Place Icons on the Map:

- Drag and drop the relevant icons onto the map view.
- Icons can be placed anywhere on the map and can be resized and rotated as needed.
- Each icon displays the live value of the associated point and changes appearance based on its alarm status, reflecting real-time alerts when an alarm is triggered.



FAQs

- **Why are the events in the XProtect server not being triggered for point alarms?**

If the user-defined event was created after the O-Insights IoT Drivers started, you need to restart the IoT driver service for the event to be recognized and triggered.

- **Where are the log files for the IoT Driver Configurator located?**

The logs for the IoT Configuration tool, is located at: <Installation Directory>\O-Insights IoT Driver\Logs\OInsightsTrayTool.txt
To get the log, IoT Driver configurator should in admin mode.

- **Why are Events/Alarms from Milestone XProtect not published on MQTT when configured?**

Verify that the MQTT driver address is properly configured in config file located at : <Installation Directory>\ O-Insights IoT Driver\O-Insights Service\Plugins\ MQTTPlugin.dll.config
<add key="MQTTServer" value="http://127.0.0.1:8094"/>

- **Why is the Data from Milestone XProtect not displayed in OPC UA clients, when server mode is enabled?**

Verify that the OPC UA driver address is properly configured in config file located at: <Installation Directory>\ O-Insights IoT Driver\ O-Insights Service\Plugins\ OPCPlugin.dll.config<add key="OPCServer" value="http://127.0.0.1:8091" />

- **Why are the events not triggered in Milestone XProtect even if I configured points in O-Insights OPC UA driver as a client?**

Verify that the trigger event settings configured in config file located at: <Installation Directory>\ O-Insights IoT Driver\O-InsightsService\OInsights.exe.config<add key="TriggerUserDefinedEventsBasedOn" value="Event" />

- **Why are OPC Servers added with security policies not connecting in the OPC UA driver?**

Check whether the certificates from the servers are trusted.

- **How to trust a certificate from an OPC UA server?**

To trust a certificate from an OPC UA server, move the rejected certificate from:
<Installation Directory>\O-Insights IoT Driver\O-Insights OPC\pki\rejected\certs

To the trusted folder:
<Installation Directory>\O-Insights IoT Driver\O-Insights
OPC\pki\trusted\certs