

The Kaspersky logo is displayed in a bold, black, lowercase sans-serif font. It is positioned in the upper left quadrant of the slide, which features a white background with a large, rounded, white shape that resembles a stylized 'K' or a protective shield. This white shape is set against a background of teal and green gradients.

Kaspersky Machine Learning for Anomaly Detection

© 2024 AO Kaspersky Lab

Contents

[About Kaspersky Machine Learning for Anomaly Detection](#)

[Distribution kit](#)

[Hardware and software requirements](#)

[Security recommendations](#)

[Fixing vulnerabilities and installing critical updates](#)

[Managing access to application functions](#)

[What's new](#)

[Kaspersky MLAD architecture](#)

[Common deployment scenarios](#)

[Telemetry and event data flow diagram](#)

[Ports used by Kaspersky MLAD](#)

[Installing and removing the application](#)

[Installing the application](#)

[Updating the application](#)

[Checking the integrity of Kaspersky MLAD archive files](#)

[Backing up the application](#)

[Rolling back the application to the previous installed version](#)

[Scenario for restoring Kaspersky MLAD from a backup](#)

[Getting started](#)

[Starting and stopping Kaspersky MLAD](#)

[Switching between Kaspersky MLAD state control modes](#)

[Updating Kaspersky MLAD certificates](#)

[First startup of Kaspersky MLAD](#)

[Removing the application](#)

[Kaspersky MLAD web interface](#)

[Connecting to Kaspersky MLAD and terminating a user session](#)

[Connecting to the web interface](#)

[Terminating a Kaspersky MLAD connection session](#)

[Changing a user account password](#)

[Selecting the localization language for the Kaspersky MLAD web interface](#)

[Licensing the application](#)

[About the End User License Agreement](#)

[About the license](#)

[About the license certificate](#)

[About the license key](#)

[About the license key file](#)

[Available functionality of Kaspersky MLAD depending on the specific license](#)

[Adding a license key](#)

[Viewing information about an added license key](#)

[Removing a license key](#)

[Processing and storing data in Kaspersky MLAD](#)

[About data provision](#)

[Directories for storing application data](#)

[System administrator tasks](#)

[Managing user accounts](#)

[Creating a user account](#)

[Editing a user account](#)

[Revoking authentication tokens for a user account](#)

[Viewing access rights for a user account](#)

[Manage roles](#)

[Creating role](#)

[Editing role](#)

[Deleting role](#)

[Viewing access rights for a role](#)

[Managing incident notifications](#)

[Creating an incident notification](#)

[Editing an incident notification](#)

[Deleting an incident notification](#)

[Configuring Kaspersky MLAD](#)

[Configuring the main settings of Kaspersky MLAD](#)

[Configuring the security settings of Kaspersky MLAD](#)

[Configuring the Anomaly Detector service](#)

[Configuring the Keeper service](#)

[Configuring the Mail Notifier service](#)

[Configuring the Similar Anomaly service](#)

[Configuring the Stream Processor service](#)

[Configuring the HTTP Connector](#)

[Configuring the MQTT Connector](#)

[Configuring the AMQP Connector](#)

[Configuring the OPC UA Connector](#)

[Configuring the KICS Connector](#)

[Configuring the CEF Connector](#)

[Configuring the WebSocket Connector](#)

[Configuring the Event Processor service](#)

[Configuring the statuses and causes of incidents](#)

[Configuring logging for Kaspersky MLAD services](#)

[Configuring time intervals for displaying data](#)

[Configuring how the Kaspersky MLAD menu items are displayed](#)

[Export and import of Kaspersky MLAD settings](#)

[Managing assets and tags](#)

[About monitored asset hierarchical structure](#)

[About tags](#)

[Create asset](#)

[Change asset settings](#)

[Create tag](#)

[Adding a tag to an asset](#)

[Editing a tag](#)

[Moving assets and tags](#)

[Deleting an asset or tag](#)

[Checking the current structure of tags](#)

[Uploading tag and asset configuration to the system](#)

[Saving tag and asset configuration to a file](#)

[Working with the main menu](#)

[Scenario: working with Kaspersky MLAD](#)

[Viewing summary data in the Dashboard section](#)

[Viewing incoming data in the Monitoring section](#)

[Viewing data for a specific preset in the Monitoring section](#)

[Selecting elements of the ML models in the Monitoring section](#)

[Selecting a time interval in the Monitoring section](#)

[Configuring how graphs are displayed in the Monitoring section](#)

[Viewing data in the History section](#)

[Viewing historical data for a specific preset](#)

[Selecting elements of the ML model in the History section](#)

[Selecting a date and time interval in the History section](#)

[Navigating through time in the History section](#)

[Configuring how graphs are displayed in the History section](#)

[Viewing data in the Time slice section](#)

[Viewing data for a specific preset in the Time slice section](#)

[Selecting a specific element of the ML model in the Time slice section](#)

[Selecting a date and time interval in the Time slice section](#)

[Navigating through time in the Time slice section](#)

[Configuring how graphs are displayed in the Time slice section](#)

[Working with events and patterns](#)

[About Event Processor](#)

[About events](#)

[About patterns](#)

[About attention](#)

[About Event Processor operating modes](#)

[About monitors](#)

[Configure display of event parameters](#)

[Configure attention settings](#)

[Adding an attention head](#)

[Editing an attention head](#)

[Removing an attention head](#)

[Managing monitors](#)

[Creating a monitor](#)

[Editing a monitor](#)

[Deleting a monitor](#)

[Viewing the events history](#)

[Viewing the pattern history](#)

[Working with incidents and groups of incidents](#)

[About incidents](#)

[About incidents detected by a predictive element of an ML model](#)

[About incidents detected by an ML model element based on a diagnostic rule](#)

[About incidents detected by an ML model element based on an elliptic envelope](#)

[About incidents detected by the Limit Detector](#)

[About incidents detected by the Stream Processor service](#)

[About anomalies](#)

[Scenario: analysis of incidents](#)

[Viewing incidents](#)

[Viewing the technical specifications of a registered incident](#)

[Viewing incident groups](#)

[Studying the behavior of the monitored asset at the moment when an incident was detected](#)

[Adding a status, cause, expert opinion or note to an incident or incident group](#)

[Exporting incidents to a file](#)

[Managing ML models](#)

[About ML models](#)

[About predictive ML model elements](#)

[About elements of an ML model based on a diagnostic rule](#)

[About elliptic envelope-based ML model elements](#)

[About statuses and states of ML models and their elements](#)

[About ML model templates](#)

[About markups](#)

[About conditions included in markups and diagnostic rules](#)

[Scenario: working with ML models](#)

[Search and filter objects in the Models section](#)

[Working with markups](#)

[Creating markup](#)

[Viewing the markup chart](#)

[Copying a markup](#)

[Modifying the markup](#)

[Removing markup](#)

[Working with imported ML models](#)

[ML model importing](#)

[Activating an imported ML model](#)

[Changing the parameters of an element of an imported ML model](#)

[Working with manually created ML models](#)

[Creating an ML model](#)

[Adding a predictive element to an ML model](#)

[Modifying an ML model predictive element](#)

[Adding an ML model element based on a diagnostic rule](#)

[Changing an ML model element based on a diagnostic rule](#)

[Adding an elliptic envelope-based ML model element](#)

[Editing an elliptic envelope-based ML model element](#)

[Cloning of the ML model element](#)

[Removing an ML model element](#)

[Cloning an ML model](#)

[Working with ML model templates](#)

[Creating a template based on an ML model](#)

[Editing an ML model template](#)

[Creating an ML model based on a template](#)

[Removing an ML model template](#)

[Changing the parameters of an ML model](#)

[Training an ML model predictive element](#)

[Training an elliptic envelope-based ML model element](#)

[Viewing the training results of an ML model element](#)

[Starting and stopping ML model inference](#)

[Viewing the data flow graph of an ML model](#)

[Preparing an ML model for publication](#)

[Publishing an ML model](#)

[Removing an ML model](#)

[Managing presets](#)

[Viewing a preset](#)

[Creating a preset](#)

[Loading presets from a file](#)

[Editing a preset](#)

[Saving presets to a file](#)

[Delete presets](#)

[Managing services](#)

[Viewing the statuses of services](#)

[Starting, stopping, and restarting services](#)

[Troubleshooting](#)

[When connecting to Kaspersky MLAD, the browser displays a certificate warning](#)

[The hard drive is running out of free space](#)

[The operating system restarted unexpectedly](#)

[Cannot connect to the Kaspersky MLAD web interface](#)

[Data graphs or graphic areas are not displayed in the History and Monitoring sections](#)

[Events are not transmitted between Kaspersky MLAD and external systems](#)

[Cannot load data to view in the Event Processor section](#)

[Data is incorrectly processed in the Event Processor section](#)

[Events are not displayed in the Event Processor section](#)

[Previously created monitors and the specified attention settings are not displayed in the Event Processor section](#)

[A markup result is not displayed](#)

[A Trainer service stopped message is displayed](#)

[Training of an ML model element completed with an error](#)

[Email notifications about incidents are not being received](#)

[You need to change the Help localization language](#)

[Contacting Technical Support](#)

[Limitations](#)

[Appendix](#)

[Settings of a .env configuration file](#)

[Settings and example of the Excel file containing tag and asset configuration](#)

[Settings and an example of JSON file that describes presets](#)

[Settings and an example of JSON file containing a configuration for the Event Processor service](#)

[Viewing the Kaspersky MLAD log](#)

[Scenario: viewing information security event logs](#)

[Scenario: assessing the main metrics of Kaspersky MLAD](#)

[Scenario: viewing container logs and metrics](#)

[Special characters of regular expressions](#)

[Cipher suites for secure TLS connection](#)

[Glossary](#)

[Account role](#)

[AMQP topic](#)

[Anomaly](#)

[Artifact](#)

[Asset](#)

[Attention](#)

[Connector](#)

[Data sampling](#)
[Event](#)
[Gradient boosting](#)
[Graphic area](#)
[ICS](#)
[Incident](#)
[Inference](#)
[Inference indicator](#)
[Learning indicator](#)
[Markup](#)
[ML model](#)
[Monitor](#)
[Monitored asset hierarchical structure](#)
[MQTT topic](#)
[Notification](#)
[Pattern](#)
[Preset](#)
[Tag](#)
[Top tag](#)
[Uniform temporal grid \(UTG\)](#)
[Information about third-party code](#)
[Trademark notices](#)

About Kaspersky Machine Learning for Anomaly Detection

The early anomaly detection system known as Kaspersky Machine Learning for Anomaly Detection (hereinafter also referred to as Kaspersky MLAD or "the application") is specialized software designed to prevent failures, accidents or degradation of industrial installations, technological processes, and complex cyberphysical systems. By analyzing telemetry data using machine learning techniques (artificial intelligence), Kaspersky MLAD detects signs of an abnormal situation before it is detected by traditional monitoring systems.

Kaspersky MLAD detects anomalies in industrial processes regardless of their causes. Anomalies may be caused by the following:

- Physical factors, such as damage to equipment or malfunctioning sensors.
- The human factor, such as intentional or inadvertent inappropriate actions by the operator, hardware configuration, change of operating mode or settings, or a switch to manual control.
- Cyberattacks.

Main capabilities of Kaspersky MLAD:

- [Detects abnormal behavior of the monitored asset](#) in real time.
- Identifies [signals that display the largest deviations](#) from normal behavior.
- Allows you to analyze incidents taking into account information about [similar incidents](#).
- Allows [expert classification and annotation of incidents](#).
- Allows you to [notify users about detected incidents through the web interface, by email](#), by sending messages to Kaspersky Industrial CyberSecurity for Networks, and using industrial data transfer protocols.
- Allows you to use [models](#) based on both machine learning and arbitrary rules for anomaly detection.
- Displays [historical](#) and [real-time](#) data as graphs according to the specified [tag sets](#), along with the results of processing this data with ML models.
- Lets you manage the [log of detected incidents](#).
- Allows you [to create ML models](#) and [add predictive elements, elliptic envelope-based elements, and diagnostic rule-based elements](#) to it.
- Provides [training of predictive elements](#) and [elliptic envelope-based elements](#).
- Allows to [create templates based on the added ML models](#) and [add ML models to Kaspersky MLAD based on the created templates](#).
- Allows you to [define the way to organize the data of the monitored asset](#) in the form of an asset tree.
- Allows you to receive telemetry data over [HTTP](#), [OPC UA](#), [MQTT](#), [AMQP](#), [CEP](#), and [WebSocket](#) protocols, and via a specialized protocol over HTTPS from Kaspersky Industrial CyberSecurity for Networks.
- Detects and handles terminations and interruptions of the incoming data stream, and restores missed observations.
- Based on data on events received from external systems, [recognizes principles as repeated events or patterns](#), and [identifies new events and patterns](#) in the event stream.

- [Displays the detected events](#) as a graph and a table, and shows [detected patterns](#) as a layered hierarchy of nested items.
- Sends [alerts about the detection](#) of certain events, patterns, or values of the event parameters received by the Event Processor in the data stream from the monitored asset.

Distribution kit

Kaspersky MLAD is delivered as an archive file named Kaspersky_MLAD_5.0.0.-<build number>_ru-RU_en-US.tar.xz, which contains the following files:

- Installation script and all files required for system installation.
- Scripts for updating, checking, and backing up the application.
- Files containing the text of the End User License Agreement in English and in Russian.
- Files containing information about the application (Release Notes) in English and in Russian.
- File containing information about third-party code (legal_notices.txt) in English.

After you unpack the archive, the "legal" directory will contain a text file named license_en.txt in which you can view the End User License Agreement. The End User License Agreement specifies the terms of use of the application.

Hardware and software requirements

The hardware requirements for each protected facility must be adjusted considering the model being used, the number of processed tags and events, the average speed of data acquisition (number of observations per second), and the volume of stored data. The more data is processed and the more sophisticated the used ML model is, the more hardware resources are required for installing the server part of Kaspersky MLAD.

Requirements for Kaspersky MLAD server

To ensure proper operation of the application, the Kaspersky MLAD server must meet the following minimum requirements.

List of supported processors:

- Intel® Xeon® E3 v3, v4, v5, v6
- Intel Xeon E5 v3, v4
- Intel Xeon E7 v3, v4
- Intel Xeon Scalable processors
- The 2nd and 3rd generation Intel Xeon Scalable processors
- Intel Xeon E

- Intel Xeon W
- Intel Xeon D
- The 4th generation and later Intel Core™ i5, i7
- Intel Core i9 processor
- Intel Core M

Minimum hardware requirements:

- 8 cores
- 32 GB of RAM
- 200 GB of free space on the hard drive (SSD recommended)

If Kaspersky MLAD receives a large data stream, increase the amount of free space on the hard drive.

You can install Kaspersky MLAD on a server with another x86 64-bit processor released in 2013 or later. The processor must meet the minimum hardware requirements listed above and support the following extensions required for the TensorFlow™ 2.15.1 library:

- Advanced Vector Extensions (avx)
- Advanced Vector Extensions 2 (avx2)

Supported operating systems:

- Ubuntu 22.04 LTS or later

The following software must be installed prior to deployment of Kaspersky MLAD:

- docker 24.0.9 or later
- docker compose 2.12.2 or later

Use the [official Docker repository](#) for installation of the software on the Kaspersky MLAD server.

To update and back up the application, the jq utility must be installed. You can install the jq utility by using the apt package manager.

User computer requirements

To work with the web interface of Kaspersky MLAD, the user's computer must meet the following minimum requirements:

- Intel Core™ i5 CPU;
- 8 GB of RAM;
- 64-bit operating system;

- Google Chrome™ browser version 107 or later
- The minimum screen resolution to display the web interface properly is 1600x900.

Security recommendations

To ensure secure operation of Kaspersky MLAD at an enterprise, it is recommended to restrict and control access to equipment on which the application is running.

Physical security of equipment

When deploying Kaspersky MLAD, it is recommended to take the following measures to ensure secure operations:

- Restrict access to the room housing the server with Kaspersky MLAD installed, and to the equipment of the dedicated network. Access to the room must be granted only to trusted persons, such as personnel who are authorized to install and configure the application.
- Employ technical resources or a security service to monitor physical access to equipment on which the application is running.
- Use security alarm equipment to monitor access to restricted rooms.
- Conduct video surveillance in restricted rooms.

Information security

The ML model settings directly affect anomaly detection, so only system administrators and users in the [Manage ML models](#) group are allowed to edit these. The change history is available only in application logs, which are saved for only a limited amount of time.

When using the web interface, it is recommended to take the following measures to ensure the data security of the intranet system:

- Provide users with access to the application through the web interface only.
- Install certificates to users' computers for authorization of the Kaspersky MLAD server with their browser. To [use a trusted certificate](#), you need to contact a qualified technical specialist of the Customer, a Kaspersky employee, or a certified integrator.
- Ensure protection of traffic within the intranet system.
- Ensure protection of connections to external networks.
- Use a secure TLS connection for data transfer.
- Change the name and password of the first application user with the system administrator role when [installing the application](#).
- Change the account password when it expires. The password expiration date is defined in the [application security settings](#). The default password expiration is set to 180 days.
- For connections through the web interface, use passwords that meet the following requirements:

- Must not match previously used account passwords. The specific number of most recently used passwords that must not be reused is defined when [configuring the application security settings](#). The password must be different from the five previous passwords by default.
- Must contain at least 8 characters.
- Must contain one or more uppercase letters of the English alphabet.
- Must contain one or more lowercase letters of the English alphabet.
- Must contain one or more numerals.
- Must contain one or more of the following special characters: _ ! @ # \$ % ^ & *.
- Ensure that passwords are confidential and unique. If the password has been possibly compromised, change the password.
- Set a time limit for a user web session.
- After you are finished working in the browser, manually [terminate the application connection session](#) by using the **Sign out** option in the web interface.
- Periodically install updates for the operating system on the server where Kaspersky MLAD is deployed.
- Use access permission control to [restrict user access](#) to application functions.

Data security

While working with Kaspersky MLAD, it is recommended to take the following measures to ensure data security:

- Configure the operating system and provide the necessary access to files of the server where Kaspersky MLAD is installed in accordance with the *Recommendations on secure configuration of Linux operating systems* issued by the Federal Service for Technical and Export Control (FSTEC) of Russia.
- Perform periodic data backups of the server that has Kaspersky MLAD installed in accordance with the internal company procedure.
- Periodically test the performance of the interface and services of the application. Special attention should be directed to the notification service and logging system.
- Check communication channels to make sure they are secure and working properly.
- Periodically test the performance of the server:
 - SMART disk check
 - Availability of sufficient free space and memory
 - RAM utilization
- Use the monitoring system to make sure that there are no problems with the server protocols.
- Store sensitive data in a secure storage location.

Fixing vulnerabilities and installing critical updates

Kaspersky may release [application updates](#) aimed at eliminating vulnerabilities and security flaws (critical updates). Urgent update packages are supplied and installed in accordance with the current *Technical Support Agreement*. Notifications regarding the release of critical updates are sent to the email addresses specified in the current *Technical Support Agreement*.

It is recommended that the personnel responsible for application operation also periodically (at least once every three months) verify the absence of detected vulnerabilities in the application by referring to the [Kaspersky website](#).

You can report security flaws or program vulnerabilities with a PGP™-encrypted message to vulnerability@kaspersky.com. Please provide the following information in your email:

- Your contact details.
- The product name, version, and type of operating system installed on the asset where the vulnerability was found.
- A detailed description of the vulnerability.
- Any plans to share information about the vulnerability with a third party.

Do not publish any information about the vulnerability until fixed by Kaspersky.

Managing access to application functions

In Kaspersky MLAD, you can use [roles](#) to restrict users' access to application functions depending on the tasks performed by users. A *role* is a set of rights to access application functions that you can assign to a user.

Depending on the assigned role, users may have access to the following functions of Kaspersky MLAD:

Accessible functions of the application

Functional scope	System administrator	Custom role
Managing license keys: <ul style="list-style-type: none">• Adding license keys• Removing license keys	✓	—
Managing user accounts: <ul style="list-style-type: none">• Creating and editing a user account• Revoking authentication tokens for a user account• Viewing rights for a user account	✓	—

<u>Managing roles:</u> <ul style="list-style-type: none"> • <u>Creating and changing a role</u> • <u>Deleting role</u> • <u>Viewing access rights for a role</u> 	✓	—
Viewing the rights of users	✓	—
<u>Managing incident notifications:</u> <ul style="list-style-type: none"> • <u>Creating and editing incident notifications</u> • <u>Deleting incident notifications.</u> 	✓	—
<u>Configuring Kaspersky MLAD</u>	✓	—
<u>Managing assets:</u> <ul style="list-style-type: none"> • <u>Creating and modifying an asset</u> • <u>Creating a tag and adding a tag to an asset</u> • <u>Editing a tag</u> • <u>Moving assets and tags</u> • <u>Deleting an asset or tag</u> • <u>Checking the current structure of tags</u> • <u>Importing and exporting</u> asset configurations 	✓	—
Managing ML models: <ul style="list-style-type: none"> • <u>ML model importing:</u> • <u>Activating an imported ML model</u> • <u>Creating an ML model</u> • <u>Changing an ML model</u> • <u>Adding and modifying a predictive element of an ML model</u> • <u>Adding and modifying an elliptic envelope-based element of an ML model</u> • <u>Adding and modifying an ML model element based on a diagnostic rule</u> • <u>Removing an ML model element</u> • <u>Cloning an ML model</u> • <u>Creating and modifying a template based on an ML model</u> 	✓	✓ (if assigned)

<ul style="list-style-type: none"> • Creating an ML model based on a template • Removing an ML model template • Training predictive elements and elliptic envelope-based elements • Viewing the training results of an ML model elements: • Preparing an ML model for publication • Publishing an ML model • Removing an ML model 		
Managing Kaspersky MLAD services : <ul style="list-style-type: none"> • Viewing the statuses of services • Starting, stopping, and restarting services 	✓	✓ (if assigned)
Manage application logs	✓	✓ (if assigned)

All application users have the following default rights:

- [Viewing summary data](#) in the **Dashboard** section
- View primary and operational data by tags in the **History** and **Monitoring** sections
- Viewing the values of the process parameters received from the monitored asset's sensors at a certain point in time in the **Time slice** section.
- [Working with events and patterns](#):
 - Configuring attention settings and display of event parameters
 - Creating and deleting monitors
 - Viewing the event and pattern history
- [Working with incidents and groups of incidents](#):
 - Viewing incidents and incident groups
 - Adding a status, cause, expert opinion or note to an incident or incident group
 - Exporting incidents to a file
- The following actions in the **Models** section:
 - [Creating, modifying, and deleting markups](#)
 - [Starting and stopping ML model inference](#)
 - [Viewing the data flow graph of an ML model](#)

- [Manage presets:](#)
 - View presets
 - Create, modify, and delete models
 - Load a preset configuration from a file
 - Save a preset configuration to a file
- [Change your own password](#)

You can also create a role with a **Rights to all actions** permission. Users with this role have access to system administrator functions.

You can view the available user roles and their access rights to application functions in the **Roles** section of [the administrator menu](#).

You can view application functionality access rights for specific users in the **Users** section of [the administrator menu](#).

What's new

Kaspersky Machine Learning for Anomaly Detection 5.0 has the following new capabilities and improvements:

- Model builder: added support for [elliptic envelope-based ML models](#). Added functionality that lets you log incidents when observing anomalous behavior for a certain interval of time. Added functionality that lets you [clone elements of ML models](#) and [markups](#). Added display of statuses and states of ML models and their elements in the asset tree. Added functionality that lets you [search and filter](#) by elements of the asset tree.
- Presets: added functionality that lets you [combine preset tags into graphic areas](#) and control the settings for displaying tag charts in a graphic area.
- **Monitoring** and **History** sections: added support for displaying charts of several tags within the same graphic area. Graphic areas and the tags within them are managed in the **Presets** section.
- Licensing: added functionality that lets you [add license keys](#) for different [types of licenses](#) depending on the required [set of functions](#).
- Application installation, update and backup scripts: improved algorithms for application installation and update scripts in accordance with information security requirements. Added functionality that lets you [switch between Kaspersky MLAD state control modes](#) by using the installation script. The backup script provides the functionality for [backing up the application](#) and [rolling back the application to a previously installed version](#).
- Starting and stopping the application: added functionality that lets you [start and stop the application](#) by using the systemctl utility.
- User accounts: added capability to [change email addresses](#) for user accounts in the web interface.
- Incident notifications: added capability to specify additional email addresses for sending incident notifications in the web interface. Added functionality that lets you [send notifications when registering certain types of incidents](#): incidents registered by certain types of ML model elements or registered by the Limit Detector and/or Stream Processor service. Added functionality that lets you prevent forwarding of notifications about incidents registered by elements of unpublished ML models.
- Event Processor service: expanded functionality of the [attention mechanism](#). Added functionality that lets you track generalized events and patterns. Improved display of monitors.
- Mail Notifier Service: Added new settings for [configuring the Mail Notifier service](#).
- HTTP Connector and OPC UA Connector: added new settings for [configuring the HTTP Connector](#) and [OPC UA Connector](#).
- MQTT Connectors, AMQP Connector and CEF Connector - added support for TLS connections by default, and added functionality that lets you use the recommended TLS connection settings.
- WebSocket Connector: added functionality that lets you use the recommended TLS connection settings.
- Push messages: added functionality that lets you transfer messages between Kaspersky MLAD services.

Kaspersky MLAD architecture

Kaspersky MLAD is installed on a server that meets the [hardware and software requirements](#). The Kaspersky MLAD Server centrally stores information about application services and connectors and provides a single web user interface for managing them.

Access to individual services or application connectors is not provided.

When installing Kaspersky MLAD, all application connectors and services are hosted on the same server and interact with each other through an internal virtual network that is isolated from external systems.

Kaspersky MLAD includes specially prepared ML models, and the following services and connectors:

ML model

An *ML model* is a model created for a specific facility based on machine learning algorithms and/or diagnostic rules using telemetry data from this facility. The [ML model](#) detects anomalies.

An ML model can be [created with a model builder](#) or provided under *Kaspersky MLAD Model-building and Deployment Service*.

Kaspersky MLAD services

Kaspersky MLAD services comprise a set of core application services supplied to each monitored asset. Kaspersky MLAD includes the following services:

- *Anomaly Detector*. Uses an ML model to process data and detect anomalies.
- *Event Processor*. Uses machine learning methods based on a semantic neural network to identify patterns and anomalous sequences of events.
- *Stream Processor*. Processes telemetry data received from the monitored asset at arbitrary real-time moments and converts this data to a uniform temporal grid.
- *Trainer*. Performs training of an ML model based on the telemetry data obtained by Kaspersky MLAD for a specific monitored asset.
- *Similar Anomaly*. Identifies and groups together similar incidents.
- *Message Broker*. Exchanges data between Kaspersky MLAD services.
- *Time Series Database*. Stores time series of observed tag values, [artifacts](#) associated with tags, and ML model element artifacts.
- *Keeper*. Performs routing of the telemetry data that should be saved in the database.
- *Database*. Stores all configuration settings of Kaspersky MLAD.
- *API Server*. Supports operation of the internal interfaces of Kaspersky MLAD.
- *Web Server*. Supports operation of the Kaspersky MLAD web interface.

- *Logger*. Stores Kaspersky MLAD operation logs.
- *Mail Notifier*. Sends emails with incident registration notifications.
- *Docker API Server*. Supports interaction with Docker.
- *Migrations*. Updates information about the Kaspersky MLAD settings in the Database.
- *Push Server*. Sends push messages to Kaspersky MLAD.

Connectors

Connectors are services that facilitate the exchange of data with external systems. For each protection object, you must select one of the following connectors:

- *KICS Connector*. Supports interaction with Kaspersky Industrial CyberSecurity for Networks version 4.0 and later.
- *OPC UA Connector*. Receives tags from industrial process control systems (ICS) according to the protocol described in the OPC Unified Architecture specification.
- *CEF Connector*. Receives events from external sources (Industrial Internet of Things, network devices and applications) and returns messages in CEF (Common Event Format) registered by event analysis monitors.
- *MQTT Connector*. Receives tags from ICS and sends messages about incidents via the MQTT (Message Queuing Telemetry Transport) protocol.
- *AMQP Connector*. Receives tags from ICS and sends messages about incidents via AMQP (Advanced Message Queuing Protocol).
- *WebSocket Connector*. Receives tags from ICS and sends messages about incidents via the WebSocket protocol.
- *HTTP Connector*. Receives telemetry data from ICS in CSV files via HTTP/HTTPS POST requests.

The figure below shows a diagram of interaction between Kaspersky MLAD services.

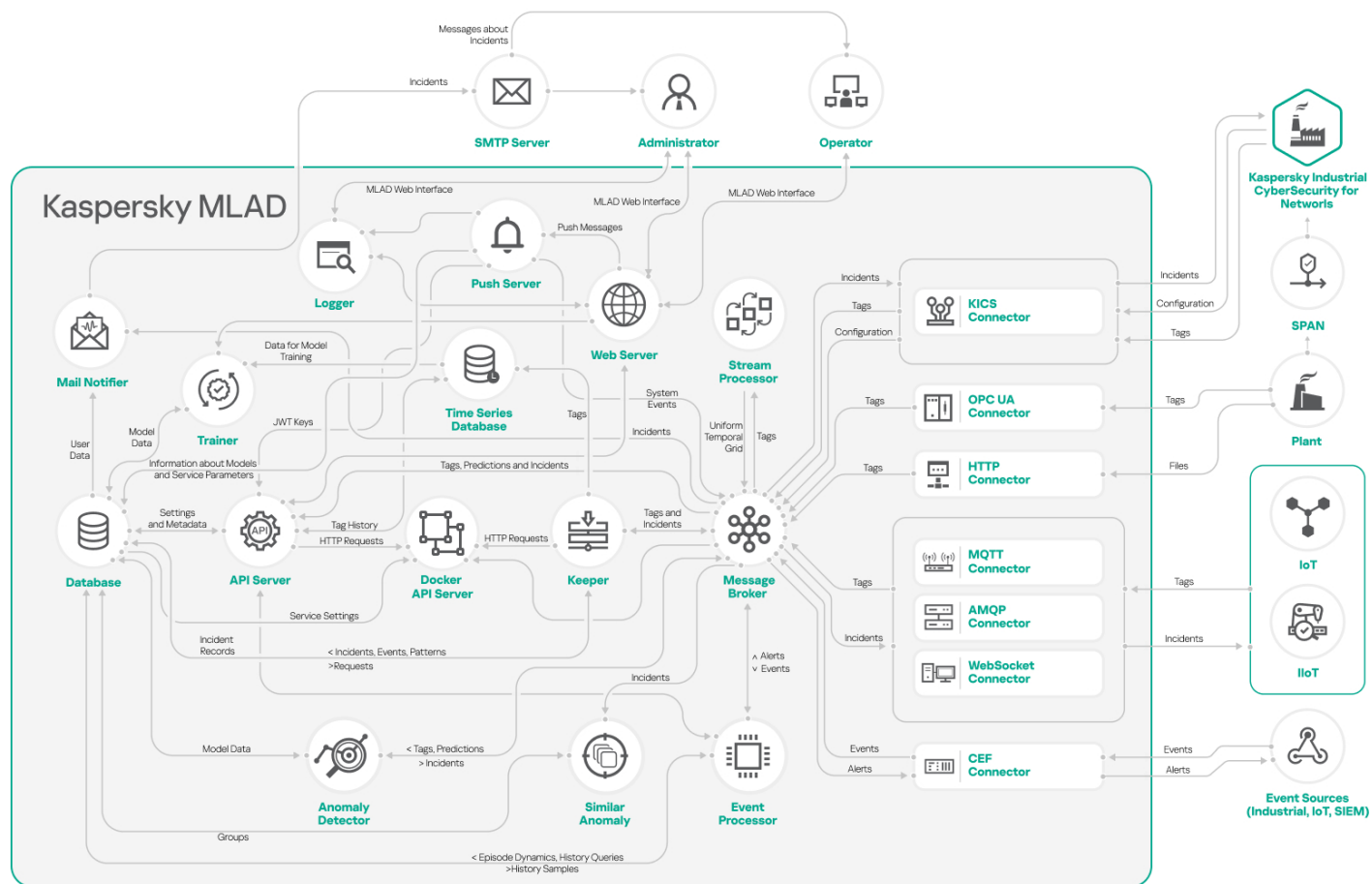


Diagram of interaction between Kaspersky MLAD services

Common deployment scenarios

This section provides a description of the standard scenarios for deploying Kaspersky MLAD in the network of a monitored asset, and provides special considerations when integrating Kaspersky MLAD with other applications.

Kaspersky MLAD supports the following installation options:

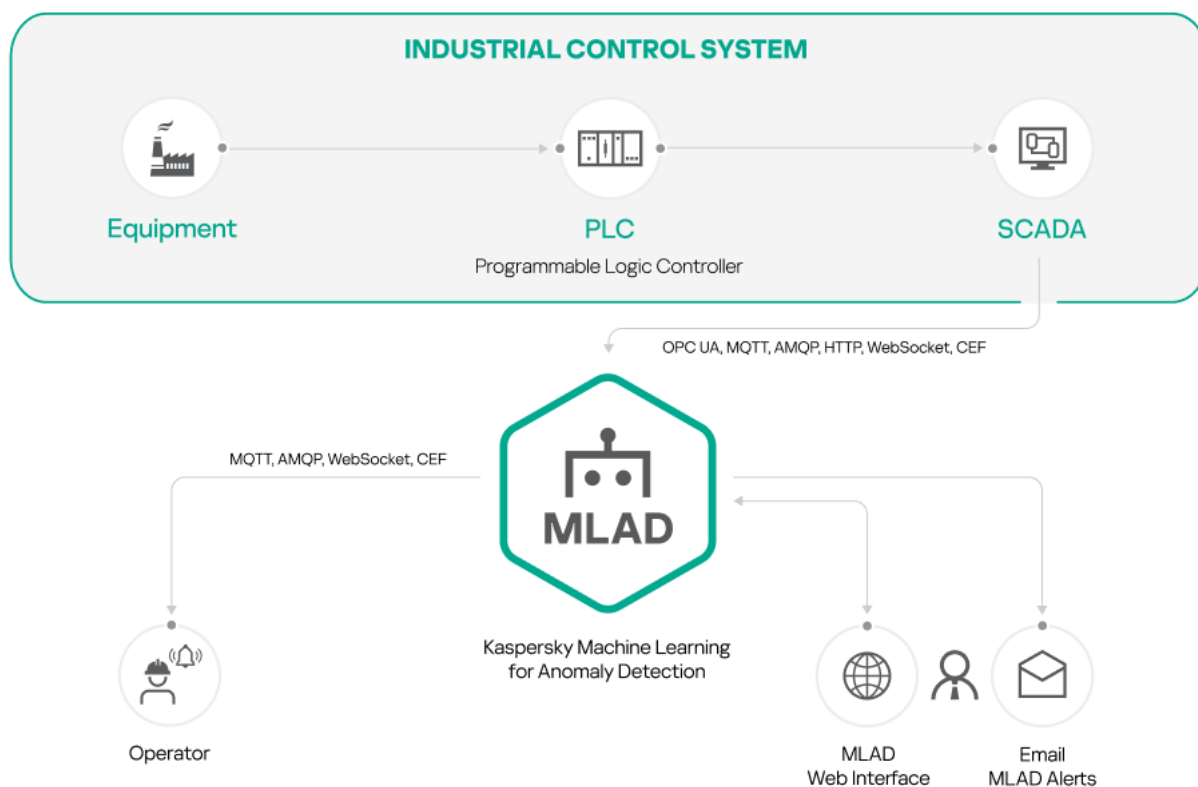
- Standalone installation.
- Installation with Kaspersky Industrial CyberSecurity for Networks version 4.0 and later.

Standalone installation of Kaspersky MLAD

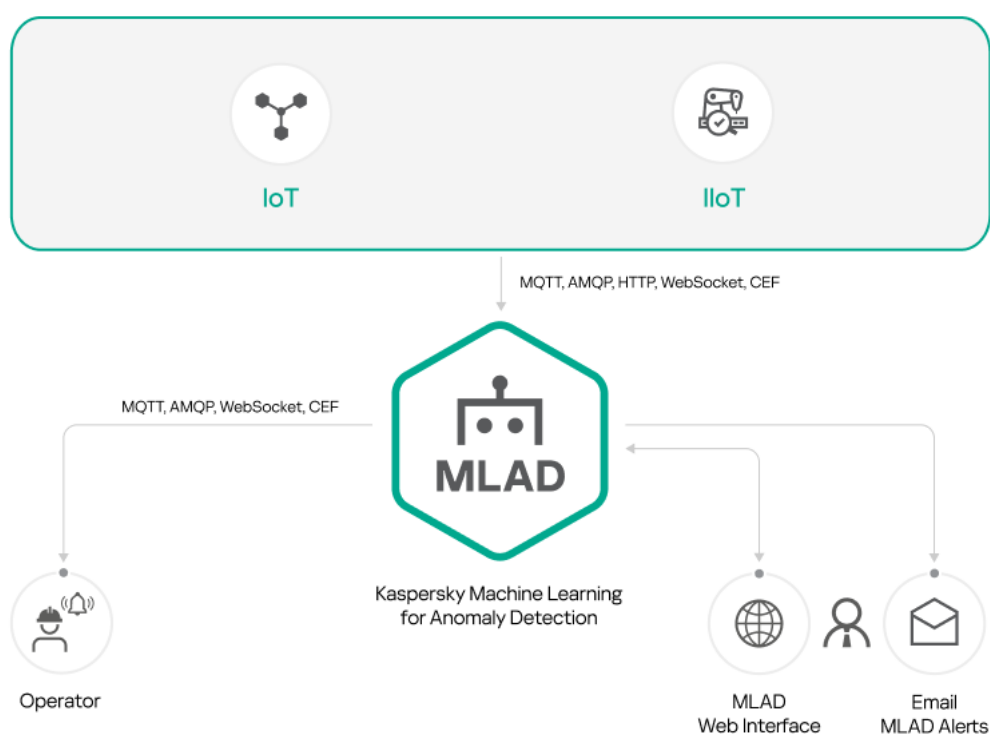
You can install only Kaspersky MLAD if you plan to use the following connectors as a data provider:

- OPC UA Connector
- MQTT Connector
- AMQP Connector
- CEF Connector
- WebSocket Connector
- HTTP Connector

The figures below show example scenarios for standalone installation of Kaspersky MLAD using the above connectors. You can use any configurations of connectors that are suitable for your monitored asset.



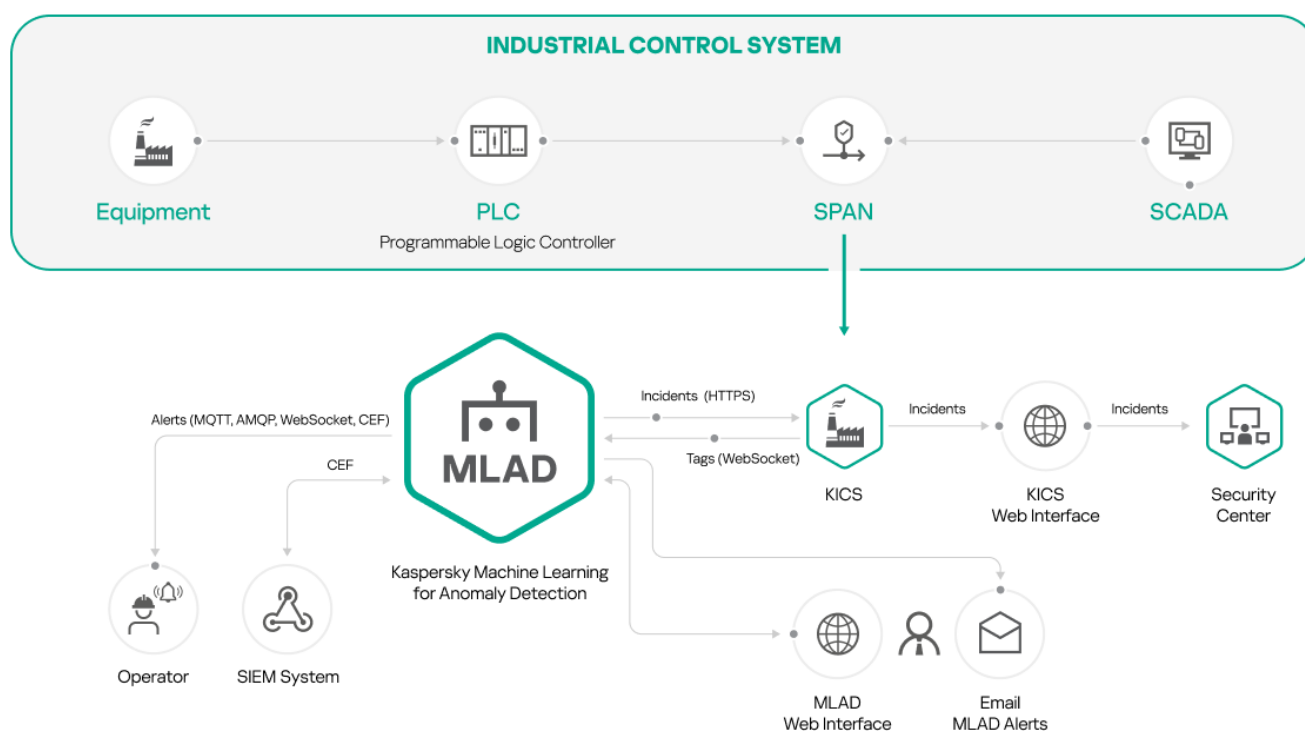
Standalone installation of Kaspersky MLAD using connectors: OPC UA Connector, MQTT Connector, AMQP Connector, HTTP Connector, WebSocket Connector, CEF Connector



Installation of Kaspersky MLAD with Kaspersky Industrial CyberSecurity for Networks

You can install Kaspersky MLAD and Kaspersky Industrial CyberSecurity for Networks if you are planning to use Kaspersky Industrial CyberSecurity for Networks as a data provider (see the figure below).

Kaspersky Machine Learning for Anomaly Detection is compatible with Kaspersky Industrial CyberSecurity for Networks version 4.0 and later.



Installation of Kaspersky MLAD with Kaspersky Industrial CyberSecurity for Networks

To use this installation option, first install Kaspersky Industrial CyberSecurity for Networks and add a **Generic** connector. Create a communication data package for the added connector and specify the settings for connecting Kaspersky Industrial CyberSecurity for Networks to Kaspersky MLAD. Upload the obtained communication data package to Kaspersky MLAD when [configuring the KICS Connector](#). For detailed information about creating and adding a connector, please refer to the *Adding a connector* section of *Kaspersky Industrial CyberSecurity for Networks Help Guide*.

Computers with Kaspersky MLAD and Kaspersky Industrial CyberSecurity for Networks installed must belong to the same network.

Telemetry and event data flow diagram

In Kaspersky MLAD, data exchange with the external systems is provided by connectors. To receive telemetry data (tags) and/or events from the external systems, you need to [configure the HTTP Connector](#), [MQTT Connector](#), [AMQP Connector](#), [OPC UA Connector](#), [KICS Connector](#), [CEF Connector](#), and [WebSocket Connector](#).

If transmission of events and incidents to recipient systems is configured in the application, the application sends registered events and incidents to recipient systems chosen by the system administrator. The application system administrator independently selects the recipient systems and the types of events and incidents to transmit to the recipient systems. The recipient system processes and stores the received data according to its functionality and purpose.

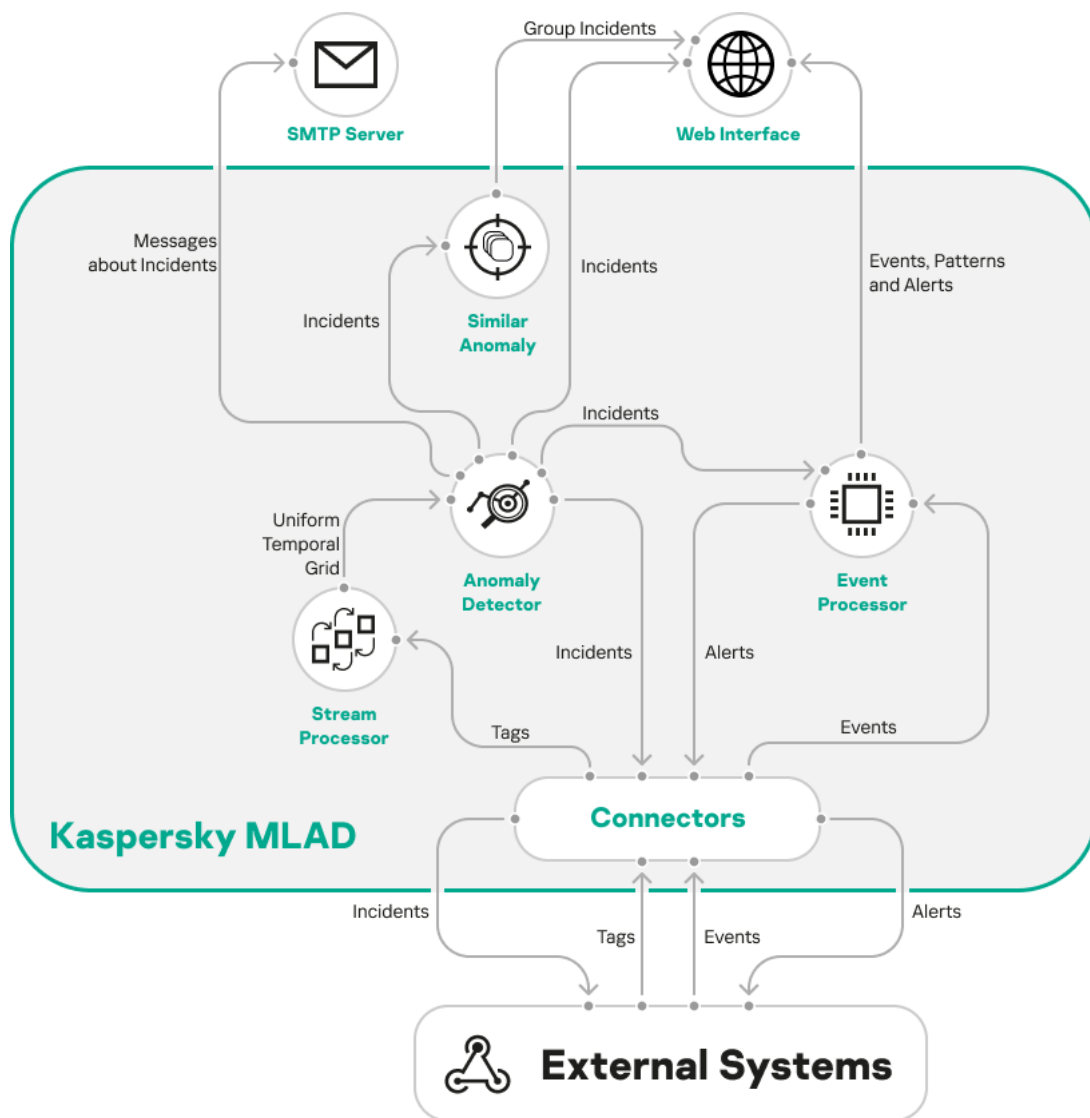
The Stream Processor service performs the initial processing of the telemetry data of the monitored asset, converting the received [tags](#) to a uniform temporal grid (UTG). When Stream Processor service detects interruptions in the telemetry data stream or observations received by Kaspersky MLAD too early or too late, [it registers incidents](#).

The Stream Processor service transfers the UTG-converted data to the [ML model](#) of the Anomaly Detector service. If the elements of the ML models detect deviations from the normal behavior of the monitored asset while processing the received data, the Anomaly Detector service registers [incidents](#). When similar incidents are detected, the Similar Anomaly service generates groups of incidents.

You can [view registered incidents and groups of incidents](#) in the **Incidents** section. Kaspersky MLAD also sends incident notifications to the [specified email addresses](#) and to external systems using connectors.

[Events](#) received by Kaspersky MLAD are processed by the Event Processor service. The Event Processor can also [process incidents registered by the Anomaly Detector service](#). In the stream of events, the Event Processor detects regularities – recurring events and [patterns](#) – as well as new events and patterns. When [monitors](#) are activated, the Event Processor service sends alerts to external systems about the detection of events, patterns, and event parameter values according to the [specified monitoring criteria](#) using the CEF Connector. You can also [view information about events, patterns, and monitors](#) in the **Event Processor** section.

The figure below shows the telemetry and event data stream in Kaspersky MLAD.



The telemetry and event data stream in Kaspersky MLAD

Ports used by Kaspersky MLAD

The table below lists the ports that must be opened on the servers where Kaspersky MLAD is installed.

Port	Protocol	Description
443	TCP (HTTPS)	Used to connect to the Kaspersky MLAD web interface.
3001	TCP (HTTPS)	Used to connect to the logging system (Grafana ™).
4999	TCP (HTTP or HTTPS)	Used by the HTTP Connector to download CSV files from external sources.
5518	TCP	Used to connect external event sources to the default CEF Connector. The port number is defined in the .env configuration file .

Installing and removing the application

This section contains step-by-step instructions on installing and removing Kaspersky MLAD.

Installing the application

This section contains a step-by-step description of Kaspersky MLAD installation. During installation, Kaspersky MLAD creates the first application user with the system administrator role.

Prior to installation, you must make sure that the [required amount of free space](#) is available on the hard drive where the application will be installed. Docker service volumes must be stored on the drive where the application is installed. If the Docker volumes are stored on a different drive, you must move them and use the Docker configuration file to specify the path to the storage location of the volumes on the hard drive where the application is installed.

To install the application, each server must have a user account with root privileges that will be used to perform the installation. The directory for installing Kaspersky MLAD must be empty.

Installation of Kaspersky MLAD is performed by a qualified technical specialist of the Customer, a Kaspersky employee, or a certified integrator.

The Kaspersky MLAD server and the software installed on the server must meet the [hardware and software requirements](#).

Kaspersky MLAD is installed according to the described procedure for application installation. Installation and use of Kaspersky MLAD is possible only on one server. Installation and use of different services and connectors on multiple servers is not possible.

Installation of Kaspersky MLAD will be interrupted if the integrity of the application archive has been breached. To obtain the correct archive of the application, please contact Kaspersky experts.

To install Kaspersky MLAD:

1. Unpack the archive named Kaspersky_MLAD_5.0.0-<build number>_ru-RU_en-US.tar.xz that is included in the [distribution kit](#):

```
tar xf Kaspersky_MLAD_5.0.0.< build number >_ru-RU_en-US.tar.xz
```

The mlad-release-5.0.0-<build number> directory appears after the archive is unpacked.

2. Navigate to the directory named mlad-release-5.0.0-<build number>:

```
cd mlad-release-5.0.0-< build number >
```

3. Run the setup.sh installation script:

```
sudo ./setup.sh
```

4. Follow the instructions of the Application Setup Wizard.

Read the License Agreement carefully during installation. You must accept the terms of the End User License Agreement to install the application. If you do not accept the terms of the End User License Agreement, the installation process will be interrupted.

Using the Application Setup Wizard, you can change the name and password of the first application user with the system administrator role.

The application is installed in `/opt/kaspersky/mlad` by default. You can specify a different directory during installation.

To install Kaspersky MLAD in non-interactive mode:

1. Unpack the archive named `Kaspersky_MLAD_5.0.0-<build number>_ru-RU_en-US.tar.xz` that is included in the [distribution kit](#):

```
tar xf Kaspersky_MLAD_5.0.0.< build number >_ru-RU_en-US.tar.xz
```

2. Navigate to the directory named `mlad-release-5.0.0-<build number>`:

```
cd mlad-release-5.0.0-< build number >
```

3. Run the `setup.sh` installation script with the following switches:

```
sudo ./setup.sh -q -e accept -f <full path to installation directory>
```

where:

`-q` means that the application is installed in non-interactive mode. When installing the application in non-interactive mode, Kaspersky MLAD creates the first application user with the system administrator role and assigns it a default user name and password. To obtain the default user name and password, contact a qualified technical specialist of the Customer, a Kaspersky employee, or a certified integrator.

`-e accept` means that you accept the terms of the End User License Agreement. You must accept the terms of the End User License Agreement to install the application. If you do not add the `-e accept` switch, installation of the application will not continue.

You can read the text of the End User License Agreement in the text file named `license_en.txt` located in the 'legal' directory.

`-f <full path to installation directory>` means the application will be installed in the specified directory. Skipping the `-f` switch will cause the application to be installed in the default directory `/opt/kaspersky/mlad`.

The application will be installed on the server. After installing the application, [start](#) it.

Some features will be unavailable until you add a [license key](#).

Updating the application

The application is updated using the `upgrade.sh` upgrade script. When Kaspersky MLAD is updated, all of the following data that was uploaded, received, or processed by the previous version of Kaspersky MLAD will be saved: tag configurations, presets, ML models, and settings of Kaspersky MLAD.

You can back up the previous version when updating the application, if needed.

A user account in the Kaspersky MLAD server operating system must have root access to update the application.

Prior to starting the update, make sure that there is free space on the hard drive:

- If the application is being updated without performing a backup, the hard drive must have enough free space [required to install Kaspersky MLAD](#).

- If a backup is performed simultaneously with the application update and the backup copy is saved on the same drive, at least 50% of the total hard drive volume must be free.
- If a backup copy is performed simultaneously with the application update and the backup copy is saved on another drive, the application installation drive must have free space in the amount [required to install Kaspersky MLAD](#), and the drive for storing the backup copy must have free space equaling at least the amount of occupied disk space on the drive where the application is installed.

Updating Kaspersky MLAD is possible starting with application version 5.0.0-001.

The application will shut down while it updates. Kaspersky MLAD will not accept data from data sources or process it.

The Kaspersky MLAD server and the software installed on the server must meet the [hardware and software requirements](#).

Kaspersky MLAD is updated to fix security flaws and application vulnerabilities or when new versions of the application are released under the current *Technical Support Agreement*. The application update is performed by a qualified technical specialist of the Customer, a Kaspersky employee, or a certified integrator.

The Kaspersky MLAD update will be interrupted if the integrity of the application archive has been breached. To obtain the correct archive of the application, please contact Kaspersky experts.

To update Kaspersky MLAD:

1. Unpack the archive named mlad-5.0.0-<new build number>.tar.xz that is included in the [distribution kit](#):

```
tar xf mlad-5.0.0-< new build number >.tar.xz
```

2. Go to the folder with the new application build:

```
cd mlad-release-5.0.0-< new build number >
```

3. Run the upgrade.sh script using one of the following methods:

- If you need to back up a previous version and save the backup copy in the directory where Kaspersky MLAD is installed, run the following command:

```
sudo ./upgrade.sh -f <full path to the application build to be updated>
```

The backup copy will be created in a directory named mlad_backup-<version number>-<build number>. The directory will be created within the directory where the application is installed.

- If you need to back up a previous version and save the backup copy in a different directory, run the following command:

```
sudo ./upgrade.sh -b <full path to backup directory> -f <full path to application build to be updated>
```

- To skip backup when updating the application, run:

```
sudo ./upgrade.sh -b nobackup -f <full path to the application build to be updated>
```

You can run the upgrade.sh script with the -h switch if you want to display the brief description of the script in the Kaspersky MLAD update interface:

```
sudo ./upgrade.sh -h
```

4. Follow the instructions of the Application Upgrade Wizard.

Accept the terms of the End User License Agreement while running the Application Update Wizard. You must accept the terms of the End User License Agreement to update the application. If you do not accept the terms of the End User License Agreement, the update process will be interrupted.

You can read the text of the End User License Agreement in the text file named license_en.txt located in the 'legal' directory.

To update Kaspersky MLAD in non-interactive mode:

1. Unpack the archive named mlad-5.0.0-<new build number>.tar.xz that is included in the [distribution kit](#):

```
tar xf mlad-5.0.0-< new build number >.tar.xz
```

2. Go to the folder with the new application build:

```
cd mlad-release-5.0.0-< new build number >
```

3. Run the upgrade script using one of the following methods:

- If you need to back up a previous version and save the backup copy in the directory where Kaspersky MLAD is installed, run the following command:

```
sudo ./upgrade.sh -q -e accept -f <full path to the application build to be updated >
```

The backup copy will be created in a directory named mlad_backup-<version number>-<build number>. The directory will be created within the directory where the application is installed.

- If you need to back up a previous version and save the backup copy in a different directory, run the following command:

```
sudo ./upgrade.sh -q -e accept -b <full path to the backup directory> -f <full path to the application build to be updated>
```

- To skip backup when updating the application, run:

```
sudo ./upgrade.sh -q -e accept -b nobackup -f <full path to the application build to be updated>
```

where:

-q means that the application is updated in non-interactive mode.

-e accept means that you accept the terms of the End User License Agreement. You must accept the terms of the End User License Agreement to update the application. If you do not add the -e accept switch, application updating will be interrupted.

You can read the text of the End User License Agreement in the text file named license_en.txt located in the 'legal' directory.

-b <full path to the backup directory> means Kaspersky MLAD will back up the current application version and save the backup to the specified directory.

-b nobackup means that Kaspersky MLAD will update the application without backing up the current version.

-f <full path to the application build to be updated> means the application installed in the specified directory will be updated.

Kaspersky MLAD will be updated to the version specified in the build number. All application files will be located in the directory where Kaspersky MLAD is installed (/opt/kaspersky/mlad by default).

Checking the integrity of Kaspersky MLAD archive files

You can check the integrity of files in the [Kaspersky MLAD archive](#) to make sure that there have been no changes to its contents before beginning [installation](#) or [upgrading the application](#).

Integrity checks are performed using the integrity.sh script. When the script is running, it sequentially verifies the checksums of files from the application archive.

To check the integrity of Kaspersky MLAD archive files:

1. Unpack the archive named Kaspersky_MLAD_5.0.0-<build number>_ru-RU_en-US.tar.xz that is included in the [distribution kit](#):

```
tar xf Kaspersky_MLAD_5.0.0.< build number >_ru-RU_en-US.tar.xz
```

The mlad-release-5.0.0-<build number> directory appears after the archive is unpacked.

2. Navigate to the directory named mlad-release-5.0.0-<build number>:

```
cd mlad-release-5.0.0-< build number >
```

3. Run the script for checking the integrity of the Kaspersky MLAD archive:

```
./integrity.sh
```

The results of checking the integrity of the application archive files on the computer are considered successful if the integrity.sh script ends with the SUCCEEDED message.

Backing up the application

You can back up the application in accordance with your company regulations. You can back up Kaspersky MLAD when [updating the application](#), if needed.

The application is backed up with the help of the backup.sh script. The Kaspersky MLAD backup procedure saves all of the following data that was uploaded, received, or processed by Kaspersky MLAD: tag configurations, presets, ML models, and settings of Kaspersky MLAD.

A user account in the Kaspersky MLAD server operating system must have root access to back up the application.

Prior to starting the backup, you must make sure that at least 50% of the hard drive space is free if you are saving the backup copy to the hard drive where the application is installed. If the backup copy is saved to another drive, you must make sure that this drive has enough free space [necessary for installing Kaspersky MLAD](#).

Kaspersky MLAD backup capabilities are available starting with application version 5.0.0-001.

To back up Kaspersky MLAD:

1. Go to the directory where Kaspersky MLAD is installed:

```
cd mlad-release-5.0.0-< build number >
```

2. Run the backup.sh script using one of the following methods:

- If you want to save a backup copy in the directory where the application is installed, run the following command:

```
sudo ./backup.sh -f <full path to application directory>
```

The backup copy will be created in a directory named `mlad_backup-<version number>-<build number>`. The directory will be created within the directory where the application is installed.

- If you need to save the backup copy to another directory, run the following command:

```
sudo ./backup.sh -b <full path to backup directory> -f <full path to application directory>
```

3. Follow the instructions of the backup wizard.

To back up Kaspersky MLAD in non-interactive mode:

1. Go to the directory where Kaspersky MLAD is installed:

```
cd mlad-release-5.0.0-<build number>
```

2. Run the `backup.sh` script by doing one of the following:

- If you want to save a backup copy in the directory where the application is installed, run the following command:

```
sudo ./backup.sh -q -f <full path to application directory>
```

The backup copy will be created in a directory named `mlad_backup-<version number>-<build number>`. The directory will be created within the directory where the application is installed.

- If you need to save the backup copy to another directory, run the following command:

```
sudo ./backup.sh -q -b <full path to backup directory> -f <full path to application directory>
```

where:

`-q` means that the application will be backed up in non-interactive mode.

`-b <full path to the backup directory>` means Kaspersky MLAD will save the backup in that directory.

`-f <full path to application directory>` means that the application installed in that directory will be backed up.

Rolling back the application to the previous installed version

The application can be rolled back to a previously installed version by using the `backup.sh` script.

A user account in the Kaspersky MLAD server operating system must have root access to roll back the application to a previous version.

Kaspersky MLAD rollback capabilities are available starting with application version 5.0.0-001.

The application will shut down while it rolls back to the previous version. Kaspersky MLAD will not accept data from data sources or process it.

When rolling back Kaspersky MLAD to the previous installed version, all data received and processed by Kaspersky MLAD from the moment the application was upgraded to the moment of the rollback to the previous version will be lost. You are advised to verify that you have a full backup copy of all Kaspersky MLAD data.

To roll back Kaspersky MLAD to the previous installed version:

1. Go to the directory containing the relevant backup copy of Kaspersky MLAD that the application rollback should restore:
`cd <directory containing the application backup copy>`
2. To roll back the application to the previous version, run the backup script named `backup.sh` with the `-r` switch:
`sudo ./backup.sh -r -f <full path to application directory>`
3. Follow the instructions of the backup wizard.

Kaspersky MLAD will be rolled back to the previous installed version.

Scenario for restoring Kaspersky MLAD from a backup

If necessary, for example, if the server hosting Kaspersky MLAD malfunctions, you can restore the application from a backup copy of Kaspersky MLAD by using the `backup.sh` script.

A user account in the Kaspersky MLAD server operating system must have root access to restore the application.

The scenario for restoring the application from a backup copy consists of the following steps:

1 Moving a backup copy of the application to the Kaspersky MLAD server

Copy the directory containing the backup copy of the application to the server where the application is being restored.

2 Restoring Kaspersky MLAD

Go to the directory containing the backup copy of Kaspersky MLAD by running the following command:

```
cd <directory containing the application backup copy>
```

To restore the application from a backup copy, run the application backup script named `backup.sh` with the `-r` switch:

```
sudo ./backup.sh -r -f <full path to the directory in which you need to restore the application>
```

Follow the instructions of the backup wizard.

Getting started

Before starting to work with Kaspersky MLAD, you must make sure that the following conditions are fulfilled:

1. Descriptions of [tags](#) of received telemetry and assets of the hierarchical structure are prepared as a XLSX file to be imported into Kaspersky MLAD. This file is created by a qualified technical specialist of the Customer, a

Kaspersky expert or a certified integrator.

2. A set of presets has been prepared to monitor data flow and evaluate the performance of Kaspersky MLAD. A description of the presets is supplied in the form of a file in JSON format. This file is created by a qualified technical specialist of the Customer, a Kaspersky expert or a certified integrator.
3. The telemetry data source is enabled and configured to send data to Kaspersky MLAD.
4. The data transfer network is prepared to deliver telemetry data from the data source to the Kaspersky MLAD server, the network equipment is properly configured, and data transfer is allowed.
5. Configuration settings and/or configuration files are prepared for the connector that will be used in Kaspersky MLAD to receive telemetry data or events from external systems. The connector must be configured and activated after Kaspersky MLAD is started.
6. If ML models are provided as part of the *Kaspersky MLAD Model-building and Deployment Service*, the [ML models](#) are created and trained based on historical telemetry data by a Kaspersky expert or a certified integrator. The ML models have been prepared for [import](#) into Kaspersky MLAD as TAR files. The Kaspersky MLAD system administrator has been sent the codes for [activating ML models](#). The ML model activation codes are stored in a secure storage location.

Starting and stopping Kaspersky MLAD

By default, Kaspersky MLAD uses the `systemctl` utility to start or stop the application. If there is an unexpected restart of the server where the application is installed, the `systemctl` utility automatically starts Kaspersky MLAD.

If necessary, you can use scripts to start and stop the application. To do so, you must [switch the application state control mode](#).

We recommend the `systemctl` utility for controlling the application state.

Starting or stopping the application with the `systemctl` utility

The user account must have root access to start or stop the application.

To start the application using the `systemctl` utility:

In the command line, run the following command:

```
sudo systemctl start mlad
```

Kaspersky MLAD will be started.

To stop the application using the `systemctl` utility:

In the command line, run the following command:

```
sudo systemctl stop mlad
```

Kaspersky MLAD will be stopped.

When stopping, the application saves service statuses. When the application starts again, the services will be restored to their previous status.

An error message is displayed if you attempt to run the start and stop scripts in control mode by using the systemctl utility.

Starting or stopping the application with the start and stop scripts

To start or stop the application using the start and stop scripts, first [switch the application state control mode](#).

To start the application:

1. Go to the folder where Kaspersky MLAD is installed (/opt/kaspersky/mlad by default).

2. In the command line, run the following command:

```
./mlad-start.sh
```

Kaspersky MLAD will be started.

To stop the application:

1. Go to the folder where Kaspersky MLAD is installed (/opt/kaspersky/mlad by default).

2. In the command line, run the following command:

```
./mlad-stop.sh
```

Kaspersky MLAD will be stopped.

When stopping, the application saves service statuses. When the application starts again, the services will be restored to their previous status.

If you try to use the systemctl utility in control mode via start and stop scripts, an error message is displayed.

Switching between Kaspersky MLAD state control modes

Kaspersky MLAD supports [application state management](#) in the following ways:

- Using the systemctl utility (by default). If there is an unexpected restart of the server where the application is installed, the utility automatically starts Kaspersky MLAD.

We recommend the systemctl utility for controlling the application state.

- Using start and stop scripts.

If necessary, you can switch between different application state control modes by using the setup.sh script.

A user account in the Kaspersky MLAD server operating system must have root access to switch between modes.

To change the application state control mode:

1. Go to the folder where Kaspersky MLAD is installed (/opt/kaspersky/mlad by default).
2. To switch between application state control modes, run the installation script with an -s switch:

```
sudo ./setup.sh -s
```

Kaspersky MLAD will change the application state control mode. When attempting to run start and stop scripts in application state control mode using the systemctl utility, or when attempting to use the systemctl utility with start and stop scripts while in application state control mode, an error message will be displayed.

Updating Kaspersky MLAD certificates

The following certificates are used in Kaspersky MLAD:

- Certificates for connecting to Kaspersky MLAD using the web interface.
- Certificates for connecting connectors and services.

It is recommended to update certificates in the following cases:

- Current certificates have been compromised.
- Certificates have expired.
- Certificates need to be updated in accordance with the enterprise information security requirements.

Updating a certificate for connecting to Kaspersky MLAD using the web interface

By default, Kaspersky MLAD uses a self-signed certificate that is automatically generated during the application installation to connect to the web interface. When using a self-signed certificate to connect to the Kaspersky MLAD web interface, the browser displays a warning that the security certificate or the established connection is not trusted.

To use trusted certificates to connect to the Kaspersky MLAD web interface, you can replace the self-signed certificate with a certificate received from a recognized certification authority or with a custom certificate that complies with the security standards of your organization.

Kaspersky MLAD store certificates for connecting to the web interface at <installation directory>/ssl/nginx/.

The certificate for connecting to Kaspersky MLAD using the web interface can be updated by a qualified technical specialist of the Customer, a Kaspersky employee or a certified integrator.

To update certificates for connecting to Kaspersky MLAD using the web interface:

1. Obtain a trusted certificate and a key for this certificate to connect to the Kaspersky MLAD web interface.
A certificate must be received for the IP address and domain name of the server on which Kaspersky MLAD is installed.
2. Go to the directory containing the trusted certificate and the key to this certificate.
3. In the command line, run the following commands:

```
sudo chown root:root <new certificate .crt> <new certificate key .key>
sudo chmod 640 <new certificate .crt> <new certificate key .key>
sudo cp <new certificate .crt> <installation directory>/ssl/nginx/mlad_nginx.crt
sudo cp <new certificate key .key> <installation directory>/ssl/nginx/mlad_nginx.key
```

The new certificate and its key are saved at <installation directory>/ssl/nginx/ as mlad_nginx.crt and mlad_nginx.key, respectively.

4. Go to the directory where Kaspersky MLAD is installed, and [restart it](#).

After restarting, Kaspersky MLAD uses the new certificate to connect to the web interface.

Updating a certificate for connecting connectors and services

In Kaspersky MLAD, you can use a secure connection for OPC UA Connector, MQTT Connector, AMQP Connector, HTTP Connector, WebSocket Connector, and the Mail Notifier service. You can update certificates for connecting these connectors and the Mail Notifier service using a secure connection in the **System parameters** section of the [administrator menu](#).

To connect the OPC UA Connector, MQTT Connector, AMQP Connector, HTTP Connector, and WebSocket Connector as well as the Mail Notifier service over a secure connection, it is recommended to use certificates created according to the X.509 standard with a certificate key length of at least 4,096 bits.

The certificate for connecting the KICS Connector is contained in the communication data package, which you can update in Kaspersky Industrial CyberSecurity for Networks. You can upload the updated communication data package to Kaspersky MLAD when [configuring the KICS Connector](#). For detailed information about creating a communication data package, please refer to the *Kaspersky Industrial CyberSecurity for Networks Help Guide*.

Kaspersky Machine Learning for Anomaly Detection is compatible with Kaspersky Industrial CyberSecurity for Networks version 4.0 and later.

First startup of Kaspersky MLAD

This section describes the sequence of application configuration steps that must be performed by the system administrator when Kaspersky MLAD is started for the first time.

The first startup of Kaspersky MLAD consists of the following steps:

1 Starting Kaspersky MLAD

[Start](#) Kaspersky MLAD. The following Kaspersky MLAD prerequisite services will run automatically:

- API Server
- Web Server
- Message Broker
- Keeper
- Time Series Database

- Database
- Logger
- Docker API Server
- Migrations
- Push server

2 Connecting to the Kaspersky MLAD web interface

[Open the application web interface](#) in a [supported browser](#) and enter the user name and password of the first Kaspersky MLAD user with the system administrator role defined during [installation of the application](#). [Change the password for your user account](#). For a secure connection to the Kaspersky MLAD web interface, you are advised to [install a trusted certificate](#).

In the **System parameters** section, in the [administrator menu](#), specify the name of the monitored asset.

3 Uploading a configuration of tags and assets of the hierarchical structure to Kaspersky MLAD

For subsequent operation, [upload tag and asset configuration](#) to Kaspersky MLAD. Tag and asset configuration is described in a XLSX file. For an example of a [tag and asset description](#), see the Appendix.

4 Configuring connectors

To work with data, configure the connectors used at your monitored asset. You can configure the following connectors:

- [KICS Connector](#).
- [OPC UA Connector](#).
- [CEF Connector](#).
- [HTTP Connector](#).
- [MQTT Connector](#).
- [AMQP Connector](#).
- [WebSocket Connector](#).

5 Configuring services

In the **System parameters** section of the [administrator menu](#), configure the [services](#) that you need to use for your monitored asset. In the **Services** section, [check the statuses of the services](#) and [start](#) them, if necessary. For example, the necessary connectors must be running to receive data, and the *Anomaly Detector* service must be running to correctly detect anomalies.

6 Connecting to a data source

When the connectors are configured, [start the connectors](#) used for your monitored asset. Go to the [Dashboard](#) section and make sure that data is being received by Kaspersky MLAD in online mode.

7 Creating user accounts

[Create accounts](#) for users of the application and assign the necessary roles to them. [Create incident notifications](#) for users.

Kaspersky Machine Learning for Anomaly Detection is prepared for operation, and the application is receiving and processing data.

Users can [start working with Kaspersky MLAD](#) using the web interface.

Removing the application

A user account in the Kaspersky MLAD server operating system must have root access to uninstall the application.

Removal of Kaspersky MLAD must be performed by a qualified technical specialist of the Customer, a Kaspersky employee, or a certified integrator.

When Kaspersky MLAD is removed, all Kaspersky MLAD data that was received, uploaded, and processed since the application was installed will be lost. You are advised to verify that you have a full backup copy of all Kaspersky MLAD data. You can perform a backup when [updating the application](#) or with the help of the backup.sh [script](#).

To remove Kaspersky MLAD:

1. Go to the folder where Kaspersky MLAD is installed (/opt/kaspersky/mlad by default).

```
cd mlad-release-5.0.0-< build number >
```

2. Run the setup.sh installation script with the -u switch:

```
sudo ./setup.sh -u
```

3. Follow the instructions of the Application Removal Wizard.

When deletion of the installed certificates is confirmed, the Wizard will delete the directory in which the backup copies are stored.

Kaspersky MLAD will be removed.

Kaspersky MLAD web interface

Kaspersky MLAD is managed through a web interface. This section provides a description of the main elements of the Kaspersky MLAD web interface.



The main window of the application web interface contains the following items:

- Main menu in the left part of the application web interface window.
- Workspace in the central part of the application web interface window.

Sections of the main menu are available to users with access rights to the corresponding [functions of the application](#). Access to application functions is determined by the list of rights assigned to the [user role](#).

System administrators have access to a menu that allows them to [manage license keys](#), [configure application settings](#), [manage user roles](#) and [accounts](#), [configure incident notifications](#), and [manage assets and tags](#).

The user menu, located at the bottom of the main and administrator menus, allows users to: [select the web interface language](#), [change their password](#), [log out](#), [view system logs](#), and navigate between menus. To switch between the main menu and the administrator menu, click one of the following buttons:








-  to go to the administrator menu.
-  to go to the main menu.


If necessary, you can collapse or expand the menu by clicking << or >>, respectively, in the upper-left corner of the page.

Main menu

The table below describes the sections of the main menu of Kaspersky MLAD.

Main menu sections








Section	Description
 Dashboard	Opens the section containing information about the latest registered incidents, application services and their statuses.
 Monitoring	Opens the section that displays data received by the system in real time. You can also configure the settings for displaying incoming data on a graphic areas.
 History	Opens the section that contains a complete history of data received by the system and the results of its analysis by ML models. You can also configure the settings for displaying historical data on graphic areas.
 Time slice	Opens the section containing information about the values of process parameters received from sensors at the same point in time. You can also configure the settings for displaying the data on the graph.
 Event Processor	Opens the section where you can view information about events received from external systems and patterns detected for them, as well as manage monitors to track specific events, patterns, or event parameter values.
 Incidents	Opens the section that contains the log of registered incidents. As part of the incident analysis, you can add a status, reason, expert opinion, and comments to an incident or an incident group.
 Models	Opens the section where you can manage all markups, and ML models, elements and

	templates used in the system. System administrators and users with permissions in the Manage ML models permission group can manage ML models, elements, and templates. All users can manage markups.
☆ Presets	Opens a section where you can manage presets and graphic areas within these.
 Services	Opens the section enabling you to view information about services and their statuses, as well as to start, stop, and restart services. Only system administrators and users with the Manage statuses of application services permission in the Working with application services permissions group can manage service statuses.

Administrator menu

The table below describes the sections of the Kaspersky MLAD administrator menu.







Administrator menu sections


Section	Description
 Licensing	Opens the section in which you can manage license keys.
 Users	Opens the section in which you can manage user accounts.
 Roles	Opens the section in which you can manage user roles.
 Rights	Opens the section containing information about the permissions of users.
 Notifications	Opens the section in which you can manage conditions for sending notifications when an incident is registered.
 System parameters	Opens the section in which you can manage the settings of Kaspersky MLAD.
 Assets	Opens the section in which you can manage assets and tags.

User menu

The table below describes the elements of the Kaspersky MLAD user menu.

User menu elements

Menu element	Description
	Lets you select the localization language for the Kaspersky MLAD web interface. It is available in English and Russian.
	Opens the Kaspersky MLAD Help Guide in a new browser tab.
	Opens the window containing brief information about the application.
	Takes you to the logging system (Grafana ™) in a new browser tab. This section is only available to system administrators and users with the Manage application logs permission.
	Navigates to the administrator menu from the main menu. In the administrator menu, you can manage license keys, Kaspersky MLAD settings, user roles and account credentials, configure incident notifications, and manage assets and tags. The administrator menu is available only to system administrators.
	Navigates to the main menu from the administrator menu. In the main menu, you can manage ML models, elements, markups, and templates, manage service statuses, and view historical and real-

	time data, incidents, events and patterns.
	Allows you to change the password for the current user account and log out of the account.

Connecting to Kaspersky MLAD and terminating a user session

Use a [supported browser](#) to connect to Kaspersky MLAD web interface.

If an authorized user does not use the application for a time period exceeding the [User inactivity period \(min\)](#), Kaspersky MLAD automatically terminates the connection session for this user. To continue working in the application, user authorization must be completed again. A user session is considered active in the following cases:

- The user interacts with elements of the application interface (for example, clicks buttons or navigates to sections of the application menu).
- The user enters parameter values using the keyboard.

Performing any of the above actions prolongs the user session for the time specified in the **User inactivity period (min)** parameter.

The user can terminate their user session ahead of time by [logging out of their user account](#).

If necessary, the system administrator can [revoke authentication tokens](#) for a user account. When a token is revoked for a user, their work sessions in the application are terminated simultaneously on all devices where they are authorized.


The web address, user name and password for signing in to the application must be requested from the Kaspersky MLAD system administrator.

Connecting to the web interface

To connect to Kaspersky MLAD using a browser:

1. Open a [supported browser](#) on your computer.
2. In the browser address bar, enter the Kaspersky MLAD server web address received from the Kaspersky MLAD system administrator.
3. On the login page that opens, enter your email as the user name and your password.
When connecting to the web interface as the system administrator for the first time, use the user name and password of the first user with the system administrator role that were specified during [installation of the application](#).
4. Click the **Sign in** button or press **ENTER**.

The [Dashboard](#) is displayed in the browser window.

When a user connects to Kaspersky MLAD for the first time, the [password change window](#)  opens in the browser. If the password change was made optional in the [security settings](#), you can skip changing the password by clicking the **Skip** button and [change it later](#). The change password window also opens in your browser upon expiration of the password that was set when [configuring the security settings](#).

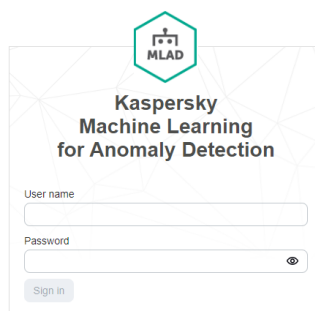
- **New password** – new password for the user account.

The new password must meet the following requirements:

- Must not match previously used passwords. The specific number of most recently used passwords that must not be reused is defined by the administrator when [configuring the security settings](#).
- Must contain the minimum number of characters defined by the administrator when [configuring the security settings](#).
- Must contain letters of the English alphabet, numerals and/or special characters in accordance with the password policy that was set by the administrator when [configuring the security settings](#).
- **Confirm password** means you must type the password again to confirm the password for the user account.

If you close the browser window without [terminating the connection session](#), the session remains active until the time limit that was set by the administrator in the **User inactivity period (min)** parameter when [configuring the security settings](#). During this time, the application continues to grant access to the Kaspersky MLAD web interface without prompting for user account credentials, provided that the connection is used by the same computer, browser, and operating system user account. If the application user is inactive for longer than the time limit that was specified in the [User inactivity period \(min\)](#) parameter, Kaspersky MLAD terminates the user session.

In case of multiple unsuccessful authorization attempts, Kaspersky MLAD will block your account when the maximum number of unsuccessful authorization attempts is reached for a certain period. The maximum number of unsuccessful authorization attempts and the account blocking period are set when [configuring the security settings of Kaspersky MLAD](#).



Terminating a Kaspersky MLAD connection session

When you are done working with Kaspersky MLAD in a browser, you must terminate the connection session.

To terminate the application connection session:

in the browser window, in the lower-left corner of the Kaspersky MLAD web interface, click the button  and select **Sign out of Kaspersky MLAD**.


After the application connection session is terminated, the browser window shows the page for entering account credentials.

Changing a user account password

You are advised to change the password in the following cases:

- You are connecting to Kaspersky MLAD for the first time after the user account was created in the application.
- The current password has been compromised.
- The password is expiring in accordance with the information security requirements at the enterprise.

To change the password of your own user account:

1. In the lower-left corner of the Kaspersky MLAD web interface page, click the button  and select **Change password**.

The **Changing password** window opens in the browser.

2. In the **Current password** field, enter your current password.

3. In the **New password** and **Confirm password** fields, enter the new password.

The new password must meet the following requirements:

- Must not match previously used passwords. The specific number of most recently used passwords that must not be reused is defined when [configuring the security settings](#).
- Must contain the minimum number of characters defined when [configuring the security settings](#).
- Must contain letters of the English alphabet, numerals and/or special characters in accordance with the password policy that was set when [configuring the security settings](#).

4. Click the **Edit** button.

Selecting the localization language for the Kaspersky MLAD web interface

Kaspersky MLAD provides the option to use English or Russian for the application web interface.

To change the localization language of the application web interface:

1. In the lower-left corner of the Kaspersky MLAD web interface page, click the **Language** button.
2. Select the required localization language: Russian or English.

Licensing the application

This section provides information about general concepts related to licensing of Kaspersky MLAD.

About the End User License Agreement

The *End User License Agreement* (EULA) is a binding agreement between you and AO Kaspersky Lab that stipulates the terms on which you may use the application.

Please carefully read the terms of the End User License Agreement before using the application.

You can review the terms of the End User License Agreement in the following ways:

- During installation and update of Kaspersky MLAD.
- By reading the `license_en.txt` file. This file is included in the application distribution kit.

You accept the terms of the End User License Agreement when you confirm your consent to the End User License Agreement during the installation or update of the application. If you do not accept the terms of the End User License Agreement, please stop the installation or update of the application and do not use the application.

About the license

A *license* is a time-limited right to use the application as granted under the End User License Agreement.

The available functionality and application usage period depend on the type of license under which the application is being used.

The following types of licenses are available:

- *Basic* is a free license.

This type of license does not have a time limit. You can use the [basic functions of the application](#) before adding a license key or after expiration of the commercial license.

- *Trial* is a free evaluation license.

When the trial license expires, the application continues to operate but the following application functions become unavailable:

- Update the application.
- Select an ML model element.
- Work with markups, ML models, and ML model templates.
- Work with events and patterns.
- Work with incidents and groups of incidents.

- Manage the statuses of the Anomaly Detector, Trainer, Similar Anomaly, Event Processor, and Stream Processor services.

You will need to purchase a commercial license to continue using the application.

A trial license has a short validity period. You can use the application under a trial license only for one trial usage period.

If you add a [commercial license key without the model builder](#) after the trial license expires, the ML models and ML model elements that were created manually during the trial period will become unavailable for editing, cloning, and deletion. When you add a license key for a valid [commercial license that includes the model builder](#), you can continue managing the ML models and ML model elements that you created manually during the trial period.

- A *commercial* license is a paid license.

This type of license has a time limit.

To activate application functionality, you need to [add a license key](#) for a valid commercial license. The following types of commercial license are available for Kaspersky MLAD:

- Commercial license without the model builder.

If you require all application functions except for [managing manually created ML models](#), [cloning](#) and [deleting ML model elements](#), you will need to purchase a commercial license that does not include the model builder.

- Commercial license including the model builder.

If you need all application functions, you will need to purchase a commercial license that includes the model builder.

After the commercial license expires, only the [basic functions of the application](#) will remain available. To continue using Kaspersky MLAD, you need to renew the commercial license.

You are advised to renew the license no later than its expiration date to ensure continuous use of Kaspersky MLAD.

Technical support services are provided if you have an active *Technical Support Agreement*. The scope of provided technical support services is determined by the current *Technical Support Agreement*.

About the license certificate

The *license certificate* is a document that is handed over to you along with the key file.

The license certificate contains the following information about the license provided:

- License key or order number
- Information about the user who is granted the license
- Information about the application that can be activated under the provided license
- Limitation on the number of licensing units (for example, tags by which the application can receive telemetry data)
- Start date of the license term
- License expiration date or license term

- License type

About the license key

A *license key* is a sequence of bits that lets you activate and start to use the application according to the terms of the License Agreement. The license key is generated by Kaspersky experts.

You can add a license key to the application by using a *license key file*. After you add a license key to the application, the license key is displayed in the application interface as a unique alphanumeric sequence.

Kaspersky can block a license key over violations of the End User License Agreement. If the license key is blocked, you need to add another license key to work with the application.

A license key may be active or reserve.

An *active license key* is the one currently in use. A license key can be added as active to a trial or commercial license. The application cannot have more than one active license key.

A *reserve license key* is a license key that confirms the right to use the application but is not currently in use. A reserve license key is activated automatically when the license associated with the currently active license key expires or the active license key is removed. A reserve license key can only be added if there is an active license key.

A license key for a trial license can only be added as the active license key. A license key for a trial license cannot be added as a reserve license key.

About the license key file

A *license key file* is a file with the KEY extension that you receive from Kaspersky. The key file is used to add a license key that activates the application.

You receive a license key with your Kaspersky MLAD purchase or trial. The method used to receive a license key file is determined by the Kaspersky distributor from whom you purchased the application (for example, the license key file may be sent to the email address you specify).

Alternatively, you can add a license key you received when purchasing the previous version of Kaspersky MLAD. A license key can be added to the application before its expiration date.


No connection to Kaspersky activation servers is required to activate the application with a key file.

You can restore a key file if it has been accidentally lost. Contact the party you bought the license from to restore the key.

Available functionality of Kaspersky MLAD depending on the specific license

The table below shows the set of available functions of Kaspersky MLAD depending on the selected license.

Functions	Basic license	Trial license	Commercial license without the model builder	Commercial license including the model builder

Basic functions  <div> <p>The following features are available:</p> <ul style="list-style-type: none"> • Installing, removing, updating, backing up, and rolling back • Connecting to, and disconnecting from, the web interface • Selecting a Kaspersky MLAD web interface language • Viewing the Kaspersky MLAD help • Viewing the application version • Viewing Kaspersky MLAD logs • Changing your own password • Managing user accounts • Managing roles • Managing incident notifications • Configuring Kaspersky MLAD • Managing assets and tags • Viewing data under Monitoring, History, and Time slice, except when selecting an ML model element • Managing presets • Viewing the statuses of services • Managing service status, except for Anomaly Detector, Trainer, Similar Anomaly, Event Processor, and Stream Processor </div>	✓	✓	✓	✓
Selecting an ML model element under Monitoring , History , and Time slice	—	✓	✓	✓
Working with events and patterns	—	✓	✓	✓
Working with incidents and groups of incidents	—	✓	✓	✓
Working with markups	—	✓	✓	✓
Working with imported ML models	—	✓	✓	✓
Working with manually created ML models	—	✓	—	✓
Cloning an ML model	—	✓	✓	✓

Cloning of the ML model element	—	✓	—	✓
Removing an ML model	—	✓	✓	✓
Removing an ML model element	—	✓	—	✓
Working with ML model templates	—	✓	✓	✓
Training of predictive elements and elliptic envelope-based elements	—	✓	✓	✓
Viewing the training results of ML model training	—	✓	✓	✓
Preparing an ML model for publication	—	✓	✓	✓
Publishing an ML model	—	✓	✓	✓
Starting and stopping ML model inference	—	✓	✓	✓
Managing the status of Anomaly Detector, Trainer, Similar Anomaly, Event Processor, and Stream Processor	—	✓	✓	✓

Adding a license key

You can add a [license key](#) to Kaspersky MLAD when connecting to the application via the web interface.

Only a system administrator can add a license key.

To add a license key:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select **Licensing**.

3. Click the **Add license key** button.

The button is unavailable if the active and reserve license keys were previously added. If necessary, you can [delete a license key](#).

The **Adding a license key** panel appears on the right.

4. Select a license key file in KEY format.

After you select the license key file, the **details of the added key** **Adding a license key** Adding license key panel.

If the application did not have any license keys and the validity period of the new license key has already started, it will be added as the active key automatically. If the validity period of the new license key has not yet begun and it does not overlap with the validity period of your current active license key, the new key will be added as a reserve key.

5. If a license key was previously added to the application, do one of the following (if necessary):

- To download a key as active when the application already has an active license key, select **Active license key** under **License key status**.

The previous active license key will be replaced and removed.

- To download a key as an active key when its validity period overlaps with the validity period of a previously uploaded reserve key, select **Active license key** under **License key status**, and select one of the following options for using the reserve key under **Apply reserve key**:
 - **When active key expires.** The reserve license key will become active after the active license key expires.
 - **When reserve key becomes valid.** The reserve key will become active at the beginning of its validity period. Previous active license key will be removed from the application. You can use the removed license key on another monitored asset until this key expires.
- To upload a key as a reserve key when its validity period overlaps with the validity period of a previously uploaded active key, select **Reserve license key** under **License key status**, and select one of the following options for using the reserve key under **Apply reserve key**:
 - **When active key expires.** The reserve license key will become active after the currently active license key expires.
 - **When reserve key becomes valid.** The reserve key will become active at the beginning of its validity period. Previous active license key will be removed from the application.

You cannot add a license key as a reserve if it is set to expire before the currently active license key.

6. Click the **Add** button.

The license key will be loaded.

Viewing information about an added license key

You can view information about the added license key when connected to Kaspersky MLAD. 30 days before the license key expires, the application web interface will display a notification informing you about the number of days remaining until the key expiration date.

To view information about the license key:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select **Licensing**.

The following information is displayed for an added license key:

- Days remaining until license expiration.
- **Application name**—the application that will be activated after adding the license key;
- **License key status** – key status: active or reserve.
- **License number** – order ID.
- **Key** – unique alphanumeric sequence.
- **Company name**—the name of the company for which the license was purchased;

- **Activation date** – date when the license key was first added to the application.
- **Key is valid until** – license key expiration date.
- **Functionality**—information about the number of available tags, as well as the ability to use ML models and/or manage ML models and their elements.

Removing a license key

When connected to the application through the web interface, you can remove an added license key from the application (for example, if you need to replace this license key with a different key). After removing the license key, only the [basic functions of the application](#) will be available to you.

After removing the license key, you will continue to receive data on tags using the connectors, and you will also be able to view data on tags in the **Monitoring** and **History** sections. In turn, ML models will stop processing data on tags, and will stop generating incidents and [artifacts](#). You will no longer be able to view previously generated artifacts. You will also not be able to use previously created ML models.



Unavailable functionality will be re-activated the next time you add a license key.

We recommend [adding a reserve license key to the application](#) before removing the active license key. When removing the active license key, the previously added reserve license key will be activated. The license period will be extended to match the validity period of the added license key.

If you have removed a license key before it was due to expire, you can [add it to the application again](#) and retain its original license expiration date.

Only a system administrator can remove a license key.

To remove a license key:

1. In the lower-left corner of the window, click .
- You will be taken to the [administrator menu](#).
2. Select **Licensing**.
3. In the upper right corner of the license key card, click .
- A window with a confirmation prompt opens.
4. Confirm deletion of the license key.

The license key will be removed from the application.

Processing and storing data in Kaspersky MLAD

This section contains information about data provision and directories for storing data.

About data provision

The application does not transfer users' personal data to Kaspersky. Users' personal data is processed locally on the computers where the application is installed.

Data transferred to external systems

Data is transmitted to external systems over encrypted communication channels.

If [sending email notifications about incident logging is enabled](#), the application transfers the following data to the SMTP server:

- [Incident](#) ID
- Date and time of the logging of the incident
- [ML model](#) name whose element registered the incident
- Name of the ML model element that registered the incident
- Registered incident status
- the reason for the registered incident
- Expert opinion on the registered incident
- [Top tag](#) name
- Top tag description
- Value of the top tag at the time the incident is registered and its measurement unit
- Name of the [detector](#) that logged the incident
- Value of the ML model [artifact](#) at the time the incident is registered
- Blocking threshold value exceeded at the time the incident was registered
- Link to the **History** section at the time of the start of the incident.

If [sending notifications about incident logging through the MQTT Connector](#), [AMQP Connector](#), [WebSocket Connector](#), and/or [KICS Connector](#) is enabled, the application transfers the following data to the MQTT broker, AMQP broker, WebSocket server, and/or to Kaspersky Industrial CyberSecurity for Networks:

- Incident ID

- Date and time of the logging of the incident
- Date and time of the incident completion
- Name and unique ID (UUID) of the ML model that logged the incident
- Unique ID (UUID) of the ML model element
- Top tag ID and description
- Name of the detector that logged the incident
- Link to the **History** section at the time of the start of the incident
- Value of the ML model element artifact at the time the incident is registered (if any)
- Blocking threshold value exceeded at the time the incident is logged (if any)
- Top tag value at the time the incident is logged
- Incident status
- Incident comment (if any)
- Incident group ID (if any)
- Incident group name (if any)
- Expert opinion (if any)
- IDs of the [relevant tags](#)
- Reason for the incident (if any)

If [notifications about the logged incidents are configured to be sent via the CEF connector](#), the application transfers the following data to the SIEM system:

- Application vendor name
- Application name
- Kaspersky MLAD version
- Application signature ID
- Date and time of the logging of the incident
- Date and time of the incident completion
- Name of the detector that logged the incident
- Name of the ML model that logged the incident
- Link to the **History** section at the time of the start of the incident
- Top tag description

- Incident comment (if any)
- Incident group name (if any)
- Top tag value at the time the incident is logged
- Incident group ID (if any)
- Incident ID
- Top tag ID.

If the logged [events](#) are configured to be sent via the CEF [connector](#), the application transfers the following data to the SIEM system:

- Application vendor name
- Application name
- Kaspersky MLAD version
- Application signature ID
- Date and time when the event was logged
- Name of the [monitor](#) that logged the event
- Monitor ID
- Type of the element that caused the monitor activation
- Name of the [attention head](#) (if any)
- Name of the daughter attention head (if any)
- Sliding window of the monitor used to track the number of activations
- Threshold of activations that, when reached, cause the monitor to send an alert to an external system
- Number of activations on the sliding window
- Indication of whether the detected element is new to the application
- Contents of the element that triggered the last monitor activation
- ID of the element that triggered the monitor activation

If sending information security event logs is enabled, the application transfers the following data to the Syslog server:

- Application vendor name
- Application name
- Kaspersky MLAD version

- Application signature ID
- ID of the information security event
- Date and time when the information security event occurred
- Information security event type
- Information security event subtype
- Information security event severity level
- Name of the user whose actions resulted in the information security event entry
- IP address of the computer from which the user performed the actions logged into the information security event log
- Information security event outcome
- Brief summary of the information security event
- Detailed description of the information security event.

Data processed locally on the Kaspersky MLAD server

To perform its [main functions](#) the application can receive, store and process the following information:

- Information about the full backup copies of the application, if the application has been [backed up](#) or [updated](#). The Kaspersky MLAD server stores information about full application backups until they are deleted by the user.
- Information about the backup copies of the Docker volumes that are created during uninstallation of the application. The Kaspersky MLAD server stores information about Docker volume backups until they are deleted by the user.
- Files containing the text of the End User License Agreement of the currently installed application version.
- File named legal_notices.txt containing information about third-party code.
- Certificates for connecting to the application using the web interface.
- Certificates and certificate keys for encrypting the connection between Kaspersky MLAD connectors and services and the external systems.
- Public keys for verifying the digital signature of the distribution package. The Kaspersky MLAD server stores public keys until they are deleted by the user.
- User account data: account ID, last name, first name, middle name, email address, account status (active or blocked), password.

Values that do not personally identify the user (for example, shop and job title) can be entered in place of the last name, first name and middle name of a user. The information specified in the **Last name**, **Name** and **Middle name** fields for users when creating user accounts is stored in plain text and is not processed by the application.

The email addresses that are specified when creating accounts are used for the user names when users connect to the web interface of the application. User names are indicated in the [information security event logs](#). Email addresses are used to [send notifications about registered incidents](#).

Users' email addresses are stored in plain text.

Kaspersky MLAD does not store user passwords in plain text. The script hash sum calculation algorithm is used to store passwords. Kaspersky MLAD adds salt to the password to prevent decoding. User passwords are not written to application logs.

The system administrator enters information about user accounts in the administrator menu.

- Data about roles and the rights assigned to these roles: role ID, role name, role status (active or inactive), list of assigned rights, date and time of role creation, date and time of role modification.

The system administrator enters information about roles in the administrator menu.

- Details of incident [notifications](#): notification ID, notification name, whether incident notifications are enabled for published ML models only, notification state (active or inactive), notification language, email addresses to notify, types of incidents to send notifications for.

The system administrator enters information about notifications in the administrator menu.

- Information about the license keys uploaded to Kaspersky MLAD.
- Data about Kaspersky MLAD settings:
 - Main application settings: monitored asset name, URLs and IP addresses for generating links to incidents in incident notifications, interval for receiving data from the [Message Broker service](#), interval for receiving statistical data about incidents from the database, and the monitored asset time zone.
 - Application security settings: number of authorization attempts, user blocking period, user inactivity period, information on whether the password must be changed upon the first connection, number of user passwords stored in the history, password validity period, minimum password length, information on whether uppercase, lowercase Latin letters, numbers and/or special characters (`_!@#$$%^&*`) must be used in the password, size and storage time of the information security event logs.
 - [Anomaly Detector](#) settings: information on whether to use the Limit Detector, Forecaster, XGBoost, and/or Rule Detector, information on whether to skip data gaps, the maximum number of records requested from the Message Broker service, the number of messages sent in one block to the Message Broker.
 - [Keeper service](#) settings: information indicating whether the values of all [tags](#) must be stored, and the timeout for receiving tag values, incidents, and metrics.
 - [Mail Notifier service](#) settings: notification sender email address, SMTP server address and port, user name and password for connecting to the SMTP server, information indicating whether to use a TLS connection, SMTP server certificate and certificate key.
 - [Similar Anomaly](#) settings: minimum and maximum number of incidents for the group, maximum interval between similar incidents.
 - [Stream Processor](#) settings: uniform grid step, configuration file with Stream Processor settings.

The Stream Processor configuration file stores the IDs of the tags processed by the service and the values of tag processing settings.

The values of the tag processing settings are set by Kaspersky experts individually for each monitored asset.

- [HTTP Connector](#) settings: size of the written block, maximum size of the uploaded file, information indicating whether TLS connection and the recommended TLS connection settings must be used, HTTPS server certificate and certificate key, root certificate to verify the signature of the client certificate, and information indicating whether the received tag values need to be scaled.
- [MQTT Connector](#) settings: information indicating whether to use a TLS connection and the recommended TLS connection settings, address and port of the MQTT broker, user name and password to connect to the MQTT broker, root certificate, client certificate and client certificate key, list of MQTT subscriptions to receive tags, [MQTT topic](#) for publishing messages, format for processing incoming data, connector configuration file, and information indicating whether to scale the received tag values.
The MQTT Connector configuration file stores IDs, names, descriptions, types, and measurement units for tags.
- [AMQP Connector](#) settings: information indicating whether to use a TLS connection and the recommended TLS connection settings, address and port of the AMQP broker, user name and password to connect to the AMQP broker, root certificate, client certificate and client certificate key, AMQP virtual node, names of AMQP exchange points for receiving tag values and publishing messages, list of subscriptions and the queue for receiving tag values, [AMQP topic](#) for publishing messages, format for processing incoming data, connector configuration file, and information indicating whether to scale the received tag values.
The AMQP Connector configuration file stores IDs, names, descriptions, types, and measurement units for tags.
- [OPC UA Connector](#) settings: connection address, timeout for connection to the OPC UA server, connector configuration file, information indicating whether the received tag values need to be scaled, connection security policy, message security mode, user name and password for connecting to the server, client application certificate, private key of the client application certificate, password for the private key of the client application certificate, root certificate, historical data interval, beginning and end of the historical data period, size of the historical data block sent by the OPC UA server, and size of the historical data block sent to the Message Broker service.
- [KICS Connector](#) settings: communication data package for the KICS Connector, password for the KICS Connector, information on whether to send messages to Kaspersky Industrial CyberSecurity for Networks, the tag [sampling](#) frequency, information on whether to scale the received tag values.
- [CEF Connector](#) settings: information indicating whether to receive events for the Event Processor service, information indicating whether to send registered incidents and/or events to a SIEM system, IP address and port for sending events and incidents to SIEM systems, information indicating whether to send the information security event logs to the Syslog server, transport protocol for sending information security events to the Syslog server, address and port of the Syslog server for sending information security events, information indicating whether to use a TLS connection and the recommended TLS connection settings, server certificate and certificate key, root certificate for verifying the signature of the client certificate, client certificate and certificate key, and root certificate for verifying the signature of the server certificate.
- [WebSocket Connector](#) settings: WebSocket server URL address, root certificate, client application certificate and client application certificate key, incoming data processing format, connector configuration file, information indicating whether to scale the received tag values, information indicating whether to send incidents, and information indicating whether to use the recommended TLS connection settings.
- [Event Processor](#) settings: service configuration file, information on whether to process incidents as events, the maximum number of network layers, the coefficient defining the permitted dispersion of the pattern duration, the interval for receiving epoch events, the epoch size in online mode, the mechanism for saving the Event Processor status, component backup frequency, the backup copy of the Event Processor status, epoch size in sleep mode, alert mode when the monitor is activated in sleep mode, sleep mode frequency and duration, event history interval for processing in sleep mode.
- Incident status settings: incident status ID, incident status names in Russian and English, sorting sequence number, information on whether to display the registered incidents with this status.

- Incident cause settings: incident cause ID, incident cause name, sorting sequence number.
- Logging service settings: logging levels of the services and application connectors.
- Settings of the time intervals for charts in the **Monitoring**, **History**, and **Time slice** sections: time interval ID, time interval name in Russian and English, sorting sequence number, ID of the user who created the time interval, ID of the user who last changed the time interval, time interval value.
- Settings for displaying the items of the main menu and administrator menu: information on whether to display the items of the main menu and administrator menu in the application web interface.

The system administrator defines Kaspersky MLAD settings in the administrator menu.

- [Asset](#) and tag data: asset name, asset ID, asset icon, parent asset ID, asset description and type, asset type ID and name, names and values of special parameters of the asset type, asset type description, tag ID and name, tag alternative name, tag icon, tag description, tag type, tag unit of measurement, upper and lower thresholds for blocking, alarms and measurement reliability, comment to the tag, spatial location coordinates of the monitored asset sensor in space along the abscissa, ordinate and applicate axes, name of the device from which the tags from the external system originated, and the offset and multiplier parameters that are used to recalculate the tag values received from the connectors.

The system administrator enters information about assets and tags in the administrator menu.

- [Preset](#) data: preset name, preset ID, preset icon, names and IDs of tags included in the preset, name and description of the [graphic area](#), axis scaling mode, upper and lower bounds for displaying tag values, additional threshold lines, information about tags included in the graphic area, information indicating whether you need to customize the expression for the **Time slice**, labels of the abscissa and ordinate axes, name of the expression for calculating tag values, expressions for calculating tag values, and the color of the graph for the preset in the **Time slice**.

Any user can enter data in the **Presets** section.

- Information about the number of tag observations and events received per second. The application calculates the data based on the data received from external systems.
- Information about the values of tags and events received by the system. Data is received from external systems for which data receipt is configured.
- Information about the generated [artifacts](#). The application calculates the data based on the data received from external systems.
- Information about the application service statuses: the name and current status of the service. The application displays the service status derived from the corresponding components.
- Data on registered incidents and groups of incidents: incident ID, date and time when the incident was registered, top tag name and ID, incident cause, name of the detector that registered the incident, incident group name, incident status, ML model name, ML model element, ML model element artifact value, threshold value, top tag value, blocking thresholds, tag description and measurement units, incident type, date and time when the observation was generated, amount of time by which observation generation is ahead or behind the receipt of this observation by the application, expert opinion on the incident and on the group, incident comment, incident group name and ID, number of incidents in the group, date and time when the incident group was created, status of the registered incidents in the group, IDs of the relevant tags, and the blocking threshold reached when the incident was registered.

The application generates this data as a result of analysis of the received data and on the basis of the settings specified by the user.

- Settings for displaying charts in the **Monitoring** and **History** sections: chart height, preset for going to the **History** section (only when configuring the chart display settings in the **Monitoring** section), information on whether to display the observation chart with the selected color, the observation chart color, information on

whether to display the prediction chart with the selected color, prediction chart color, information on whether to display the names and descriptions of tags on the charts, the predicted value of the tag and/or an individual tag error, information on whether to display indicators for all incidents on the charts, information on whether to display blocking thresholds and/or additional threshold lines on the charts, ML model element used to generate predicted values, presets, time intervals, date and time for displaying charts.

Any user can enter data in the **Monitoring** and **History** sections.

- Chart display settings in the **Time slice** section: chart height, ML model element used to generate predicted values, presets, time intervals, date and time for displaying charts.

Any user can enter data in the **Time slice** section.

- Settings for processing and displaying data for the Event Processor: ID, name and state of the [attention head](#), attention subject parameters (individual for each monitored asset), information indicating whether to generalize the parameters of conditions, and the parameters of attention head conditions (individual for each monitored asset).

If the **Process incidents as events** option is enabled in the [Event Processor settings](#), the application stores and processes the following data:

- Name of the detector
- Name of the ML model being used
- Top tag name and ID
- Name of the incident group to which the registered incident belongs
- Top tag value
- Incident ID.

Any user can enter the event processor data in the **Event Processor** section.

- Data on monitoring events and patterns in the Event Processor: monitor name and ID, monitor state, number of registered activations on the sliding window, date and time of the last activation, activation stack limit, parameter that determines what is tracked by the monitor, sliding window, activation threshold, attention head, attention subject parameter, information indicating whether the monitor tracks events or patterns for a generalized attention subject, activation type, names of event parameters whose values are tracked by the monitor, types of filters, types of values tracked by the monitor, values of event parameters tracked by the monitor, ID of the event parameter value, event or pattern whose detection triggered activation of the monitor, date and time of event detection in the event stream, time interval between the current event and the previous event in the event stream on the sliding window, number of event repetitions in the event stream on the sliding window, number of event parameters whose values were received from the monitored asset, date and time of the last event detection in the stream of events on the sliding window, attention subject parameter and its value that triggered activation of the monitor, date and time of monitor activation, parameters of the event received from the monitored asset, number of events included in the pattern that triggered monitor activation, total number of pattern repetitions in the stream of events, generalized event ID, generalized pattern ID, number of monitor activations by the generalized event or pattern, number of events in the generalized pattern, number of values of the attention subject parameter whose detection triggered activation of the monitor, time interval between the first and the last event in the detected generalized pattern, detected generalized event, detected generalized pattern, and the values of attention subject parameters whose detection triggered activation of the monitor.

The application generates data by analyzing the received data and the settings specified in the **Event Processor** section.

- Data on registered patterns in the Event Processor: pattern ID, date and time of the last pattern detection in the interval, number of pattern detections in the event stream of the monitored asset for the given period, number of events in the pattern, date and time of the last pattern detection in the event stream or in sleep

mode, date and time of the beginning and end of the pattern loading period, type of pattern, attention head, values of the event parameters for which the patterns are registered, template parameters for which the patterns are registered (individual for each monitored asset), ID of the subpattern, end date and time of the subpattern in the sequence of patterns, number of detections of the subpattern, number of events in the subpattern, time interval between the subpattern and the pattern detected in the sequence of patterns in the current layer before the subpattern, date and time of the last detection of the subpattern in the sequence of patterns in the current layer, IDs of events included in the pattern, date and time of event detection in the pattern structure, time interval between the selected event and the previous event, number of event repetitions in the structure of the selected pattern, number of event parameters whose values were received from the monitored asset, and the date and time of the last event detection in the event stream.

The application generates data by analyzing the data and the settings specified in the **Event Processor** section.

- Information about ML models and their parameters: ID and unique ID (UUID) of the ML model, name, description, status and state of the ML model, name of the user who last modified the ML model, date and time when the ML model was last modified, name of the user who created the ML model, date and time when the ML model was created or loaded, the names and IDs of its elements, the time interval, and [markup](#) for the [inference](#).

A system administrator or a user with the [Manage ML models](#) rights set in the **Models** section can enter and/or upload the data.

- Information about the ML model elements and their parameters:
 - Parameters common for all types of ML model elements: ID, name and description of the ML model element, status and state of the ML model element, time interval after which a repeated incident is generated, time interval during which repeated incidents are not registered, anomaly observation interval, anomaly duration share in the interval, incident cause and status, color of the incident indicator points, and expert opinion.
 - Main parameters of predictive ML model elements: element architecture, grid step in seconds, names and IDs of input tags, names and IDs of output tags, incident registration threshold, cumulative prediction error power, cumulative prediction error smoothing factor, number of steps in the input window for the input values, number of steps by which the beginning of the output window is shifted relative to the beginning of the input window, and the number of steps in the output window.
 - Parameters of a neural network element with a [Dense architecture](#): multipliers for calculating the number of neurons on layers, activation on layers, and the regularization coefficient to prevent overfitting of the ML model element.
 - Parameters of a neural network element with an [RNN architecture](#): number of [GRU](#) neurons on layers, number of neurons distributed over time on the layers of the decoder, information indicating whether to restore the data received as input to the network, and the regularization factor to prevent overfitting of the ML model element.
 - Parameters of a neural network element with a [CNN architecture](#): size of filters on the layers, number of filters on the layers, regularization factor to prevent overfitting of the ML model element, size of the maximum sampling window, number of neurons on the layers of the decoder, information whether it is necessary to restore the data received as network input.
 - Parameters of a neural network element with a [TCN architecture](#): regularization factor to prevent overfitting of the ML model element, filter size, number of layers in the residual block, number of filters on layers, extensions on layers, type of layer before the output, packet size, and the activation function.
 - Parameters of a neural network element with a [Transformer architecture](#): regularization factor in the encoder, number of attention heads, number of coding blocks, and multipliers for calculating the number of neurons on layers of the decoder.
 - Training settings of a predictive element: training time interval, names and IDs of the training markups, maximum training duration, ratio between the training and the validation sample, maximum number of epochs for training, number of epochs during which there must be no validation losses when training is

stopped early, chart resolution to display the training results, size of the dataset for training, number of blocks, inference mode, training mode, automatic data division into blocks, memory size used for training, information indicating whether to initialize the model weights with values from the previous training results and/or shuffle the data, value for pseudo-random number generator initialization, learning rate coefficient, training optimization algorithm, and the loss optimization algorithm.

- Information about the training results of the predictive element: training queue (IDs and names of ML model elements that are waiting in the queue for training), training status, names and IDs of the elements being trained, number of blocks into which the training data is divided, name of the user who started the training of the element, training duration, date and time of the training beginning and end, duration of the data time intervals in the training set, number of [UTG](#) nodes included in the training set, training and validation errors, and the prediction made by the trained ML model on the training set.
- Settings for elements based on diagnostic rules: information indicating whether to interpret the impossibility of evaluating a condition as rule fulfillment, time filtering settings: interval type, years, days, days of the week, and the time interval during which to validate the input data in accordance with the specified rule; tag behavior condition settings: tag for which the condition is added, tag behavior, rule fulfillment condition, number of UTG steps, tag threshold value, minimum number of times a rule is triggered before logging an incident, trend slope value, time interval between adjacent trend estimates, change threshold value, direction of the tag value change, tag value, maximum tag deviation from the specified value, direction of change in the tag value spread, indicator of whether the rule uses a pause and the pause settings: minimum and maximum timeouts, and the utilized group and logical operators.
- Parameters of elements based on an elliptic envelope: incident registration threshold, grid step in seconds, and the names and IDs of input tags that must be included in the ML model.
- Parameters for training an element based on an elliptic envelope: time interval for training, names and IDs of markups for training, sample fraction for estimating the mean and covariance, fraction of outliers in the sample, value for initializing the pseudo-random number generator, resolution of graphs for displaying training results, and information indicating whether to assume that the tag values are centered.
- Information about the results of training an element based on an elliptic envelope: training queue (IDs and names of ML model elements that are waiting in the queue for training), training status, name and IDs of the elements being trained, name of the user who started the training of the element, training duration, date and time of the training beginning and end, duration of the data time intervals in the training set, number of [UTG](#) nodes included in the training set, tag deviation, tag values, tag value distribution and tag correlation.

A system administrator or a user with the [Manage ML models](#) rights set in the **Models** section can enter and/or upload the data.

- Information about markups: ID, name and description of the markup, interval used to calculate data on UTG, markup color, method used for the markup to appear in the application, information indicating whether the markup is used as the main inference indicator, time filtering settings: interval type, years, days, days of the week, and the time interval during which the input data should be validated in accordance with the specified markup conditions; tag behavior condition settings: tag for which the condition is added, tag behavior, rule fulfillment condition, number of UTG steps, tag threshold value, minimum number of times a rule is triggered before logging an incident, trend slope value, time interval between adjacent trend estimates, change threshold value, direction of the tag value change, tag value, maximum tag deviation from the specified value, direction of change in tag value spread, indicator of whether the rule uses a pause and the pause settings: minimum and maximum timeout intervals, and the utilized group and logical operators.

A system administrator or a user with the [Manage ML models](#) rights set in the **Models** section can enter and/or upload the data.

- [Information security event logs](#): information security event ID, date and time of the information security event, type of information security event, subtype of information security event, severity level of the information security event, the name of the user whose actions resulted in registration of the information security event, the IP address of the computer from which the user performed the actions logged into the information security

event log, the result of the information security event, a brief summary of the information security event, a detailed description of the information security event.

The IP addresses of computers that established a connection to the web interface of the application are indicated in the information security event logs.

The data is generated by Kaspersky MLAD automatically.

Kaspersky MLAD stores information security event logs for the time period specified in the **Retention time for information security event logs (days)** when [configuring security settings](#). The program deletes early entries in the information security event log when exceeding the space allocated for storing information security events set in **Volume of information security event logs (MB)**.

- Kaspersky MLAD container logs: event date and time, event severity level, name of the container for which the event is registered, event description.

The data is generated by Kaspersky MLAD automatically.

Kaspersky MLAD stores container logs for two days.

The logging system (Grafana) does not transmit users' data to Kaspersky or any third-party servers. You can read the procedure for storing and processing data in the logging system in the [Grafana Logging System User Guide](#).

Data processed on users' computers

When working with the Kaspersky MLAD web interface, the following data is stored in the browser cookie files:

- Individual JSON Web Tokens to support a user session for connecting to the application web interface. An individualized token is stored in the user's browser cookie files for the user inactivity period defined when [configuring the security settings](#).
- ID of the running Grafana session, if the user views the application logs. The Grafana session ID is stored in the user's browser cookie files for 30 days.

The user browser stores data that is used to display the web interface: the last used localization language of the application web interface, the last used option for displaying the main menu (hidden or maximized display), the last used values of the time interval, preset, date and time, ML model element, and the chart display settings in the **Monitoring**, **History**, and **Time slice** sections, the last used page numbering settings, the last set filters for displaying data in the **Event Processor** section, the last used values of the incident status and cause in the **Incidents** section, information about the **Tags for incident #<incident ID>** presets, generated for a registered incident, information about the current installed version of Kaspersky MLAD. This data is stored in the browser indefinitely. You can delete this data from the browser local storage yourself.

When [exporting incidents](#), the application saves an XLSX file with the following data to the user computer:

- Name of monitored asset
- Period during which incidents were uploaded
- ID of the registered incidents
- Date and time when the incidents were registered
- Registered incidents statuses
- Names of the groups that include the registered incidents

- Names and IDs of the top tags that have the greatest impact on the registration of incidents
- Top tag values
- Top tags measurement units
- Top tags descriptions
- Name of the ML models that registered the incidents
- Name of the detectors that registered the incidents.

When [exporting information security event logs](#) from the Grafana logging system, the application saves a CSV file with the following data to the user computer:

- IDs of the information security events
- Date and time when the information security events occurred
- Information security events types
- Information security events subtypes
- Information security events severity levels
- Names of the users whose actions resulted in the registration of the information security events
- IP addresses of the computers from which the users performed the actions stored in the information security event log
- Information security event outcomes
- Brief summaries of the information security events
- Detailed descriptions of the information security events.

When [exporting container logs](#) from the Grafana logging system, the application saves a CSV file with the following data to the user computer:

- Date and time when the events occurred
- Event severity levels
- Name of the container for which the events are registered
- Event description.

When [exporting asset and tag configuration](#), the application saves an XLSX file with the following data to the user computer:

- Asset type ID
- Unique name of the asset type
- Names of the special asset type settings (if any)

- Asset type description (if any)
- Asset ID
- Asset name
- Unique name of an asset within its parent asset
- Asset description (if any)
- Name of the parent asset to which the asset belongs (if any)
- Parent asset ID (if any)
- Names of the special asset settings (if any)
- Values of the special asset settings (if any)
- Tag ID
- Unique name of the tag
- Unique alternative name of the tag (if any)
- Tag description
- Name of the parent asset to which the tag belongs (if any)
- Parent asset ID
- Tag type (if any)
- Tag measurement units
- Lower and upper blocking thresholds (if any)
- Lower and upper signaling thresholds (if any)
- Lower and upper measurement confidence thresholds (if any)
- Lower and upper boundaries for displaying the tag values on charts (if any)
- The expression used to calculate the tag value from the value passed to Kaspersky MLAD
- Tag comment
- Location coordinates of the monitored asset sensor along the abscissa, ordinate, and applicate axes (if any)
- Offset value that must be added to the tag value received from the connector
- Multiplier value by which the tag value received from the connector must be multiplied

When [exporting presets](#), the application saves a JSON file with the following data to the user computer:

- Preset name

- Preset ID
- Sequence number for displaying the preset in the **Presets** section
- List of IDs of tags included in the preset
- Name of the preset icon
- Name of the CSS class for displaying the preset icon
- Information indicating whether the preset should be displayed in the **Time slice** section
- Graphic area parameters within a preset:
 - Graphic area name
 - Graphic area description
 - Sequence number for displaying the graphic area in the preset under **Monitoring, History, and Presets**
 - Upper and lower bounds for displaying tag values in the graphic area
 - Parameters of additional threshold lines:
 - ID of the additional threshold line
 - Threshold value
 - Color of the additional threshold line
 - Axis scale mode
 - Method of scaling the chart in single axis mode
 - List of IDs of tags included in the graphic area
 - ID of graphic area
 - ID of the preset to which the graphic area belongs
- When using a preset to display data in the **Time slice** section, the application saves the following data:
 - Text on the abscissa axis of the chart in the **Time slice** section
 - Name of the expression used to calculate the tag values
 - Text on the ordinate axis of the chart in the **Time slice** section
 - Expression used to calculate the tag values
 - Preset chart color in the **Time slice** section.

When [exporting Kaspersky MLAD settings](#), the application saves configuration files with the following data to the user's computer:

- A file with the settings of the incident statuses, which contains the following data:

- Incident status ID
- Name of the incident status in Russian
- Name of the incident status in English
- Ordinal number of the incident status for sorting
- Information on whether to display registered incidents with this status.
- A file with the settings of the incident causes, which contains the following data:
 - Incident cause ID
 - Name of the cause of the incident
 - Sequential number of the cause of the incident to be sorted.
- A file with the settings of the time intervals for displaying data on the **Monitoring**, **History**, and **Time slice** charts, which contains the following data:
 - Time interval ID
 - Name of the time interval in Russian
 - Name of the time interval in English
 - Ordinal number of the time interval for sorting
 - ID of the user who created the time interval
 - ID of the user who last changed the time interval
 - Time interval value in milliseconds.
- Settings of Kaspersky MLAD services and connectors:
 - Settings IDs
 - Names of the settings in the Kaspersky MLAD database
 - Types of the entered values
 - Entered or selected values
 - Name of the group to which the current setting belongs
 - Serial number of the setting displayed in the current section
 - Requirements for the setting value.
- The Stream Processor configuration file containing the following data:
 - IDs of tags processed by Stream Processor
 - Values of the tag processing settings.

The values of the tag processing settings are set by Kaspersky experts individually for each monitored asset.

- Configuration files of the MQTT Connector, AMQP Connector, and WebSocket Connector containing the following data:
 - Tag IDs obtained from the MQTT Connector, AMQP Connector, or WebSocket Connector
 - Tag timestamp measurement units
 - Type of the received data
 - Template format for decoding the received data type.
- The OPC UA Connector configuration file containing the following data:
 - Tag ID
 - Name of the asset to which the tag belongs
 - Data type passed to the tag value.
- The Event Processor configuration file containing the following data:
 - Rules for mapping event parameters received by the CEF Connector to the names of event parameters to be processed in the Event Processor service
 - List of event parameters to be processed
 - Time and time scale for event processing
 - Order and relationship of the event parameters for display on the relationship graph in the **Event history** section.
- The communication data package for the KICS Connector containing the following data:
 - Encrypted public key of the Kaspersky Industrial CyberSecurity for Networks server certificate, and the certificate issued by the Kaspersky Industrial CyberSecurity for Networks server for the KICS Connector (with the private key).

The contents of the file are encrypted with the password that was set when the KICS Connector was added or when a new communication data package was created for this connector.
 - KICS Connector configuration data: the name of the Kaspersky MLAD user for connecting to the Kaspersky Industrial CyberSecurity for Networks server, the KICS Connector ID, and the address of the Kaspersky Industrial CyberSecurity for Networks server for connection.

Folders for storing application data

Kaspersky MLAD uses the following directories and subdirectories for storing data:

- Application directories (/opt/kaspersky/mlad by default):

- . – root directory of the application. It is used to store configuration files, Kaspersky MLAD installation and update logs, scripts for installing, updating, starting, and stopping Kaspersky MLAD, and the distribution package signatures. The root directory of the application contains notes on the current release of Kaspersky MLAD (Release Notes).
- ./configs – directory for storing configuration files of Kaspersky MLAD. The directory contains the logger subdirectory, which stores the configuration files of the Logger service.
- ./data – directory for storing data that is loaded using the HTTP Connector.
- ./legal – directory for storing the text of the End User License Agreement, the date of its acceptance by the user, and the legal_notices.txt file, which contains information about third-party code.
- ./ssl – directory for storing the script for generating a self-signed certificate that provides an HTTPS connection to the Kaspersky MLAD user's browser.
- ./ssl/tokens – directory for storing a JWT (JSON Web Token) key.
- ./ssl/nginx – directory for storing certificates supporting an HTTPS connection with the browser of the Kaspersky MLAD user.
- ./ssl/public_cert – directory for storing public keys used to verify the digital signature of the distribution package.
- ./mlad_backup-<version number>-<build number> – default directory for storing backup copies of Kaspersky MLAD. A backup copy of Kaspersky MLAD can be created by using the [backup.sh script](#) or during the [application upgrade](#) process by using the upgrade.sh script. When performing a backup of the application, you can specify a different directory for storing the backup copy. The contents repeat the structure of the root directory where Kaspersky MLAD is installed, and they contain the following subdirectories:
 - ./containers_backup – directory for storing an archive containing a backup copy of containers for Kaspersky MLAD services.
 - ./volumes_backup – directory for storing an archive containing a backup copy of Docker volumes.
- ./volumes_backup_<date of deletion> – directory for storing backup copies of Docker volumes that are created during removal of Kaspersky MLAD.
- Directory /var/lib/docker/volumes/:
 - ./<application directory name>_postgres-volume – directory for storing Postgres database files.
 - ./<application directory name>_influxdb-volume – directory for storing Time Series Database service files.
 - ./<application directory name>_logger-volume – directory for storing files of the logging subsystem.
 - ./<application directory name>_webstatic-volume – directory for storing static data of the application web interface.
- /etc/hosts – service file describing the mapping between IP addresses and host names of the external servers.

Application files can be modified by an administrator or by the user who unpacked the archive containing the installation script and all the files required for installation of Kaspersky MLAD.

Deleting or modifying any file of Kaspersky MLAD can negatively impact the performance of the application.

System administrator tasks

This section contains a description of the system administrator tasks performed in the [administrator menu](#) of the application.

Managing user accounts

To ensure that users securely work with Kaspersky MLAD, [create an account](#) for each user.

Kaspersky MLAD user accounts can be managed only by system administrators.

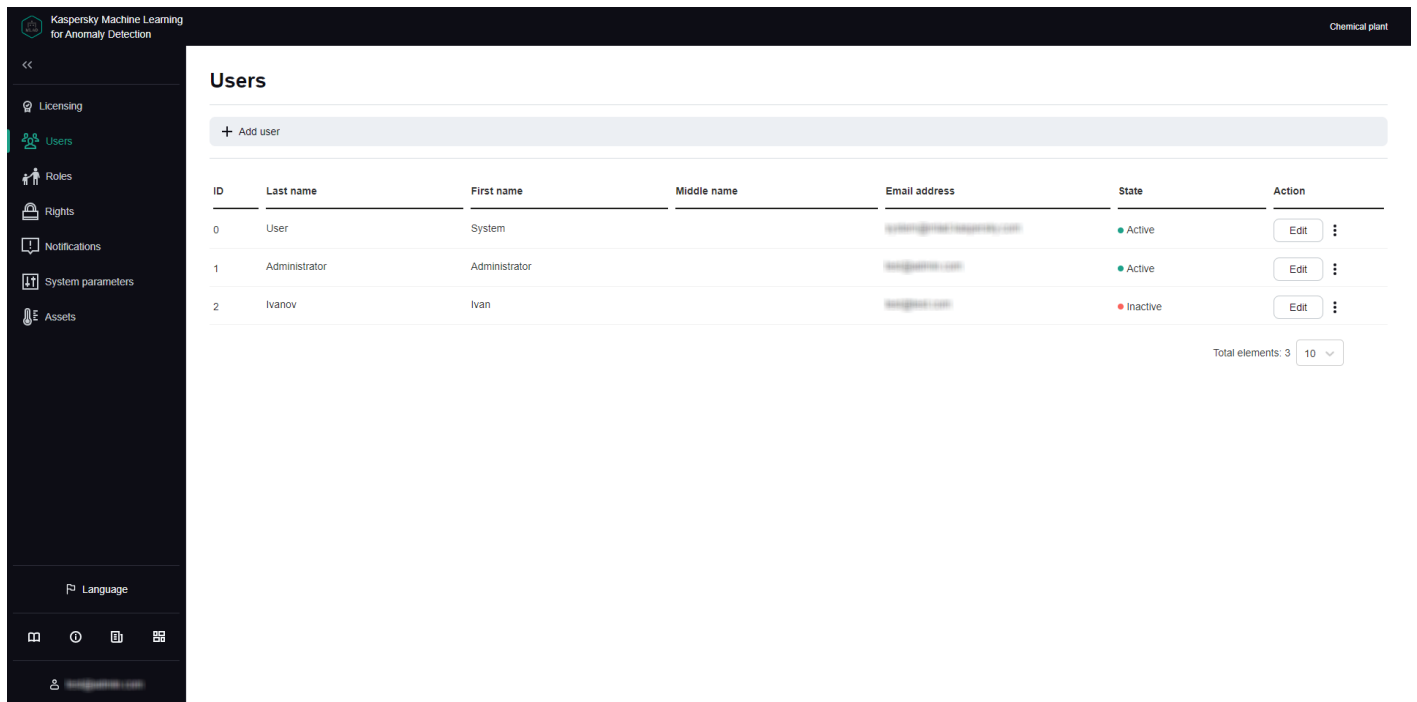
All created user accounts and [information about them](#) are displayed in the table in the **Users** section of [the administrator menu](#).

- **ID** is the user identifier.
- **Last name** refers to the last name of the user.
- **Name** refers to the first name of the user.
- **Middle name** refers to the middle name of the user.
- **Email address** refers to the user's email address.
- **State** is a parameter that describes the blocking status of a user account. If the user is locked, a **State** red dot appears in the **Inactive** column. A blocked user cannot log in to Kaspersky MLAD to use the application. If the user is unlocked, an **Active** green dot appears in the **State** column.
- **Action** is a button that lets you [change the user account](#).

When installing the application, a special User **System** account is created. This account is not intended for use by personnel when working with Kaspersky MLAD. This account cannot be used to connect to the application web interface. To clarify whether or not you can change its settings, you are advised to consult with Kaspersky experts or a certified integrator.

You can [modify](#) user accounts as needed. Kaspersky MLAD does not allow you to delete user accounts. To prevent a specific account from accessing Kaspersky MLAD web interface, it is recommended to [block](#) this account. You can unblock this user account later if necessary. If an account was locked when the number of unsuccessful login attempts for that user was reached, you can unblock this account before the blocking period expires. You can specify the number of unsuccessful authorization attempts and the account blocking period when [configuring the security settings of Kaspersky MLAD](#).

Next to each account, there is a vertical menu that lets you [revoke authentication tokens](#) or [view the list of rights](#) for the specific user account.



Users section

Creating a user account

Kaspersky MLAD user accounts can be managed only by system administrators.

To create a user account:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Users** section.

3. Click the **Add user** button.

The **Add user** panel appears on the right.

4. In the **Last name** field, enter the last name of the user.

5. In the **Name** field, enter the first name of the user.

6. If necessary, enter the middle name of the user in the **Middle name** field.

7. In the **Email address** field, enter the email address of the user.

8. In the **Password** field, enter a password for the user account.

The password must meet the following requirements:

- Must contain the minimum number of characters defined in the [Minimum password length](#) setting.
- Must contain letters of the English alphabet, numerals and/or special characters in accordance with the password policy that was set when [configuring the security settings](#).

9. In the **Confirm password** field, type the password again to confirm the password for the user account.

10. Click the **Save** button.

Information about the new user will be displayed in the table. If necessary, you can [modify](#) user accounts and [revoke](#) their authentication tokens

When creating an account, you cannot assign a role to a user. You can assign a role to a user only when [editing the user account](#).

Editing a user account

Kaspersky MLAD user accounts can be managed only by system administrators.

When you edit a user account, you can assign the desired [role](#) to the user. You can also block or unblock a user account.

When a user is blocked in Kaspersky MLAD, their authentication tokens are automatically revoked, and their user sessions are terminated. A blocked user cannot log in to Kaspersky MLAD to use the application.

If the password for a user account is changed, Kaspersky MLAD also revokes authentication tokens automatically and terminates the user session of the user account whose password was changed. A user whose password has been changed can log in to the web interface with the updated password.

To edit a user account:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Users** section.

3. Click the **Edit** button in the row of the user account that you want to edit.

The **Edit user** pane appears on the right.

4. Enter the user's new last name, first name, and/or middle name, if needed.

5. In the **Roles** field, assign a [role](#) for the user account by selecting the corresponding check box.

6. If required, enter the user's new email address.

7. If you need to change the password, enter the new password in the **Password** and **Confirm password** fields.

The new password must meet the following requirements:

- Must not match previously used passwords. The specific number of most recently used passwords that must not be reused is defined by the value of the [Number of user passwords stored in history](#) setting.
- Must contain the minimum number of characters defined in the [Minimum password length](#) setting.
- Must contain letters of the English alphabet, numerals and/or special characters in accordance with the password policy that was set when [configuring the security settings](#).

8. If you want to block or unblock a user account, perform one of the following actions:

- If you want to unblock a user account, set the **State** toggle switch to the **Active** position.
- If you want to block a user account, set the **State** toggle switch to the **Inactive** position.

Kaspersky MLAD does not allow you to delete user accounts. If you want to prevent a specific user account from accessing Kaspersky MLAD, it is recommended to block this user account.

9. Click the **Save** button.

The updated information about the user will be displayed in the table.

Revoking authentication tokens for a user account

Kaspersky MLAD user accounts can be managed only by system administrators.


After a user connects to the Kaspersky MLAD web interface, an individualized token is created so that the user authorization in the application can be saved between connection sessions to the application web interface, including when the browser is restarted. If a user is authorized on multiple assets, a token is created for each user session.

If necessary, you can revoke tokens for a user account at any time. For the user whose tokens are revoked, the connection session is terminated simultaneously on all assets where they were authorized. Revoking tokens may be useful if you need to immediately terminate application connection sessions for a specific user. After the tokens are revoked, the user can log in again and continue using Kaspersky MLAD.

To revoke tokens for a user account:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Users** section.
3. Open the vertical menu  next to the account you want to revoke tokens from, and select **Revoke tokens**.
4. In the window that opens, confirm that you want to revoke the authentication tokens.

The user account tokens are revoked, and the user session is terminated.

Viewing access rights for a user account

Kaspersky MLAD user accounts can be managed only by system administrators.


In the **Users** section, you can view the list of rights for a specific user account.

To view the access rights for a user account:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Users** section.

3. Open the vertical menu  next to the account you want to view the list of permissions for, and select **List of rights**.

The page displays a window containing information about the role and access rights of the selected user account.

Manage roles


In Kaspersky MLAD, you can use common roles to restrict user access to [application functions](#) depending on the tasks performed by specific users.

Role management is available to system administrators.

A *role* is a set of rights to access application functions that you can assign to a user.

Accounts with the following roles can be used to access application functions:

- The system administrator role is created automatically during installation of the application. The system administrator role is automatically assigned to the first user created during installation of Kaspersky MLAD. A user with the system administrator role has access to all functions of the application. The system administrator role cannot be modified or removed.
- A custom role is created manually in the **Roles** section. Access to application functions depends on the list of rights granted to the custom role. The number of user roles is unlimited.

The **Roles** section displays a table with [information about all created roles](#) .

- **ID** is the user role ID.
- **Role** is the name of the user role.
- **State** is a setting that describes the state of the role. If the role is in use, an **Active** green dot appears in the **State** column. If the role is not in use, an **Inactive** red dot appears in the **State** column. If a role is inactive, the features associated with the role's permissions are unavailable to the user assigned to that role.
- **Rights** is a button that lets you view a list of the rights of a user role in Kaspersky MLAD.
- **Created** refers to the date and time when the user role was created.
- **Updated** refers to the date and time when the user role was updated.

Creating role

Role management is available to system administrators.

You can create custom roles and select the [access rights to application functions](#) for them. After an active role is created, it will become available for [assignment to application users](#).

To create a role:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Roles** section.

3. Click the **Create** button.

The **Creating role** pane appears on the right.

4. In the **Role name** field, specify the required role name.

You can enter up to 30 characters.

5. If necessary, enter a new description for the tag in the **Role description** field.

6. To grant access rights to a role, do the following:

- a. Click the **Select rights** button.

The **Grant rights to role** pane appears on the right.

- b. In the list of rights, select the access rights to application functions that you want to grant to the role.

When you select **Rights to all actions**, all [system administrator functions](#) will be available to the role.

- c. Click the **Save** button.

7. Perform one of the following actions:

- If you need to use a role for application users, set the **State** switch to the **Active** position.
- If you need to disable the use of a role for application users, set the **State** toggle switch to the **Inactive** position.

If a role is inactive, the features associated with the role's permissions are unavailable to the user assigned to that role.

8. Click the **Save** button.

Editing role

Role management is available to system administrators.

To change a role:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Roles** section.
3. Check the box next to the role you want to edit.
4. Click the **Edit** button.

The **Editing role** pane appears on the right.

5. Edit the role settings. For a description of the settings, see the [instructions on creating a role](#).
6. Click the **Save** button.


Deleting role

Role management is available to system administrators.

When a user role assigned to Kaspersky MLAD users is deleted, those users will lose access to the features associated with that role.

System administrator role cannot be deleted.

To delete a role:


1. In the lower-left corner of the window, click .
- You will be taken to the [administrator menu](#).
2. Select the **Roles** section.
3. Select the check boxes next to the names of the roles that you want to remove.
4. Click the **Delete** button.
5. In the window that opens, confirm that you want to delete the roles.

Viewing access rights for a role

Role management is available to system administrators.

In the **Roles** section, you can view a list of access rights to application functions for users with a specific role.

To view the access rights for a role:

1. In the lower-left corner of the window, click .
- You will be taken to the [administrator menu](#).
2. Select the **Roles** section.

3. Click the **List of rights** button next to the role for which you want to view the list of rights.


A window opens on the page with information about access rights to application functions for the selected role.

Managing incident notifications

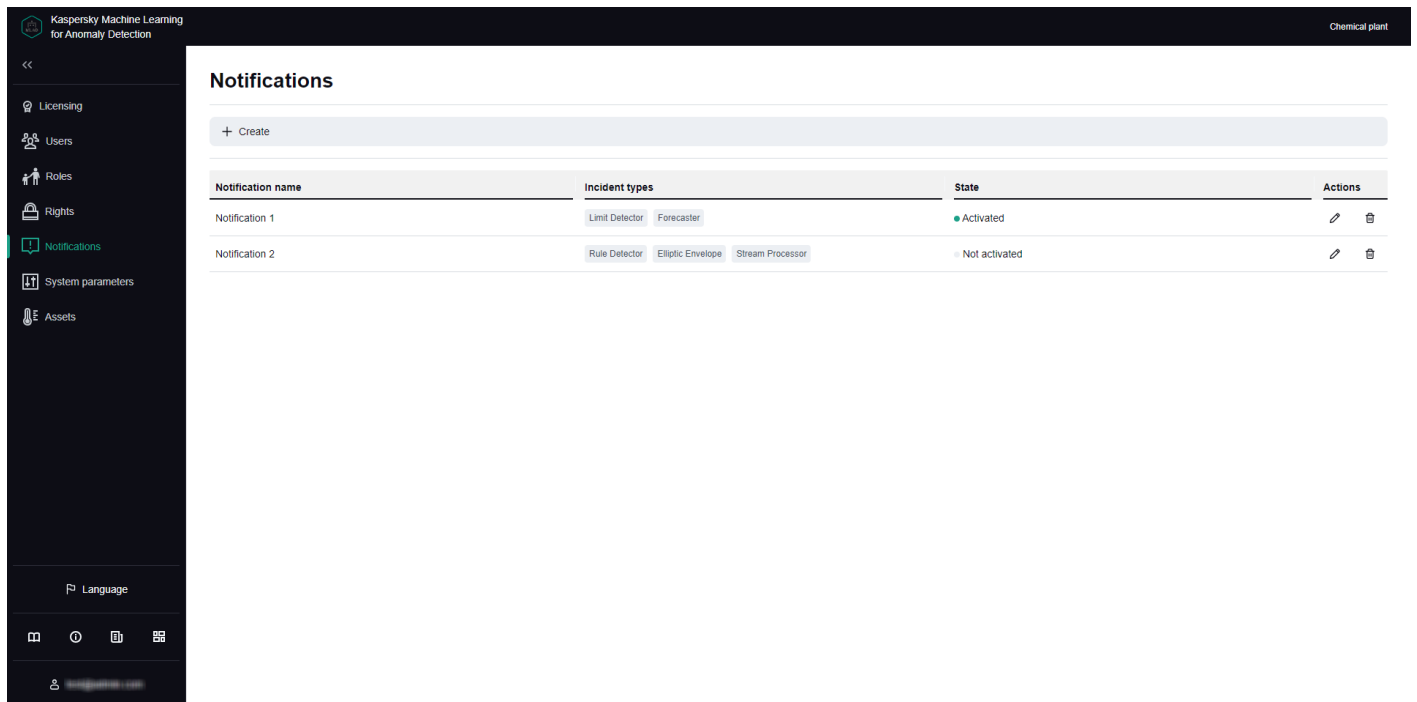
You can [set up notifications](#) for Kaspersky MLAD to send to users about incidents identified through the analysis of telemetry data or from ML model output. Notifications are sent to the email addresses specified in the notifications. You can [edit](#) and [delete](#) notifications regarding incidents for Kaspersky MLAD users.

Only system administrators can manage incident notifications.

The Mail Notifier service must be [configured](#) and [started](#) in advance.

All created notifications about incidents and [information about them](#)  are displayed in the **Notifications** section in the [administrator menu](#). If necessary, you can change the number of notifications displayed on one page.

- **Notification name:** incident notification name.
- **Incident types** refers to the types of incidents that the user receives notifications about. You can be notified about the following types of incidents:
 - [Predictive elements](#) are the incidents detected by ML model predictive elements.
 - [Diagnostic rules](#) are the incidents detected by ML model diagnostic rules.
 - [Elliptic envelopes](#) are the incidents detected by ML model elliptic envelopes.
 - [Limit Detector](#): incidents registered when a tag reaches an upper or lower blocking threshold.
 - [Stream Processor](#): incidents registered if data loss is detected or if observations are received by Kaspersky MLAD too early or too late.
- **State** indicates whether this notification is in use.
- **Actions:** buttons for [editing](#) or [deleting](#) an incident notification.



Notifications section

Creating an incident notification

Only system administrators can manage incident notifications.

You can create notifications about incidents recorded during telemetry data processing and/or as a result of ML model operation.

To create an incident notification for a user:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Notifications** section.

3. On the opened page, click the **Create** button.

The **Create notification** window opens.

4. Specify the name of the notification in the **Name** field.

5. To enable sending of notifications, set the **State** toggle switch to the **Activated** position.

6. In the **Notification language** field, select the language of the delivered incident notifications.

By default, the current localization language of the Kaspersky MLAD web interface is used for incident notifications. It is available in English and Russian.

7. In the **Users email addresses** drop-down list, select the email addresses of the application users to send notifications to.

8. In the **Additional email addresses** field, specify additional email addresses to send incident notifications to, separated with a semicolon.
9. To enable notifications about incidents registered by ML models, do the following:
 - a. To configure notifications about [incidents registered by predictive elements of ML models](#), select **Predictive elements**.
 - b. To configure notifications about [incidents registered by ML model diagnostic rules](#), select **Rules**.
 - c. To configure notifications about [incidents registered by elliptic envelopes of ML models](#), select the **Elliptic envelopes** check box.
 - d. Select one or more ML models to send notifications for.
To select all ML models within an asset, check the box next to the asset name.
10. To enable notifications about incidents registered only by published ML models, set **Send incident notifications for published models only** to **Activated**.
11. To enable incident notifications for tags:
 - a. To configure notifications for when [a tag has reached an upper or lower blocking threshold](#), select **Limit Detector**.
 - b. If you want to configure sending of the notifications [about the termination or interruption of the input data stream for a specific tag](#), or about the detection of observations that arrived too soon or too late, select the **Stream Processor** check box.
 - c. Select one or more tags you want to send notifications for.
To select all tags within an asset, check the box next to the asset name.

12. Click the **Save** button.



Information about the new notification will be displayed in the table. If necessary, you can [edit](#) or [delete](#) notifications.

A separate notification will be sent to the specified email addresses for each incident registered for selected ML models and/or tags.

Editing an incident notification

Only system administrators can manage incident notifications.

To edit an incident notification:

1. In the lower-left corner of the window, click .
- You will be taken to the [administrator menu](#).
2. Select the **Notifications** section.
3. Click the  button next to the notification that you want to edit.

4. Adjust the incident notification settings, if needed. For a description of the settings, see the [instructions on setting up an incident notification](#).

5. Click the **Save** button to save the changes.

A separate notification will be sent to the specified email addresses for each incident registered for selected ML models and/or tags.

The updated information about the notification will be displayed in the table. If necessary, you can [delete](#) notifications.

Deleting an incident notification

Only system administrators can manage incident notifications.

To delete an incident notification:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Notifications** section.

3. Click the  button next to the notification that you want to delete.

4. In the window that opens, confirm the deletion of the notification.

Information about the notification will be deleted from the table.

Kaspersky MLAD allows you to temporarily disable sending of notifications instead of deleting their configuration. Information about notifications is saved in the **Notifications** section. You can enable sending of notifications at any time. You can enable or disable notifications by [editing the corresponding incident notification](#).

Configuring Kaspersky MLAD

This section contains instructions on configuring the settings of Kaspersky MLAD services and connectors, as well as on configuring security settings, logging levels for application services, settings for displaying the application menu, and on managing typical statuses and causes of incidents.

Configuring the main settings of Kaspersky MLAD

Kaspersky MLAD lets you specify the name of the monitored asset, web address and IP address for connecting users to the application web interface, and the frequency of receiving new data from the monitored asset. The name of the monitored asset will be displayed in the upper right corner of each section of the Kaspersky MLAD web interface.

System administrators can configure the main settings of Kaspersky MLAD.

To configure the main settings of Kaspersky MLAD:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **System parameters** → **Main** section.

A list of options appears on the right.

3. In the **Name of monitored asset** field, specify the name of the monitored asset.

4. In **URL address for incident notifications**, specify a URL for generating a link to incidents in email incident alerts.

If **IP address for incident notifications** is set, this is used for generating the link.

5. In **IP address for incident notifications**, specify an IP for generating a link to incidents in email incident alerts.

6. In the **Interval for receiving data from the Message Broker service (ms)** field, specify the interval for updating telemetry data in the application web interface.

The higher the specified parameter value, the less frequently the data is updated. The default value of this parameter is 500.

7. In the **Interval for receiving incident statistics from the database (ms)** field, indicate how frequently data on incidents registered by the application should be updated in the application web interface.

The default value of this parameter is 5000.

8. In the **Monitored asset time zone** drop-down list, select the time zone of the computer where Kaspersky MLAD is installed

The graphs in the application show data in the time zone you select.

9. Click the **Save** button.

Configuring the security settings of Kaspersky MLAD

Kaspersky MLAD lets you specify the conditions for temporarily blocking user accounts, the user inactivity period in accordance with the enterprise security policy, and the settings for storing information security event logs in the Kaspersky MLAD database. Information security event logs are automatically written to the database. If necessary, you can [specify the settings of an external system](#) to which the information security event logs should be sent.

System administrators may be responsible for configuring the security settings of Kaspersky MLAD.

To configure the main settings of Kaspersky MLAD:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **System parameters** → **Security** section.

A list of options appears on the right.

3. In the **Authorization parameters** block, do the following:

- a. In the **Number of authentication attempts** field, specify the number of unsuccessful authorization attempts. When this number is reached, Kaspersky MLAD temporarily blocks the corresponding user account.
The default value of this parameter is 3.
- b. In the **User lock duration (sec)** field, specify the time period (in seconds) to block a user account after reaching the specified number of unsuccessful authorization attempts.
The default value of this parameter is 120.
- c. In the **User inactivity period (min)** field, specify the permissible duration of an inactive user session (in minutes).
When the specified time period is reached, Kaspersky MLAD automatically terminates the inactive user session. The default value of this parameter is 1440.
- d. If you need to prevent users from ignoring the password change recommendation when they connect to the application web interface for the first time, turn on the **Require password change on first login** toggle switch.
This switch is disabled by default.

4. In the **Password policy** settings block, do the following:

- a. In the **Number of user passwords stored in history** field, specify the number of most recent user passwords that are stored in the application.
You can specify a value starting with 1. The default value of this parameter is 5.
When the user password is changed, the new password must not match any passwords stored in Kaspersky MLAD. The application stores passwords in encrypted form.
- b. In the **Password expiration period (days)** field, specify the number of days during which the user can use their current password to connect to the application without changing it.
The default value of this parameter is 180.
- c. In the **Minimum password length** field, specify the minimum number of characters for user passwords.
You can specify a value in the range of 8 to 128. The default value of this parameter is 8.
- d. If your security policy stipulates that user passwords must contain uppercase letters of the English alphabet, turn on the **Require to use uppercase letters of the English alphabet (A-Z)** toggle switch.
This switch is enabled by default.
- e. If your security policy stipulates that user passwords must contain lowercase letters of the English alphabet, turn on the **Require to use lowercase letters of the English alphabet (a-z)** toggle switch.
This switch is enabled by default.
- f. If your security policy stipulates that user passwords must contain numerals, turn on the **Require to use numerals (0-9)** toggle switch.
This switch is enabled by default.
- g. If your security policy stipulates that user passwords must contain special characters, turn on the **Require to use special characters (_!@#\$%^&*)** toggle switch.
This switch is enabled by default.

5. In the **Retention settings for information security event logs** block, do the following:

- a. In the **Volume of information security event logs (MB)** field, specify the volume limit (in megabytes) for storing information security event logs in the database.

If the field is blank, Kaspersky MLAD stores all information security event logs for the time period specified in the **Retention time for information security event logs (days)** setting. This setting has no value by default.

If the specified volume of information security event logs in the database is exceeded, Kaspersky MLAD deletes the oldest entries.

- b. In the **Retention time for information security event logs (days)** field, specify the number of days to store information security event logs in the database.

The default value of this parameter is 100.

6. Click the **Save** button.

Configuring the Anomaly Detector service

You can configure the procedure for detecting anomalies based on the specific features of your monitored asset by enabling or disabling specific anomaly detection in the Anomaly Detector service settings.

System administrators can configure the Anomaly Detector service.

To configure the settings of the Anomaly Detector service:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select **System parameters** → **Anomaly Detector**.

A list of options appears on the right.

3. Enable or disable the Limit Detector using the **Use Limit Detector** toggle switch.

Limit Detector logs incidents when the upper or lower blocking thresholds set for the tag are exceeded.

4. Use the **Use Forecaster detector** toggle switch to enable or disable anomaly detection with [ML model predictive elements](#).

ML model predictive elements register incidents when detecting discrepancies between observed and predicted tag values.

5. Enable or disable the XGBoost detector using the **Use XGBoost detector** toggle switch.

6. Use the **Use Rule Detector** toggle switch to enable or disable anomaly detection with [ML model elements based on diagnostic rules](#).

Diagnostic rules register incidents when the output of a diagnostic rule exceeds a predetermined limit.

7. Enable or disable the function for skipping gaps in the incoming data stream using the **Skip gaps in data** toggle switch.

If the toggle switch is on, during ML model inference, its components do not generate any [artifacts](#) when no data is received for the ML model element tags for a period longer than the UTG period as specified in **Grid step (sec)** for that element.

8. In the **Maximum number of records requested from the Message Broker service** field, enter the number of records that must be requested from the Message Broker service for subsequent processing in the Anomaly Detector.

The higher the value, the less frequently Anomaly Detector requests records from Message Broker. The value depends on the amount of telemetry data received by Kaspersky MLAD in real time.

9. In the **Number of messages sent in one block to the Message Broker service** field, enter the number of incidents that must be sent to the Message Broker service at one time.

10. Click the **Save** button.

Configuring the Keeper service

Kaspersky MLAD uses the Keeper service to route telemetry data that should be saved in the database. You can configure the settings that define the rate of incoming data received from connectors and external sources, and the volume of data to be saved in the Kaspersky MLAD database.

System administrators can configure the data routing settings in Kaspersky MLAD.

To configure the Keeper service:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **System parameters** → **Keeper** section.

A list of options appears on the right.

3. Perform one of the following actions:

- To save both known and unknown tags values from external sources to the database, turn on the **Save values of all tags** toggle.
- To save only the tags values that are known to the application, turn off the **Save values of all tags** toggle.

4. In the **Timeout for receiving tag values (ms)** field, enter the maximum timeout (in milliseconds) for receiving the values of tags.

The higher the value, the less frequently Keeper receives tag values and writes these to the database. We recommend setting a value that matches the frequency of receiving telemetry data from the monitored asset.

5. In the **Timeout for receiving incidents (ms)** field, enter the maximum timeout in milliseconds for receiving incidents.

The higher the value, the less frequently Keeper receives incident data and writes these to the database. We recommend setting a value that matches the incident frequency.

6. In the **Timeout for receiving metrics (ms)** field, enter the maximum timeout in milliseconds for receiving metrics.

The higher the specified value, the less often Keeper receives metrics (the number of received observations for tags, the number of events, and the number of generated artifacts) and writes these to the database. We recommend setting a value that matches the frequency of receiving metrics from CEF Connector.

7. Click the **Save** button.

Configuring the Mail Notifier service

Kaspersky MLAD uses the Mail Notifier service to notify users when incidents are registered by the application.

System administrators can configure the Mail Notifier service.

Setting up Mail Notifier is optional and only required for receiving email [notifications about new registered incidents](#). First, you need to configure an SMTP server on the network where the monitored object is located.

To configure the Mail Notifier service:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select **System parameters** → **Mail Notifier**.

A list of options appears on the right.

3. In the **Notification sender email** field, specify the email address that will send incident notifications.

4. In the **SMTP server address** field, enter the IP address of the SMTP server.

5. In the **SMTP server port** field, enter the port of the SMTP server.

6. In the **SMTP server user name** field, enter the user name for connecting to the SMTP server.

7. In the **SMTP server password** field, enter the password for connecting to the SMTP server.

8. Use the **Use TLS connection** toggle switch to enable or disable secure TLS connection.

By default, use of a secure TLS connection is disabled.



To avoid compromising the sent data, it is recommended to enable the use of a secure TLS connection. It is recommended to use a secure TLS connection via the TLS-1.2 or TLS-1.3 protocol using a cipher suite from the [list of recommended ciphers](#).

9. If you are using a secure TLS connection, do the following:

- a. Upload the SMTP server certificate using the **Browse** button under **SMTP server certificate**.

- b. Upload the key to the SMTP server certificate file using the **Browse** button under the **Key to SMTP server certificate** setting.

It is recommended to use a certificate with a certificate key length of 4096 bits when using the RSA algorithm, or 256 bits when using the ECDH algorithm.
Certificates and certificate keys can be uploaded only as files in DER or PEM format.

To delete the certificate file or certificate key, click the  button in the corresponding field. To save the certificate file or certificate key on your computer, click the  button in the corresponding field.

10. Click the **Save** button.

Configuring the Similar Anomaly service

Kaspersky MLAD uses the Similar Anomaly service to identify similar incidents and combine them into groups. In groups, you can [view similar incidents](#) that were registered at different times.

System administrators can configure the Similar Anomaly service.

To configure the Similar Anomaly service:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select **System parameters** → **Similar Anomaly**.

A list of service settings appears on the right.

3. In the **Minimum number of incidents in group** field, enter the minimum number of similar incidents for forming a group.

4. In the **Maximum number of incidents in group** field, enter the maximum number of incidents that can be put into one group.

The larger the specified value, the more incidents the application can assign to one group.

5. In the **Maximum distance between similar incidents** field, enter the maximum distance that similar incidents can lag behind each other.

You can specify a value in the range of 0 to 1.

6. Click the **Save** button.

Configuring the Stream Processor service

The Stream Processor service gathers real-time telemetry data (input stream) received from the monitored asset at arbitrary points in time and converts this data to a UTG (output stream). Based on the accumulated data, the Stream Processor service determines the values of tags in the output data stream. After converting data into an output stream, the Stream Processor service forwards this data to the ML model for processing.




When converting incoming telemetry data, the Stream Processor service accounts for potential data losses (for example, if the network of the monitored asset temporarily goes down) and processes observations that were received in Kaspersky MLAD too early or too late. In these cases, the Stream Processor service generates default incidents and/or forwards default tag values to the output data stream.

The Stream Processor service can also compute derivative tags based on incoming telemetry data (for example, to calculate the moving average or average value of a group of tags).

The Stream Processor service configuration file for uploading is provided by Kaspersky specialists or a certified integrator.

System administrators can configure the Stream Processor service.

To configure the Stream Processor service:

1. In the lower-left corner of the window, click .
You will be taken to the [administrator menu](#).
2. Select **System parameters** → **Stream Processor**.
3. In the **Uniform temporal grid period (sec)** field, specify the period (in seconds) for which the Stream Processor service will process incoming telemetry data.
4. Using the **Browse** button under the **Configuration file** setting, add a file that contains configuration settings for the Stream Processor service.
If you need to delete the configuration file for the Stream Processor service, click the  button. To save the configuration file on your computer, click the  button.
5. Click the **Save** button.

Configuring the HTTP Connector


Kaspersky MLAD uses the HTTP Connector to receive data from CSV files by uploading data using the POST method. You can download data over port 4999 via HTTP or HTTPS by specifying the relevant protocol in a request.

In Kaspersky MLAD, certificate files and key files can be uploaded only in DER or PEM format. If necessary, you can use the OpenSSL utility to change the format of certificate files and the certificate key. Thus, to change the format of a certificate file from P12 to PEM, run the following command in the console:

```
openssl pkcs12 -in <certificate name>.p12 -clcerts -nokeys -out <certificate name>.pem
```

System administrators can configure the HTTP Connector.

To configure the HTTP Connector:

1. In the lower-left corner of the window, click .
You will be taken to the [administrator menu](#).
2. Select **System parameters** → **HTTP Connector**.
A list of options appears on the right.
3. In the **Size of written block (tag count)** field, specify the number of tags values that are written to the Message Broker service at one time.
The specified value affects the number of iterations for writing tag values from the CSV file to Message Broker according to the total number of observations for tags retrieved via HTTP Connector.
4. In the **Maximum size of uploaded file (MB)** field, specify the maximum size in megabytes of a file transmitted to the HTTP Connector.
If you try to download a larger CSV file, the file would not be passed to the HTTP Connector.

5. Use the **Use TLS connection** toggle switch to enable or disable secure TLS connection.

By default, use of a secure TLS connection is enabled.

To avoid compromising the received and/or sent data, it is recommended to keep the use of a secure TLS connection enabled.

6. If you are using a secure TLS connection, use the **Use the recommended TLS connection settings** toggle switch to enable or disable use of the recommended TLS connection settings.

By default, use of the recommended TLS connection settings is enabled.

When the toggle switch is on, a secure TLS connection is used via the TLS-1.2 or TLS-1.3 protocol with a cipher suite from the [list of recommended ciphers](#).



7. If you are using a secure TLS connection, do the following:

- a. Add the HTTPS server certificate and the certificate key by using the **Browse** button under the **HTTPS server certificate** and **Private key to HTTPS server certificate** settings.

It is recommended to use a certificate with a certificate key length of 4096 bits when using the RSA algorithm, or 256 bits when using the ECDH algorithm.

- b. If you are using client certificates, add the root certificate to verify the signature of the client certificate by using the **Browse** button under the **CA certificate for verifying the client certificate signature** setting.

Certificates and certificate keys can be uploaded only as files in DER or PEM format.

To delete the certificate file or certificate key, click the  button in the corresponding field. To save the certificate file or certificate key on your computer, click the  button in the corresponding field.

8. Toggle **Scale obtained tag values** switch to enable or disable the conversion of tag values according to the **Bias** and **Multiplier** settings that were set when [creating the tag](#).

Conversion of received tag values is disabled by default.

9. Click the **Save** button.

Kaspersky MLAD will receive data from CSV files using the HTTP Connector.

The following is an example of sending a CSV file to the HTTP Connector via cURL over HTTP using the POST method to port 4999 of the Kaspersky MLAD server:

```
curl -F "file=@<file name>.csv" -X POST "http://<Kaspersky MLAD server IP address or domain name>:4999/upload"
```

The HTTP Connector accepts CSV files with the following fields:

timestamp;tag_name;value

where:

- **timestamp** is the time stamp in the format %Y-%m-%dT%H:%M:%S.
- **tag_name** is the name of the tag.

- value is the tag value.

If a tag value contains a fractional portion, use a dot to separate the integer from the fractional portion.

Configuring the MQTT Connector

Kaspersky MLAD uses the MQTT Connector to receive data and send messages about incident registration via the MQTT (Message Queuing Telemetry Transport) protocol.

System administrators can configure the MQTT Connector.

To configure the MQTT Connector:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select **System parameters** → **MQTT Connector**.

A list of options appears on the right.

3. Use the **Use TLS connection** toggle switch to enable or disable secure TLS connection.

By default, use of a secure TLS connection is enabled.

To avoid compromising the received and/or sent data, you are advised to keep the use of a secure TLS connection enabled.

4. If you are using a secure TLS connection, use the **Use the recommended TLS connection settings** toggle switch to enable or disable use of the recommended TLS connection settings.

By default, use of the recommended TLS connection settings is enabled.

When the toggle switch is on, a secure TLS connection is used via the TLS-1.2 or TLS-1.3 protocol with a cipher suite from the [list of recommended ciphers](#).



5. In the **MQTT broker (address:port)** field, specify the host name and port of the external MQTT broker that the MQTT Connector will interact with.

The default value of this parameter is `mqtt_broker:1883`.

6. In the **User name for MQTT connection** field, enter the user name to connect to the MQTT broker.

7. In **Password for MQTT connection**, enter the user password for connecting to the MQTT broker.

8. If you are using a secure TLS connection and a self-signed certificate is installed on the MQTT broker, add the root certificate for the MQTT broker by using the **Browse** button under the **CA certificate** setting.



To delete the certificate file, click the  button. To save the certificate file on your computer, click the  button.

9. If you are using a secure TLS connection and client authentication is enabled on the MQTT broker, do the following:

- a. Add the client certificate by using the **Browse** button under the **Client certificate** setting.

- b. Add the key for the client certificate by using the **Browse** button under the **Key to client certificate** setting.

It is recommended to use a certificate with a certificate key length of 4096 bits when using the RSA algorithm, or 256 bits when using the ECDH algorithm.
Certificates and certificate keys can be uploaded only as files in DER or PEM format.

To delete the certificate file or certificate key, click the  button in the corresponding field. To save the certificate file or certificate key on your computer, click the  button in the corresponding field.

10. In the **List of MQTT subscriptions for receiving tags** field, enter the name of the list of MQTT subscriptions from which the MQTT Connector will receive tag values.

The default value of this parameter is **tags**.

11. In the **MQTT topic for publishing messages** field, specify the name of the topic where the MQTT Connector will publish messages about incident registration.

If no value is defined for this setting, messages are not sent.

This setting has no value by default.

12. In the **Data format** drop-down list, select the format to receive data from external systems and send messages about incidents.

The following options are available: **JSONBatch**, **Topic**, **SmartHome**, **KISG**.

The default value of this parameter is **JSONBatch**.

If you are having difficulty selecting a data format, consult Kaspersky or a certified integrator.

If none of the incident data and message formats suits you, you can contact Kaspersky Lab experts to add the required format.

13. If you have selected the **Topic** data format, add a configuration file containing the connector settings for this data format using the **Browse** button under the **Connector configuration file** setting.

To delete the certificate file, click the  button. To save the certificate file on your computer, click the  button.

14. Toggle **Scale obtained tag values** switch to enable or disable the conversion of tag values according to the **Bias** and **Multiplier** settings that were set when [creating the tag](#).

Conversion of received tag values is disabled by default.

15. Click the **Save** button.


Kaspersky MLAD will receive data and send messages about incident registration via the MQTT protocol.

Configuring the AMQP Connector

Kaspersky MLAD uses the AMQP Connector to receive data and send messages about incident registration via AMQP (Advanced Message Queuing Protocol).

System administrators can configure the AMQP Connector.

To configure the AMQP Connector:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select **System parameters** → **AMQP Connector**.

A list of options appears on the right.

3. Use the **Use TLS connection** toggle switch to enable or disable secure TLS connection.

By default, use of a secure TLS connection is enabled.

To avoid compromising the received and/or sent data, you are advised to keep the use of a secure TLS connection enabled.

4. If you are using a secure TLS connection, use the **Use the recommended TLS connection settings** toggle switch to enable or disable use of the recommended TLS connection settings.

By default, use of the recommended TLS connection settings is enabled.

When the toggle switch is on, a secure TLS connection is used via the TLS-1.2 or TLS-1.3 protocol with a cipher suite from the [list of recommended ciphers](#).

5. In the **AMQP broker (address:port)** field, specify the host name and port of the external AMQP broker that the AMQP Connector will interact with.



The default value of this parameter is `rabbitmq:5672`.

6. In the **User name for AMQP connection** field, enter the user name to connect to the AMQP broker.

7. In **Password for AMQP connection**, enter the user password for connecting to the AMQP broker.

8. If you are using a secure TLS connection and a self-signed certificate is installed on the AMQP broker, add the root certificate for the AMQP broker by using the **Browse** button under the **CA certificate** setting.

A certificate can be downloaded as a DER or PEM file only.

To delete the certificate file, click the  button. To save the certificate file on your computer, click the  button.



9. If you are using a secure TLS connection and client authentication is enabled on the AMQP broker, do the following:

- a. Add the client certificate by using the **Browse** button under the **Client certificate** setting.

- b. Add the key for the client certificate by using the **Browse** button under the **Key to client certificate** setting.

It is recommended to use a certificate with a certificate key length of 4096 bits when using the RSA algorithm, or 256 bits when using the ECDH algorithm.



A certificate and certificate key can be uploaded only as a file in DER or PEM format.

To delete the certificate file or certificate key, click the  button in the corresponding field. To save the certificate file or certificate key on your computer, click the  button in the corresponding field.

10. In the **AMQP virtual host** field, specify the virtual host for establishing a connection between the AMQP Connector and the external AMQP broker.
The default value of this parameter is `/`.
11. In the **AMQP exchange point name for receiving tag values** field, specify the name of the exchange point to receive tags values from an external AMQP broker.
If a value is not defined for this parameter, tags values will not be received via the AMQP Connector.
This setting has no value by default.
12. In the **List of AMQP subscriptions for receiving tag values** field, specify the name of the list of subscriptions from which the AMQP Connector will receive tag values.
The default value of this parameter is `#`.
13. In the **AMQP queue for receiving tag values** field, specify the name of the queue for the AMQP connector.
14. In the **AMQP exchange point name for publishing messages** field, specify the name of the exchange point for sending incident registration messages.
If no value is defined for this parameter, messages will not be sent. You can specify the same name that you indicated in step 10 of these instructions.
This setting has no value by default.
15. In the **AMQP topic for publishing messages** field, specify the name of the topic where the AMQP Connector will publish messages about incident registration.
The default value of this parameter is `alert`.
16. In the **Data format** drop-down list, select the format to receive data from external systems and send messages about incidents.
The following options are available: `JSONBatch`, `Topic`, `SmartHome`, `KISG`.
The default value of this parameter is `JSONBatch`.

If you are having difficulty selecting a data format, consult Kaspersky or a certified integrator.

If none of the incident data and message formats suits you, you can contact Kaspersky Lab experts to add the required format.

17. If you have selected the `Topic` data format, add a configuration file containing the connector settings for this data format using the **Browse** button under the **Connector configuration file** setting.
To delete the connector configuration file, click the  button. To save the connector configuration file on your computer, click the  button.
18. Toggle **Scale obtained tag values** switch to enable or disable the conversion of tag values according to the **Bias** and **Multiplier** settings that were set when [creating the tag](#).
Conversion of received tag values is disabled by default.
19. Click the **Save** button.




Kaspersky MLAD will receive data and send messages about incident registration via the AMQP protocol.

Configuring the OPC UA Connector

Kaspersky MLAD uses the OPC UA Connector to receive data over a protocol described by the OPC Unified Architecture specification.

System administrators can configure the OPC UA Connector.



To configure the OPC UA Connector:

1. In the lower-left corner of the window, click .
You will be taken to the [administrator menu](#).
2. Select **System parameters** → **OPC UA Connector**.
A list of options appears on the right.
3. In the **Connection point** field, specify the connection address.
For example: `opc.tcp://10.0.0.0:8001/freeopcua/server/`.
4. In the **OPC UA server connection timeout (sec)** field, specify the time period (in seconds) that the OPC UA Connector will attempt to establish a connection with the OPC UA server.
5. Using the **Browse** button under the **Configuration file** setting, add a file containing settings for the OPC UA Connector.
To delete the connector configuration file, click the  button. To save the connector configuration file on your computer, click the  button.
6. Toggle **Scale obtained tag values** switch to enable or disable the conversion of tag values according to the **Bias** and **Multiplier** settings that were set when [creating the tag](#).
Conversion of received tag values is disabled by default.
7. Select a message encryption algorithm from the **Connection security policy** drop-down list.
The following options can be selected: **None**, **Basic256**, **Basic128Rsa15**, **Basic256Sha256**, **Aes128Sha256RsaOaep**.
8. In the **Secure messaging mode** drop-down list, select one of the following values:
 - a. If you do not want to sign or encrypt messages, select **None**.
 - b. If you want to sign messages, select **Sign messages**.
 - c. If you want to sign and encrypt messages, select **Sign and encrypt messages**.
9. In **User name**, enter the user name for connecting to the OPC UA server.
10. In **Password**, enter the password for connecting to the OPC UA server.
11. If it is necessary to use a secure connection and client authentication is enabled on the OPC UA server, do the following:
 - a. Add the client application certificate by using the **Browse** button under the **Client certificate** setting.

b. Add the private key to the client application certificate by using the **Browse** button under the **Client private key** setting.

c. In **Client private key password**, specify the password to use for unlocking the private key.

It is recommended to use a certificate with a certificate key length of 4096 bits when using the RSA algorithm, or 256 bits when using the ECDH algorithm.
A certificate and certificate key can be uploaded only as a file in DER or PEM format.

To delete the certificate file or certificate key, click the  button in the corresponding field. To save the certificate file or certificate key on your computer, click the  button in the corresponding field.

12. If it is necessary to use a secure connection and a self-signed certificate is installed on the OPC UA server, add the root certificate for the OPC UA server using the **Browse** button under the **OPC UA server CA certificate** setting.

To delete the certificate file, click the  button. To save the certificate file on your computer, click the  button.

13. In the **Historical data interval (sec)** field, specify the time interval (in seconds) for which the OPC UA Connector requests historical data stored on the OPC UA server.

Enter 0 if you do not need to download historical data. Enter -1 if you need to download all historical data.

If **Start of the historical data period** and **End of the historical data period** are set, historical data is loaded for the specified period.

14. In the **Start of the historical data period** field, select the start date and time of the period for which you want to download data from the OPC UA server.

15. In the **End of the historical data period** field, select the end date and time of the period for which you want to download data from the OPC UA server.

16. In the **Size of historical data block sent by OPC UA server (numvalues parameter)** field, specify the number of tags values that will be transmitted in the historical data block sent to the OPC UA Connector from the OPC UA server.

The specified value affects the number of iterations for sending historical data to OPC UA Connector according to the total number of observations for tags received from the OPC UA server.

17. In the **Size of historical data block sent to Message Broker service** field, specify the number of tags that will be transmitted in the historical data block sent from the OPC UA Connector to the Message Broker service.

The specified value affects the number of iterations for sending historical data to Message Broker according to the total number of observations for tags received via OPC UA Connector.

18. Click the **Save** button.

Configuring the KICS Connector

Kaspersky MLAD uses the KICS Connector to receive data from Kaspersky Industrial CyberSecurity for Networks 4.0 and later and to send back incident registration messages.

The connector for integration with Kaspersky MLAD must be created and added to Kaspersky Industrial CyberSecurity for Networks in advance. For detailed information about creating and adding a connector, please refer to the *Adding a connector* section of *Kaspersky Industrial CyberSecurity for Networks Help Guide*.

System administrators can perform integration with Kaspersky Industrial CyberSecurity for Networks version 4.0 or higher.

To configure the KICS Connector:

1. In the lower-left corner of the window, click .



You will be taken to the [administrator menu](#).

2. Select **System parameters** → **KICS Connector**.

A list of options appears on the right.

3. Using the **Browse** button under the setting **Communication data package for KICS Connector (zip)** field, add the file containing the settings for configuring interaction between Kaspersky MLAD and Kaspersky Industrial CyberSecurity for Networks.

For detailed information about creating a communication data package, please refer to the *Kaspersky Industrial CyberSecurity for Networks Help Guide*. The created communication data package must be saved on the computer where Kaspersky MLAD is installed.

If you need to delete a communication data package, click the  button in the **Communication data package for KICS Connector (zip)** field. To save the communication data package on your computer, click the  button.

4. In the **Password for KICS Connector** field, enter the password that you specified when adding the connector to Kaspersky Industrial CyberSecurity for Networks.

5. Toggle **Send messages to Kaspersky Industrial CyberSecurity for Networks** switch to enable or disable forwarding of messages about registered incidents to Kaspersky Industrial CyberSecurity for Networks.

6. In the **Tag value sampling frequency (Hz)** field, specify the frequency in Hz at which you need to receive tag values from Kaspersky Industrial CyberSecurity for Networks.

Indicate 0 in this field if [data sampling](#)  is not required.

7. Toggle **Scale obtained tag values** switch to enable or disable the conversion of tag values according to the **Bias** and **Multiplier** settings that were set when [creating the tag](#).

Conversion of received tag values is disabled by default.

8. Click the **Save** button.

Kaspersky MLAD receives data from Kaspersky Industrial CyberSecurity for Networks and sends back messages about incident registration.

Configuring the CEF Connector


Kaspersky MLAD uses the CEF Connector to receive data from external sources of events (such as the Industrial Internet of Things, network devices and applications) and to send incident registration messages to an external system.

You can also use the CEF Connector to send information security event logs of Kaspersky MLAD to an external system. Information security event logs are automatically written to the Kaspersky MLAD database.

To receive events from external sources using the CEF Connector, [configure the Event Processor service](#). Before configuring the CEF Connector settings in the Kaspersky MLAD web interface, the IP address and port number to be used for connecting the external event source to the CEF Connector must be specified in the [.env file](#). The settings of the configuration file can be changed only by a qualified technical specialist of the Customer, a Kaspersky employee, or a certified integrator.

System administrators can configure the CEF Connector.

To configure the CEF Connector:

1. In the lower-left corner of the window, click .
You will be taken to the [administrator menu](#).
2. Select **System parameters** → **CEF Connector**.
A list of options appears on the right.
3. Use the **Receive events for the Event Processor service** toggle switch to enable or disable use of the CEF Connector for receiving events from an external system.
4. Toggle **Send registered incidents to SIEM system** switch to enable or disable forwarding of messages about incidents registered by the application to the external system.
5. Toggle **Send registered events to SIEM system** switch to enable or disable forwarding of messages about events registered by Event Processor service to the external system.
6. In the **IP address for sending events and incidents to SIEM system** field, specify the IP address for connecting an external system to the CEF Connector and forwarding events processed by the Event Processor service and incidents.
7. In the **Port for sending events and incidents to SIEM system** field, specify the port number for connecting an external system to the CEF Connector and forwarding events processed by the Event Processor service and incidents.
8. If you need to send information security event logs of Kaspersky MLAD to an external system, turn on the **Send information security event logs to a Syslog server** toggle switch and do the following:
 - a. In the **Transport protocol for sending information security events to a Syslog server** drop-down list, select the protocol that you want to use for sending information security event logs.
Kaspersky MLAD supports the TCP and UDP protocols for sending information security event logs to an external system.
 - b. In the **Syslog server address for sending information security events** field, specify the IP address or host name of the external system to which the information security event logs must be sent.
 - c. In the **Syslog server port for sending information security events** field, specify the port number of the external system to which the information security event logs must be sent.
9. Use the **Use TLS connection** toggle switch to enable or disable the use of a secure TLS connection when using Kaspersky MLAD as a client.
By default, use of a secure TLS connection is enabled.

To avoid compromising the received and/or sent data, it is recommended to keep the use of a secure TLS connection enabled.

10. If you have enabled the use of a secure TLS connection, use the **Use the recommended TLS connection settings** toggle switch to enable or disable use of the recommended TLS connection settings.



When the toggle switch is on, a secure TLS connection is used via the TLS-1.2 or TLS-1.3 protocol with a cipher suite from the [list of recommended ciphers](#).

11. If you need to use a secure TLS connection for the server side of the CEF Connector, do the following:
 - a. Add the server certificate and the certificate key by using the **Browse** button under the **Server certificate** and **Private key to the server certificate** settings.

It is recommended to use a certificate with a certificate key length of at least 4096 bits when using the RSA algorithm, or 256 bits when using the ECDH algorithm.

- b. If you are using client certificates, add the root certificate to verify the signature of the client certificate by using the **Browse** button under the **CA certificate for verifying the client certificate signature** setting.

Certificates and certificate keys can be uploaded only as files in DER or PEM format.



To delete the certificate file or certificate key, click the  button in the corresponding field. To save the certificate file or certificate key on your computer, click the  button in the corresponding field.

12. If you need to use a secure TLS connection for the server side of the CEF Connector, do the following (if necessary):
 - a. Add the client certificate and the certificate key by using the **Browse** button under the **Client certificate** and **Private key to the client certificate** settings.

It is recommended to use a certificate with a certificate key length of at least 4096 bits when using the RSA algorithm, or 256 bits when using the ECDH algorithm.

- b. If you are using server certificates, add the root certificate to verify the signature of the server certificate by using the **Browse** button under the **CA certificate for verifying the server certificate signature** setting.

Certificates and certificate keys can be uploaded only as files in DER or PEM format.

To delete the certificate file or certificate key, click the  button in the corresponding field. To save the certificate file or certificate key on your computer, click the  button in the corresponding field.

13. Click the **Save** button.

Configuring the WebSocket Connector

Kaspersky MLAD uses the WebSocket Connector to receive data and send messages about incident registration via the WebSocket protocol.

System administrators can configure the WebSocket Connector. The instructions in this section are provided for information purposes.

To configure the WebSocket Connector:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).



2. Select **System parameters** → **WebSocket Connector**.

A list of options appears on the right.

3. In the **WebSocket server URL address** field, specify the web address of the WebSocket server that the WebSocket Connector will interact with.

Enter the web address in the format: `WebSocket protocol://address:port/`.



4. If it is necessary to use a secure connection and a self-signed certificate is installed on the WebSocket server, add the root certificate for the WebSocket server using the **Browse** button under the **CA certificate** setting.

To delete the certificate file, click the  button. To save the certificate file on your computer, click the  button.

5. If it is necessary to use a secure connection and client authentication is enabled on the WebSocket server, do the following:

- a. Add the WebSocket client application certificate by using the **Browse** button under the **Client certificate** setting.
- b. Add the key to the WebSocket client application certificate by using the **Browse** button under the **Key to client certificate** setting.

It is recommended to use a certificate with a certificate key length of 4096 bits when using the RSA algorithm, or 256 bits when using the ECDH algorithm.

To delete the certificate file or certificate key, click the  button in the corresponding field. To save the certificate file or certificate key on your computer, click the  button in the corresponding field.

6. In the **Data format** drop-down list, select the format to receive data from external systems and send messages about incidents.

The following options are available: JSONBatch, Topic, SmartHome, KISG.

The default value of this parameter is JSONBatch.

If you are having difficulty selecting a data format, consult Kaspersky or a certified integrator.

If none of the incident data and message formats suits you, you can contact Kaspersky Lab experts to add the required format.

7. If you have selected the Topic data format, add a configuration file containing the connector settings for this data format using the **Browse** button under the **Connector configuration file** setting.

To delete the connector configuration file, click the  button. To save the connector configuration file on your computer, click the  button.

8. Toggle **Scale obtained tag values** switch to enable or disable the conversion of tag values according to the **Bias** and **Multiplier** settings that were set when [creating the tag](#).

Conversion of received tag values is disabled by default.

9. Toggle **Submit incidents** switch to enable or disable the forwarding of messages about incidents registered in Kaspersky MLAD to a WebSocket server.

10. If you are using a secure TLS connection, use the **Use the recommended TLS connection settings** toggle switch to enable or disable use of the recommended TLS connection settings.

By default, use of the recommended TLS connection settings is enabled.

When the toggle switch is on, a secure TLS connection is used via the TLS-1.2 or TLS-1.3 protocol with a cipher suite from the [list of recommended ciphers](#).

11. Click the **Save** button.

Kaspersky MLAD will receive data and send messages about incident registration via the WebSocket protocol.

Configuring the Event Processor service

Kaspersky MLAD uses the Event Processor service to identify patterns and anomalous sequences of events and patterns. You can configure the settings of the Event Processor service.

If Kaspersky MLAD is restarted, you do not need to re-configure the Event Processor service settings. Kaspersky MLAD restores the Event Processor service state from the database or file in bit format. This restoration process may take several minutes if there is a significantly large number of processed events or registered patterns. Until the state of the Event Processor service is restored in the **Event Processor** section, requests will not be fulfilled, data will not be updated, and data received from the CEF Connector will not be processed. This data is temporarily stored in the system message queue and is processed after the state of the Event Processor service is restored.

The Event Processor service may require a large amount of RAM on the server where Kaspersky MLAD is installed. The amount of RAM usage depends on the rate of the event stream and the volume of events history that is processed. The specific configuration of the Event Processor service also has an effect on the amount of RAM usage.

System administrators can configure the Event Processor service.

To configure the Event Processor service:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).



2. Select **System parameters** → **Event Processor**.

A list of service settings appears on the right.

3. In the **Online mode** section, do the following:

- a. Using the **Browse** button under the setting **Event processor configuration file** field, add the file containing the configuration settings for the Event Processor service.

The [Configuration file](#) is created by a qualified technical specialist of the Customer, a Kaspersky Lab employee or a certified integrator.

If you need to delete the configuration file for the Event Processor service, click the  button. To save the configuration file on your computer, click the  button.

Changing the configuration file of the Event Processor service results in a complete loss of the service's data.

b. If you need to process incidents registered by the Anomaly Detector service, turn on the **Process incidents as events** toggle switch.

c. In the **Maximum number of network layers** field, specify the number of layers of the semantic neural network that will be used.

The default number of network layers for event data that is based on a specific structure is ten layers. In most cases, ten layers are enough for the hierarchical presentation of data in the semantic neural network at the core of the Event Processor. To identify patterns of periodic processes that span an extended period of time, you may need to increase the value of the **Maximum number of network layers** parameter.

d. In the **Coefficient defining the permitted dispersion of the pattern duration** field, specify the coefficient used to determine the permissible dispersion of intervals between elements in the same pattern.

If the actual dispersion value is less than or equal to one that is specified, the identified sequences of events will be registered as one pattern.

e. In the **Interval for receiving batch events (sec.)** field, specify the time interval (in seconds) for which the Event Processor service forms an episode from incoming events received for processing.

If the rate of incoming events is approximately 1000 events per second, it is recommended to indicate this value as the interval for receiving new events so that you receive a number of events close to the value indicated in the **Batch size in online mode (number of events)** field during the specified period. If the rate of incoming events is a lot lower than this value, you should adjust the interval for receiving new events to ensure an optimal frequency of event processing.

f. In the **Batch size in online mode (number of events)** field, specify the maximum number of events per episode to be subsequently processed by the Event Processor service.

If the rate of incoming events is approximately 1000 events per second, it is recommended to indicate a value equal to 4096 in this field.

g. In the **Method of saving the state of the Event Processor service** drop-down list, select one of the following options for saving the Event Processor service state:

- **Database table** – Kaspersky MLAD saves the results from processing each episode in the database table.
- **File in bit format** – Kaspersky MLAD saves the state of the Event Processor service according to the frequency defined in the **Component backup frequency** field. The application saves the state of the service to the file specified in the **File containing a backup copy of the component state** field.

Saving the Event Processor service state to a file in bit format is recommended for debugging and configuring the application settings by Kaspersky employees during the deployment of Kaspersky MLAD.



By default, the Event Processor service saves the results of event stream processing in a database table.

Changing the way of saving the Event Processor service state results in a complete loss of the service's data.

h. If you select to store the Event Processor service state in a file in bit format, in the **Component backup frequency** field, specify how often (in days, hours, and minutes) to perform a backup of the Event Processor service.

i. If you chose to store the status of the Event Processor service as a bitmap file, add the file that contains a backup copy of the Event Processor service via the **Browse** button under the **File containing a backup copy of the component state** setting.

This file will be used if you ever need to restore the state of the Event Processor service. The state of the Event Processor service can be restored by Kaspersky experts as part of their extended technical support.

If you need to delete the file containing the backup copy of the Event Processor service, click the  button. To save the file containing a backup copy of the service on your computer, click the  button.

4. In the **Sleep mode** section, do the following:

a. In the **Batch size in sleep mode (number of events)** field, specify the number of events for forming an episode in sleep mode.

The Event Processor service generates episodes based on the history of events received for reprocessing during the time interval specified in the **Events history interval for processing in sleep mode** field.

b. In the **Send alerts when the monitor is activated in sleep mode** field, select one of the following values:

- **Send alerts when the monitor is activated by any pattern** – Kaspersky MLAD sends alerts when the monitor is activated in the sleep mode if the patterns are detected in accordance with the specified monitoring criteria. The number of monitor activations is refreshed in the **Event Processor** section on the **Monitoring** tab.
- **Do not send alerts when the monitor is activated** – Kaspersky MLAD does not send alerts when the monitor is activated in the sleep mode.
- **Send alerts when the monitor is activated by a new pattern** – Kaspersky MLAD sends alerts when the monitor is activated in the sleep mode if new patterns are detected in accordance with the specified monitoring criteria. The number of monitor activations is refreshed in the **Event Processor** section on the **Monitoring** tab.
- **Send alerts when the monitor is activated by a previously registered pattern** – Kaspersky MLAD sends alerts when the monitor is activated in the sleep mode if stable patterns are detected in accordance with the specified monitoring criteria. The number of monitor activations is refreshed in the **Event Processor** section on the **Monitoring** tab.

c. In the **Sleep mode frequency** field, specify how often (in days) and at what time (according to the UTC standard) the Event Processor service goes to the sleep mode to reprocess events.

It is recommended to specify the time when the event stream is the least intensive as the start time for the sleep mode.

If the specified sleep time has not yet come on the current day, the Event Processor will go to the sleep mode on that day. If the sleep time has already been missed on the current day, the Event Processor will go to the sleep mode at the specified time after the specified number of days.

d. In the **Sleep mode duration (HH:MM)** field, specify the time period (in hours and minutes) during which the Event Processor service processes events in the sleep mode.

- e. In the **Events history interval for processing in sleep mode** field, specify the time interval (in days, hours, and minutes) during which the analyzed events must be forwarded for reprocessing in the sleep mode to the Event Processor service.

5. Click the **Save** button.

Configuring the statuses and causes of incidents

Kaspersky MLAD lets you specify the causes of incidents and the statuses of incidents and groups of incidents.

The status of an incident or a group of incidents is a mark about the status of incident analysis performed by an expert. After installation of Kaspersky MLAD, the following statuses of incidents and incident groups are available by default: **Under review**, **Decision pending**, **Instructions issued**, **Problem closed**, **Cause unknown**, **Ignore** and **False positive**. For the **Problem closed** and **Ignore** statuses, the **Notify about an incident** check box is cleared by default. If during registration, incidents are automatically assigned one of these statuses, no email alerts will be sent, and no incident dot indicators will be displayed under **Monitoring** or **History**. An incident status can be assigned automatically in one of the following cases:

- If the incident was automatically assigned to a group with that status.
- If the incident is registered by an ML model element that sets that incident status by default.

The incident cause is a mark of the cause of the incident [added by an expert](#) based on the results of the incident analysis.

You can add causes and statuses for incidents. The created causes and statuses of incidents will become available for selection in the [Incidents](#) section. You can also change and delete statuses and causes of incidents.

System administrators can configure the causes and statuses of incidents.

To add an incident status:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **System parameters** → **Incidents** section.

3. In the **Statuses of incidents** section, click the **Create** button.

The **Create element** pane appears on the right.

4. In the **Value, in Russian** field, specify the name of the incident status in Russian.

5. In the **Value, in English** field, specify the name of the incident status in English.

6. In the **Sort** field, indicate the sequence number for which the incident status will be sorted in the **Incident status** drop-down list in the **Incidents** section.

The statuses of incidents will be sorted by their names if the sequence numbers of incident statuses coincide.

7. To send incident registration notifications together with the added status and display its indicator in the prediction error subsection of the **Monitoring** and **History** sections, select the **Notify about an incident** check box.

If you remove the checkbox, no email alerts about incidents that receive the status automatically will be sent, and no incident dot indicators will be displayed under **Monitoring** or **History**.

8. Click the **Save** button.

To add a cause for incidents:

1. In the [administrator menu](#), select **System parameters** → **Incidents**.

2. In the **Causes of incidents** section, click the **Create** button.

The **Create element** pane appears on the right.

3. In the **Incident cause** field, specify the name of the incident cause.

4. In the **Sort** field, indicate the sequence number for which the incident cause will be sorted in the **Incident cause** drop-down list in the **Incidents** section.

The causes of incidents will be sorted by their names if the sequence numbers of incident causes coincide.

5. Click the **Save** button.

To change the statuses or causes of incidents:

1. In the [administrator menu](#), select **System parameters** → **Incidents**.

2. To change the parameters of incidents, do one of the following:

- If you need to change the statuses of incidents or groups of incidents, use the **Statuses of incidents** settings group to select one or more incident statuses and click the **Edit** button.
- If you need to change the causes of incidents, use the **Causes of incidents** settings group to select one or more incident causes and click the **Edit** button.

3. Make the necessary changes.

4. Click the **Save** button.

To remove statuses or causes of incidents:

1. In the [administrator menu](#), select **System parameters** → **Incidents**.

2. To remove parameters of incidents, do one of the following:

- If you need to delete the statuses of incidents or groups of incidents, use the **Statuses of incidents** settings group to select one or more incident statuses and click the **Delete** button.
- If you need to delete the causes of incidents, use the **Causes of incidents** settings group to select one or more incident causes and click the **Delete** button.

3. In the window that opens, confirm the deletion.

Kaspersky MLAD will remove information about the incident statuses and causes from the corresponding tables and will remove them from the information about incidents and incident groups in the [Incidents](#) section for which these incident causes or statuses were selected.

Configuring logging for Kaspersky MLAD services

You can configure the log level for Kaspersky MLAD services to write specific information about the state of the application and display it in the logging system (Grafana). To view how Kaspersky MLAD services are mapped to the names of Docker containers and images, see the [Appendix](#).

System administrators can configure the logging of Kaspersky MLAD services.

To configure the log levels of Kaspersky MLAD services:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **System parameters** → **Logging** section.

The list of Kaspersky MLAD services will be displayed on the right.

3. If necessary, use the drop-down list next to the name of the relevant service to change the log level of the service.

The following log levels are available in Kaspersky MLAD:

- **Debug** – log all information in the application.
- **Info** – log basic information about application operations.
- **General** – log important information about application operations.
- **Warning** – log errors that occur during operation of the application and log events that could lead to errors in application operations.
- **Error** – log errors that occur in application operations.
- **Critical** – log critical errors that occur in application operations.

The **General** log level is used for most services by default. The **Info** log level is used for the API Server service by default.

4. Click the **Save** button.

Configuring time intervals for displaying data

Kaspersky MLAD lets you specify the time interval (scale) for displaying data on graphs in the [Monitoring](#), [History](#) and [Time slice](#) sections. After installation of Kaspersky MLAD, the following time intervals are available by default:

- 1, 5, 10, 15, and 30 minutes
- 1, 3, 6, and 12 hours
- 1, 2, 15, and 30 days

- 3 and 6 months
- 1, 2, and 3 years

You can add time intervals for displaying data on graphs. The created time intervals will become available for selection in the **Monitoring**, **History** and **Time slice** sections. You can also edit and delete time intervals.

System administrators can configure the time intervals for displaying data on charts.

To add a time interval for displaying data:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **System parameters** → **Graphs** section.

3. In the **Time intervals** settings group, click the **Create** button.

The **Create element** pane appears on the right.

4. In the **Time interval (sec.)** field, specify the time interval for which you want to display data on graphs.

When a time interval is entered, Kaspersky MLAD automatically breaks down the time interval into specific units of time (years, months, weeks, days, hours, minutes, and seconds) in the **Value, in Russian** and **Value, in English** fields.

5. If necessary, change the Russian name of the time interval in the **Value, in Russian** field.

6. If necessary, change the English name of the time interval in the **Value, in English** field.

7. In the **Sort** field, indicate the sequence number for which the time interval will be sorted in the drop-down lists in the **Monitoring**, **History** and **Time slice** section.

8. Click the **Save** button.

To change the time intervals for displaying data:

1. In the [administrator menu](#), select **System parameters** → **Graphs**.

2. In the **Time intervals** settings group, select one or more time intervals and click the **Edit** button.

3. Make the necessary changes.

4. Click the **Save** button.

To delete time intervals for displaying data:

1. In the [administrator menu](#), select **System parameters** → **Graphs**.

2. In the **Time intervals** settings group, select one or more time intervals and click the **Delete** button.


3. In the window that opens, confirm the deletion.

Information about the time intervals will be deleted from the table.

Configuring how the Kaspersky MLAD menu items are displayed

System administrators can configure settings for displaying the menu items of Kaspersky MLAD.

To configure the way Kaspersky MLAD menu items are displayed:

1. In the lower-left corner of the window, click .
You will be taken to the [administrator menu](#).
2. On the opened page, in the menu on the left, select **System parameters** → **Menu**.
A list of options appears on the right.
3. In the **Availability of main menu items** settings group, use the toggle switch to enable or disable the display of a specific section in the main menu.
4. In the **Availability of administrator menu items** settings group, use the toggle switch to enable or disable the display of a specific section in the administrator menu.
5. Click the **Save** button.


Export and import of Kaspersky MLAD settings

Kaspersky MLAD allows you to export and import configuration files that contain the settings of application services and connectors, as well as security settings, application service logging levels, settings for displaying the application menu and settings for typical statuses and causes of incidents, which are configured through the web interface. This could substantially reduce the time spent on configuring Kaspersky MLAD if you have to re-deploy the application.

When exporting settings, Kaspersky MLAD does not save to the archive file the passwords specified in the **System parameters** section, as well as the certificate files and certificate file keys uploaded in this section.

Only system administrators are allowed to export and import configuration files for Kaspersky MLAD services.

To export configuration files from Kaspersky MLAD:

1. In the lower-left corner of the window, click .
You will be taken to the [administrator menu](#).
2. Select **System parameters**.
3. Click the **Export** button in the upper part of the opened page.

Kaspersky MLAD configuration files will be saved in an archive named mlad-settings.tar.gz on the local computer.

To upload configuration files to Kaspersky MLAD:

1. In the [administrator menu](#), select **System parameters**.
2. Click the **Import** button in the upper part of the opened page.
3. In the opened window, select the archive file containing the necessary configuration of Kaspersky MLAD parameters.



Kaspersky MLAD configuration files will be uploaded to the application.

Managing assets and tags

System administrators can manage assets and tags.

Assets and tags are the primary elements of the [monitored asset hierarchical structure](#). The hierarchical structure is displayed as an asset tree.

Observations of the monitored asset are transmitted to Kaspersky MLAD as tags. Based on the obtained tag values you can perform training and inference of ML models.

In the **Assets** section of the [administrator menu](#), you can view [assets](#) and [tags](#) that have been created or uploaded to Kaspersky MLAD. Using the  and  buttons to the left of the asset names, you can display or hide the asset tree data. You can [create assets](#) and [tags](#), and [edit tag settings](#), such as tag blocking thresholds.

Kaspersky MLAD uses connectors to [receive telemetry data according to tags](#) from devices of the monitored asset. Tag values, defined within the asset tree and fed into the application, along with their corresponding tag IDs, are stored in [Time Series Database](#). When [saving of all tags values is enabled](#), the Time Series Database service also saves IDs and values of unknown tags (not listed in the asset tree). You can [compare the current asset tree structure with the structure in the Time Series Database service](#) and add missing tags to the asset tree, if necessary.

If Kaspersky MLAD detects unknown tags received from external devices through [KICS Connector](#), these tags will be automatically created in the **KICS** section of the asset tree. The application automatically assigns IDs to tags and fills in the following information received from Kaspersky Industrial CyberSecurity for Networks:

- IDs of the tags
- Names of the tags
- Descriptions of the tags
- Units of measure for the tags
- Names of the assets for which the tags are received

Kaspersky MLAD is compatible with Kaspersky Industrial CyberSecurity for Networks version 4.0 and later.

You can also [delete existing tags](#), [import tags and assets from an XLSX file](#), or [export them to a an XLSX file](#).

About monitored asset hierarchical structure

Monitored asset hierarchical structure (hierarchical structure) is a way of representing the monitored asset as a tree, where the leaf nodes correspond to the [tags](#) from which telemetry data is received.

Monitored asset tags are organized as a hierarchy of assets representing, for example, units, plants, shops, and factories. The specific way assets are organized, their types and quantity are decided by the customer and depend on the structure of the asset being monitored. Each asset has only one parent element. This element can be another asset (parent asset) or a head element of the hierarchical structure (*Root*) that corresponds to the monitored asset as a whole.

Tags and assets are the primary elements of a hierarchical structure. You can [import](#) or [export an asset tree](#) as an [XLSX file](#), and create and manage them in the [Assets](#) section.

In addition to primary elements, the following functional elements can be added to the hierarchical structure in the process of or as a result of the operations of Kaspersky MLAD:

- [Markups](#)
- [ML models](#)
- [Templates for ML models](#)

About tags

Tags are the main objects of observation in Kaspersky MLAD. A *tag* is a process parameter transmitted in the industrial network (temperature, for example). Measurements of physical parameters, as well as setpoints, commands, or states of control systems can be transmitted as tags. The values of tags are transmitted and received by the assets over specific protocols. The values of tags are displayed on graphs in the **History** and **Monitoring** sections and are also used to detect incidents.

Kaspersky MLAD supports several methods for obtaining telemetry data (tags). Depending on the monitored asset attributes and the tag transmission capabilities, you can select one of the following methods for receiving tag values in real time:

- Use the connectors of Kaspersky Industrial CyberSecurity for Networks that analyze mirrored traffic and send tags to Kaspersky MLAD in online mode. Kaspersky MLAD sends back information about detected incidents.
- Use the **OPC UA Connector** if the monitored asset provides the capability to [transmit tags from ICS over the OPC UA protocol](#) in the online mode.
- Use the **MQTT Connector** if the monitored asset provides the capability to [transmit tags over the MQTT protocol and receive messages about incident registration](#) in the online mode.
- Use the **AMQP Connector** if the monitored asset has the capability to [transmit tags over the AMQP protocol and receive messages about incident registration](#) in online mode.
- Use the **WebSocket Connector** if the monitored asset provides the capability to [transmit tags over the WebSocket protocol and receive messages about incident registration](#) in the online mode.
- Use the **CEF Connector** if the monitored asset provides the capability to [transmit tags using the CEF Connector technology and receive messages about incident registration](#) in the online mode.
- If the above methods of tag transmission are not available, you can write a tag export script for using the [HTTP Connector](#) to configure a periodic export of tags as CSV files over HTTP or HTTPS (for example, once per hour or once per minute).

You can also retrieve tag values for a specific historical period with one of the following methods:

- Using **OPC UA Connector** if the monitored asset supports access to historical data according to the OPC UA HDA standard
- Using **HTTP Connector** by uploading CSV files containing historical data

Create asset

System administrators can manage assets and tags.

In Kaspersky MLAD, you can create assets and categorize tags by asset as you see fit. For example, you can create assets to align with the production structure of the monitored asset, right down to the devices that send telemetry data.

To create a new asset:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Assets** section.

3. In the upper part of the page, click the **Create** button.

4. In the panel that opens, in the **Element type** drop-down list, select the **Asset** value.

5. If necessary, click the **Choose icon** button and select an icon for the asset in the opened window.

You can upload an asset icon by clicking the **Load icon** button. Images of any format larger than 128x128 pixels are shrunk to 128x128 while maintaining the aspect ratio. The size of the uploaded image in SVG format must not exceed 200 KB.

If you need to delete the asset icon, click the tag icon and then click **Delete** in the opened window.

6. In the **Asset** drop-down list, select the section of the asset tree within which you want to create the asset.

7. Specify the asset name in the **Name** field.

Asset names must be unique within a single asset tree section.

8. Enter a description for the asset in the **Description** field.

9. Select the type of asset from the **Asset type** drop-down list.

If you have [uploaded the configuration of assets and tags](#) to Kaspersky MLAD, you can select one of the asset types defined in the configuration file. Asset types are specified in the **directory_types** tab of the [configuration file](#). If you have not loaded an asset and tag configuration, select **System asset**.

10. If you have selected one of the asset types defined in the imported configuration file, specify values for the special parameters of the assets, if necessary.

The names of special parameters are specified in the **directory_types** tab of the configuration file.

11. Click the **Save** button.

The asset will be created. If necessary, you can [change the position of an asset](#) in the tree.

You can also create nested assets using the asset tree. To do that, next to the name of the section to which you want to add the asset, open the vertical menu **...** and select **Add asset**.

Change asset settings

System administrators can manage assets and tags.

You can edit the settings of previously created assets.

To edit the settings of an asset in the asset tree:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Assets** section.

3. In the asset tree, next to the name of the asset whose settings you want to change, open the vertical menu **...** and select **Edit asset**.

The **Edit asset** pane opens on the right.

4. Adjust the following settings in the panel that opens, if needed:

- Asset icon
- The section of the asset tree you want to assign the asset to
The asset subsections and their tags are moved to the new asset.
- Asset name.
- Asset description
- Asset type
- Values for custom asset settings

5. Click the **Save** button.


The asset will be modified. If necessary, you can [change the position of an asset](#) in the tree.

Create tag

System administrators can manage assets and tags.

In Kaspersky MLAD, you can create new tags to describe data received from the monitored asset (source tags) or from the Stream Processor service.

To create a new tag:

1. In the lower-left corner of the window, click .
You will be taken to the [administrator menu](#).
2. Select the **Assets** section.
3. In the upper part of the page, click the **Create** button.
4. In the panel that opens inside the **Element type** drop-down list, select **Tag**.
5. If necessary, click the **Choose icon** button and select an icon for the tag in the opened window.
You can upload the tag icon by clicking the **Load icon** button. Images of any format larger than 128x128 pixels are shrunk to 128x128 while maintaining the aspect ratio. The size of the uploaded image in SVG format must not exceed 200 KB.
If you need to delete the tag icon, click the tag icon and then click **Delete** in the opened window.
6. In the **Asset** drop-down list, select the section of the asset tree to which you want to assign the created tag.
Assets in the asset tree must be [preloaded](#) or [created manually](#).
7. Specify the unique tag name in the **Name** field. If you want to receive tag values from an external system, specify the tag name in the external system.
8. If necessary, enter a description for the tag in the **Description** field.
9. If necessary, specify an alternative name for the tag in the **Alternative name** field.
10. Enter the unique numerical identifier of the tag in the **ID** field.
You can specify a value in the range of 1 to 1,000,000.
11. If necessary, in the **Dimension** field, specify the measurement units for the tag (for example % or mPa).
12. Optionally, in the **Lower** and **Upper** fields under **Blocking threshold**, enter the lower and upper thresholds for acceptable tag values.
These settings are required for the Limit Detector to operate correctly. Whenever the tag value reaches its upper or lower blocking threshold, the Limit Detector registers an incident.
If [Always display blocking threshold](#) is enabled, the vertical scale of the graph will be defined by threshold lines drawn at the lower and upper boundaries of the [graphic area](#) that the tag belongs to, provided that the tag values are within the specified range. If the tag values go beyond the specified thresholds, the vertical scale will be automatically changed to display the tag values exceeding the limits.
13. If necessary, in the **Alarm threshold**, in the **Lower** and **Upper** fields, specify the lower and upper thresholds of the tag values, upon reaching which it is necessary for the ICS to take emergency response measures.
14. Optionally, enter the minimum and maximum physically possible tag values in the **Lower** and **Upper** fields under **Measurement confidence thresholds**. Be sure to consider any limitations of the measuring equipment.
15. Optionally, in the **Bias** and **Multiplier** fields under **Value conversion**, enter the value to add to the tag value and the value to multiply the tag value obtained via connectors by.
The application convert the tag values according to the formula: $a * x + b$, where:
 - a is the value of the multiplier specified in **Multiplier**.
 - x : the received tag value

- **b** is the value of the offset specified in **Bias**.

The application performs conversion when **Scale obtained tag values** in the connector settings is enabled. Value conversion is possible for tags obtained with the help of [MQTT Connector](#), [AMQP Connector](#), [OPC UA Connector](#), [WebSocket Connector](#), or [KICS Connector](#).

Value conversion is required when obtaining tag values in a unit of measurement that differs from the one specified in the **Dimension** field.

16. If the application receives tag values via KICS Connector, optionally, in the **External system asset** field, specify the name of the device created in the external system that you want to receive tags for.
17. If necessary, in the **X**, **Y**, and **Z** fields, specify the spatial coordinates for the location of the monitored asset's sensor.
You can use an arbitrary point as the origin of the coordinate system.
You can use sensor coordinates to calculate tag values when [creating a preset](#) and displaying them on the graph in the **Time slice** section.
18. If necessary, in the **Comment** field, enter a brief comment for the tag.
19. Click the **Save** button.


The new tag appears in the **Tags** group of the asset tree. The **Tags** group is created automatically and displayed as part of the selected section of the asset tree. If necessary, you can [change the position of tags](#) in the tree.

Adding a tag to an asset

System administrators can manage assets and tags.

In Kaspersky MLAD, you can add tags to created assets.

To add a tag to an asset:

1. In the lower-left corner of the window, click .
- You will be taken to the [administrator menu](#).
2. Select the **Assets** section.
3. In the asset tree, next to the section to which you want to add a tag, open the vertical menu **...** and select **Add tag**.
The **Create tag** pane opens on the right.
4. If necessary, click the **Choose icon** button and select an icon for the tag in the opened window.
You can upload the tag icon by clicking the **Load icon** button. Images of any format larger than 128x128 pixels are shrunk to 128x128 while maintaining the aspect ratio. The size of the uploaded image in SVG format must not exceed 200 KB.
If you need to delete the tag icon, click the tag icon and then click **Delete** in the opened window.
5. Specify the unique tag name in the **Name** field. If you want to receive tag values from an external system, specify the tag name in the external system.
6. If necessary, enter a description for the tag in the **Description** field.

7. If necessary, specify an alternative name for the tag in the **Alternative name** field.

8. Enter the unique numerical identifier of the tag in the **ID** field.

9. If necessary, in the **Dimension** field, specify the measurement units for the tag (for example % or mPa).

10. Optionally, in the **Lower** and **Upper** fields under **Blocking threshold**, enter the lower and upper thresholds for acceptable tag values.

These settings are required for the Limit Detector to operate correctly. Whenever the tag value reaches its upper or lower blocking threshold, the Limit Detector registers an incident.

If [Always display blocking threshold](#) is enabled, the vertical scale of the graph will be defined by threshold lines drawn at the lower and upper boundaries of the [graphic area](#) that the tag belongs to, provided that the tag values are within the specified range. If the tag values go beyond the specified thresholds, the vertical scale will be automatically changed to display the tag values exceeding the limits.

11. If necessary, in the **Alarm threshold**, in the **Lower** and **Upper** fields, specify the lower and upper thresholds of the tag values, upon reaching which it is necessary for the ICS to take emergency response measures.

12. Optionally, enter the minimum and maximum physically possible tag values in the **Lower** and **Upper** fields under **Measurement confidence thresholds**. Be sure to consider any limitations of the measuring equipment.

13. Optionally, in the **Bias** and **Multiplier** fields under **Value conversion**, enter the value to add to the tag value and the value to multiply the tag value obtained via connectors by.

The application convert the tag values according to the formula: $a * x + b$, where:

- a is the value of the multiplier specified in **Multiplier**.
- x : the received tag value
- b is the value of the offset specified in **Bias**.

The application performs conversion when **Scale obtained tag values** in the connector settings is enabled. Value conversion is possible for tags obtained with the help of [MQTT Connector](#), [AMQP Connector](#), [OPC UA Connector](#), [WebSocket Connector](#), or [KICS Connector](#).

Value conversion is required when obtaining tag values in a unit of measurement that differs from the one specified in the **Dimension** field.

14. If the application receives tag values via KICS Connector, optionally, in the **External system asset** field, specify the name of the device created in the external system that you want to receive tags for.

15. If necessary, in the **X**, **Y**, and **Z** fields, specify the spatial coordinates for the location of the monitored asset's sensor.

You can use an arbitrary point as the origin of the coordinate system.

You can use sensor coordinates to calculate tag values when [creating a preset](#) and displaying them on the graph in the **Time slice** section.

16. If necessary, in the **Comment** field, enter a brief comment for the tag.

17. Click the **Save** button.

The new tag appears in the **Tags** group of the asset tree. The **Tags** group is created automatically and displayed as part of the selected section of the asset tree. If necessary, you can [change the position of tags](#) in the tree.

Editing a tag

System administrators can manage assets and tags.

You can edit previously created tags.

To edit a tag:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Assets** section.

3. In the asset tree, next to the name of the tag that you want to change, open the vertical menu **...** and select **Edit tag**.

You can show or hide the data in the asset tree by using the  and  buttons to the left of the asset names.

The **Edit tag** pane opens on the right. In the upper part of the pane that opens, the number of ML models that use the selected tag is displayed.

4. Adjust the tag settings, if needed. For a description of the settings, see the [instructions on creating a tag](#).

Kaspersky MLAD periodically verifies information about tags received from Kaspersky Industrial CyberSecurity for Networks. If a tag name and/or information about the tag asset were changed manually, the application automatically updates the tag name and/or information about the asset according to the tag and asset names in Kaspersky Industrial CyberSecurity for Networks after the next scan.

5. Click the **Save** button.

If necessary, you can [change the position of tags](#) in the tree.

Moving assets and tags

System administrators can manage assets and tags.

You can move assets and/or tags within the asset tree. All assets and tags that are part of the selected asset will be moved.

To move an asset and/or tag:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Assets** section.

3. In the asset tree, select the check boxes next to the names of the assets and/or tags that you want to move.

4. In the upper part of the page, click the **Move** button.

5. In the panel that opens, in the **Asset** drop-down list, select the asset to which you want to transfer the selected assets and/or tags.

6. Click the **Save** button.

The modified asset tree appears in the **Assets** section.

You can also change the location of assets and tags in the tree using the dots (::) to the left of the name of the required asset or tag. To do this, click and hold the dots (::) to the left of the relevant asset or tag and drag the relevant asset or tag up or down in the tree.

Deleting an asset or tag

System administrators can manage assets and tags.

You can delete assets and/or tags from the asset tree if the selected tags or tags associated with the selected asset are not used by ML models.

Removing a tag from the application does not remove it from the monitored asset. Kaspersky MLAD will continue to receive telemetry data on the tag and, if [saving of all tags values is enabled](#), the data will be stored in [Time Series Database](#).

The tag must be removed from the connector configuration file before deletion.

To delete a tag:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Assets** section.

3. Perform one of the following actions:

- In the asset tree, select the check box next to the name of the tag that you want to delete and click the **Delete** button at the top of the page.
- In the vertical menu ... to the right of the relevant tag, select **Delete tag**.

4. In the window that opens, confirm the deletion of the tag.

To delete an asset:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Assets** section.

3. Perform one of the following actions:

- In the asset tree, select the check box next to the name of the asset that you want to delete and click the **Delete** button at the top of the page.
- In the vertical menu ... to the right of the relevant asset, select **Delete asset**.

4. In the window that opens, confirm the deletion of the asset.

To remove one or more assets and/or tags:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).


2. Select the **Assets** section.


3. In the asset tree, select the check boxes next to the names of the assets and/or tags.

If you need to remove one or more tags from an asset, expand the corresponding section of the asset tree by clicking the plus sign (+), and select the relevant tags.

4. Click the **Delete** button in the upper part of the page.

5. In the window that opens, confirm the removal of assets and/or tags.

If the selected assets and/or tags are not used by ML models, the  icon is displayed in the window opposite the line **Checking links between tags and added models**. The selected tags will be permanently deleted from Kaspersky MLAD.

If the selected assets and/or tags are used by ML models, the  icon is displayed in the window opposite the line **Checking links between tags and added models**. In this case, you cannot delete the selected assets and/or tags. To delete assets and/or tags, you must [delete the ML models](#) in which they are used.

Checking the current structure of tags

System administrators can manage assets and tags.

Kaspersky MLAD receives telemetry data from monitored devices via connectors and stores tag IDs and values as defined in the asset tree, in [Time Series Database](#). When [saving of all tags values is enabled](#), the Time Series Database service also saves IDs and values of unknown tags (not listed in the asset tree).

Kaspersky MLAD allows you to compare the current tag structure displayed in the asset tree and used for a monitored asset to the one saved for this monitored asset in the Time Series Database service. Kaspersky MLAD detects tags that were received from external assets, but are missing in the current tag structure and are not used for the monitored asset. If necessary, you can add these tags to the current tag structure.

When receiving telemetry data on unknown tags via KICS Connector, the application also automatically creates these tags in the **kics** asset tree section.

To compare the current tag structure with the structure in the Time Series Database service:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Assets** section.

3. In the upper part of the page, click the **Check tags** button.

The current tag structure used for the monitored asset is compared with the tag structure stored in the Time Series Database service. The comparison result is displayed in the upper part of the page.

If missing tags are detected, Kaspersky MLAD displays a list of these tags with the names in the **Tag <tag ID>** format.

4. To add missing tags, do the following:

- a. For each detected tag, in the **Asset** field select the asset to which you want to assign the tag.
- b. Click the **Add** button.

Kaspersky MLAD will add tags to the asset tree. Only the IDs, names in the **Tag <tag ID>** format, and the assets to which the tags are assigned are specified for these tags. If necessary, you can [change the added tags](#).

Uploading tag and asset configuration to the system

Tag and asset configuration is created while deploying Kaspersky MLAD and building an ML model. Tag and asset configuration is provided in [XLSX file format](#).

System administrators can manage assets and tags.

To upload tag and asset configuration to Kaspersky MLAD:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Assets** section.

3. Click the **Import** button.

The **Hierarchical structure import** pane opens on the right.

4. In the **File import** field, add an XLSX file containing the required configuration of assets and hierarchical structure tags.

If you need to edit the asset and tag configuration file, click  and select a new file.

5. In the **Asset** drop-down list, select the section of the asset tree to which you want to load the configuration of assets and tags from a file.

For configuration files with top-level assets, designate the **Root** of the hierarchy as the target.

6. In the **Import mode** drop-down list, select one of the following values:

- **Add and update.** Kaspersky MLAD will add new assets and tags from the configuration file and update information about previously created and/or imported assets and tags within the selected section.
- **Only update.** Kaspersky MLAD will update information about previously created and/or imported assets and tags within the selected section.

Any new assets or tags defined in the configuration file will not be incorporated into the hierarchy.

- **Overwrite.** Kaspersky MLAD will delete previously created and/or imported assets and tags from the selected section and create new assets and tags from the configuration file.

7. If you want all assets and tags from the configuration file to be treated as new occurrences, enable the **Treat all elements as new** check box.

You can use this toggle switch to upload nested assets repeated in different sections of the asset tree. To do this, load the nested asset configuration in **Add and update** import mode multiple times, and specify different parent assets each time you upload. However, you cannot load tags with names that match the names of previously created and/or loaded tags.

8. Click the **Save** button.

Tag and asset configuration will be uploaded to Kaspersky MLAD. The assets and tags are displayed as an asset tree.

Saving tag and asset configuration to a file

System administrators can manage assets and tags.

You can save the structure of tags and assets to a file in XLSX format for subsequent use. For instance, you can edit existing asset and/or tag settings, or introduce new ones to the hierarchy. Then, upload the revised configuration file to the application again.

To save tag and asset configuration to a XLSX file:

1. In the lower-left corner of the window, click .

You will be taken to the [administrator menu](#).

2. Select the **Assets** section.

3. Click the **Export** button.

The asset and tag configuration will be saved to a file named mlad_structure.xlsx (see the example in the [Appendix](#)).

Working with the main menu

This section contains a description of user tasks performed in the [main menu](#) of the application.

Access to application functions in the main menu depends on the role [assigned to the user account](#). Users with the system administrator role have access to all functions of the application.

Scenario: working with Kaspersky MLAD

This section describes the actions that can be taken by a user when working in the main menu of Kaspersky MLAD.

The scenario for working with the application consists of the following steps:

1 Creating presets to monitor the section of the protected facility

For quick access to data, [upload a preset configuration](#) to Kaspersky MLAD. A preset configuration is created by a Kaspersky employee or certified integrator. A preset configuration is described in a JSON file. For an example of a [preset configuration description](#), see the Appendix.

You can [create presets](#) that include tags corresponding to industrial units, in the application web interface. If necessary, you can [modify](#) existing presets.

2 Preparing an ML model

To analyze the telemetry on the monitoring object and detect anomalies, [prepare ML models](#). Add ML models and markups to Kaspersky MLAD. Train the ML model elements and [check the training results](#). Should adjustments be required, modify the training parameters and retrain the relevant elements. [Start ML model inference](#) to register incidents. If required, [deploy the ML model](#) to register incidents.

3 Viewing historical data

Go to the [History](#) section. [Choose the appropriate preset](#) and [define the date and time range](#) to view historical data on process parameters and the results of their processing by ML models: generated [artifacts](#) and/or registered incidents. You can use [navigation](#) when viewing the historical data.

4 Monitoring in online mode

To view the received values of process parameters and the results of their processing by ML models, go to [Monitoring](#). [Select the relevant preset](#) and [time interval](#) to display the incoming data.

5 Viewing data in the Time slice section

To view the values of the process parameters received from the monitored asset's sensors at a certain point in time, go to the [Time slice](#) section. [Select the relevant preset](#) and [specify the date and time interval](#) for viewing the data. You can use [navigation](#) when viewing the data.

6 Working with incidents

Go to the [Incidents](#) section and [view information about the registered incidents](#). [Analyze the incidents](#) and [add expert opinions or comments](#) where you can indicate if the registered incidents are anomalies.

If you are subscribed to incident notifications, you will receive an email message when an abnormal situation arises. The message will indicate the date and time when the incident began and will provide a link you can use to go to the [History](#) section.


7 Working with events and patterns

To work with events and patterns, [configure attention settings](#) and [display of event parameters](#). Navigate to [Event Processor](#) and [create monitors](#) to track specific events, patterns, or event parameters. [View the events](#) and [patterns](#) detected by the Event Processor.

Viewing summary data in the Dashboard section

The **Dashboard** section provides summary information on the number of events and observations for tags received by Kaspersky MLAD, registered incidents, and the status of services.

The information on the page is divided into the following blocks:

- **Incoming data** is a graph that displays the number of events and observations for tags received by Kaspersky MLAD. You can enable or disable the display of incoming events and observations for tags on the graph by clicking the corresponding data signature legend under the graph. The left scale of the graph displays the range for the number of observations for tags per second. The right scale of the graph displays the range for the number of incoming events per second.
- **Latest incidents** is a table that contains [information about the latest registered incidents](#) .

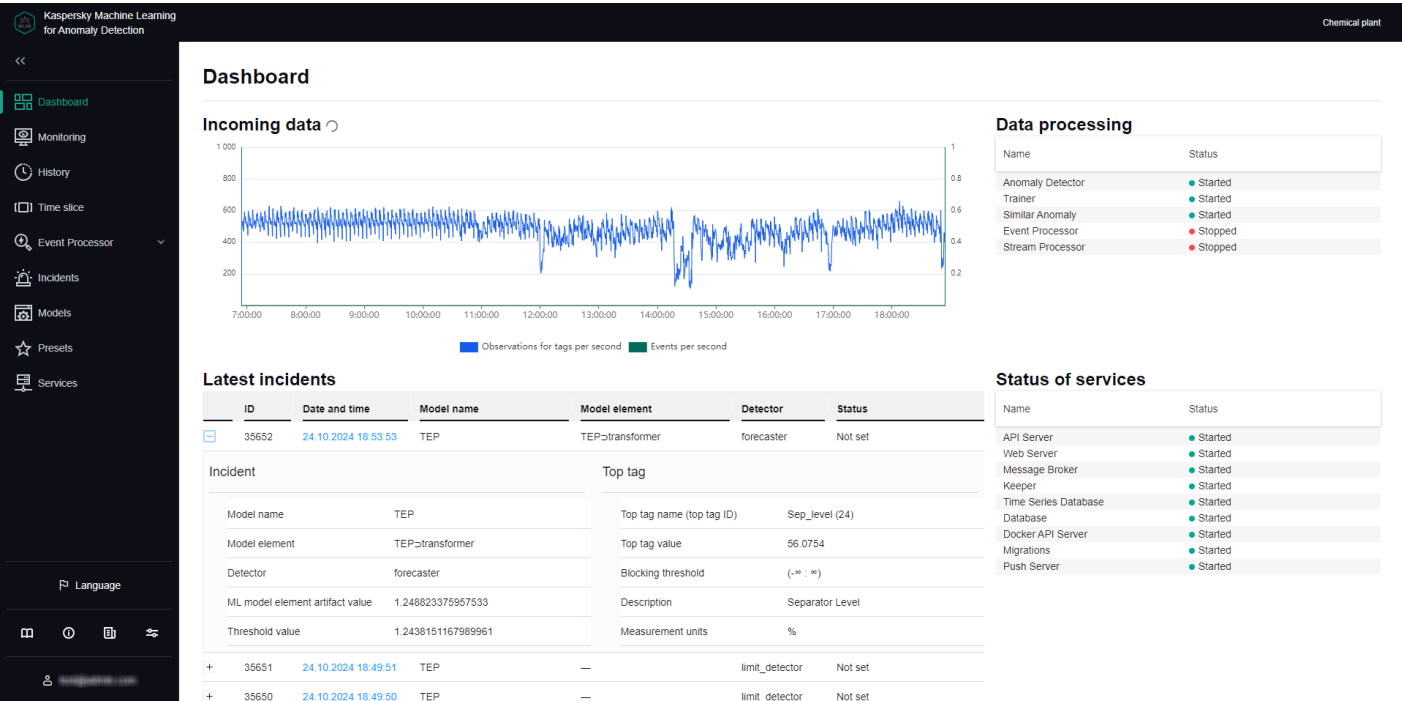
- **ID** refers to the ID of the registered incident.
- **Date and time** refers to the date and time when the incident occurred.
- **Model name** refers to the name of the ML model whose element registered the incident.
- **Model element** refers to the name of the ML model element that registered the incident.
- **Detector** refers to the type of incident registered.
- **Status** refers to the status of the registered incident [as entered by an expert](#) following an incident analysis or assigned automatically according to the incident status defined for the ML model element that registered the incident

Clicking the + button near an incident in the incidents table opens a window containing the technical attributes of the selected incident and tag:

- **Incident** is a section containing information about the incident:
 - **Model name** refers to the name of the ML model whose element registered the incident.
 - **Model element** refers to the name of the ML model element that registered the incident.
 - **Detector** refers to the type of incident registered.
 - **ML model element artifact value** refers to deviation of the monitored asset's behavior from normal at the time of incident registration. This is absent if the incident is registered by Limit Detector or Stream Processor.
 - **Threshold value** refers to the threshold value for registering an incident by an ML model element. For any incident detected by Limit Detector, the specific threshold (upper or lower) reached by the tag is recorded.
- **Top tag** is a section that contains information about the tag that had the greatest impact on incident registration:
 - **Top tag name (top tag ID)** is the name and ID of the tag that had the greatest impact on incident registration.
 - **Top tag value** is the value of the top tag registered when the incident occurred.
 - **Blocking threshold** refers to maximum permissible top tag values. Limit Detector requires these settings to function correctly. Whenever the tag value reaches its upper or lower blocking threshold, the Limit Detector registers an incident.
 - **Description** refers to a description of the top tag.
 - **Measurement units** refer to the units for measuring the top tag values.

- **Data processing** refers to a table that displays the statuses of services used for processing events and observations according to tags received by Kaspersky MLAD, training and inferencing ML models, and registering incidents.
- **Status of services** is a table that displays the status of each service.

You can proceed to the [History](#) section from the **Dashboard** section by clicking the date and time of an incident in the **Latest incidents** table. The **History** section displays detailed information about the incidents registered by Kaspersky MLAD.



Dashboard section

Viewing incoming data in the Monitoring section

In the **Monitoring** section, you can view the real-time values of the tags included in the preset and their predicted values.

The central part of the **Monitoring** section consists of a set of horizontal segments designed to display graphs. Each such segment is called a *graphic area*. The graphic areas for the [selected preset](#) are displayed first. A single graphic area of a preset can display a graph of one tag or graphs of multiple tags superimposed over each other. The composition of tags whose data is shown in the graphic area can be determined when you [create a preset](#). The graphs display the values of preset tags received by Kaspersky MLAD from the monitored object. You can [choose ML model elements](#) and [customize graph display](#) for the graphs for individual graphic areas to show [artifacts](#) linked to the tags associated with those areas and generated by the ML model elements that use these tags.

Graphic areas for each selected ML model element are displayed after the preset graphic areas. These graphical areas display graphs of [ML model element artifacts](#). The value of an ML model element artifact depends on the analytical algorithms used by the element. It is displayed as a colored line. The color of the line corresponds to the color selected for the **Color of incident dot indicators** setting when the corresponding element was created. Graphs also display an orange line that represents the threshold. When a value exceeds this threshold, the ML model element registers an incident.

At the bottom of the section, there is a graphic area that displays a graph of the ML model element artifact selected in the [ML model element artifact graph display settings](#) panel. The red line on the graph corresponds to the value of the ML model element artifact, while the orange line represents a threshold. When the value crosses this threshold, Kaspersky MLAD registers an incident. The area on the graph where the value of the ML model element artifact exceeds the specified threshold is colored red. Below the graph, color-coded dots that represent recorded incidents are displayed.

Depending on the [selected time scale](#) and the density of incidents, one dot indicator may correspond to one or multiple closely-spaced incidents that were registered by one or multiple ML model elements. The color of the indicator points relating to incidents recorded by a single ML model element is assigned when that element is created. Purple is reserved for indicator points that correspond to a group of incidents recorded by different elements. Red is reserved for indicator points that correspond to incidents recorded by Limit Detector.



Monitoring section

Viewing data for a specific preset in the Monitoring section

Kaspersky MLAD allows you to select presets for which real-time data is displayed.

To view incoming data for a specific preset in real time:

1. In the [main menu](#), select the **Monitoring** section.
2. On the opened page, select the relevant preset from the **Preset** drop-down list.

The page will display graphs for the tags included in the selected preset, according to the graphic area settings specified when [that preset was created](#).

You can [change the time interval](#) for data display, [customize graph display](#), or [select a specific ML model element](#) to view their output. You can also change which tags are displayed by [editing the preset](#).

Selecting elements of the ML models in the Monitoring section

Under **Monitoring**, you can view real-time values of tags included in the preset, artifacts generated by selected ML model elements, and the number of registered incidents.

When multiple ML models are applied to processing data for a monitored object, Kaspersky MLAD gives you the option to select several components of these models to visualize their inference results: An ML model element is not created for the Limit Detector. The dot indicators of incidents registered using this detector are displayed if [use of the Limit Detector is enabled](#) and the [display of indicators for all incidents is enabled](#).

The functionality is available after [a license key is added](#).

To view the inference results of an ML model element:

1. In the [main menu](#), select the **Monitoring** section.
2. On the opened page, select one or several elements of the ML model from the **Model element** drop-down list.

Element names are displayed as < ML model name > _ < element name >.

Graphic areas for the selected preset will display the values of tags received by Kaspersky MLAD within the selected time interval. When you [customize graph display](#), graphs for individual graphic areas will show artifacts linked to the tags associated with those areas and generated by the ML model elements that use these tags.

The central part of the section will display graphs for artifacts from the selected ML model elements. The values shown on the graphs depend on the analytical algorithms used by the elements to identify anomalies.

To hide the artifacts for a selected ML model element, click  next to the element.

3. To display a graph of a specific ML model element's artifact at the bottom of the section, do the following:

- a. Click the  button below the tag graphs on the left side of the page.

The **ML model element artifact graph display settings** pane appears on the right.

- b. From the **Model element** drop-down list, select the ML model element. You can select only one ML model element from the list.

- c. Click the **Close** button.

The graph will show the value of the ML model element's artifact as a red line. The graph area above the orange threshold line is highlighted in red to indicate above-threshold artifact values.

The lower part of the graph displays the dot indicators of incidents that were registered by the selected ML model elements. If the [display of indicators for all incidents is enabled](#), dot indicators for incidents that were registered by all ML models and [Limit Detector](#) will be displayed.

Selecting a time interval in the Monitoring section

Kaspersky MLAD lets you select the time interval (scale) for displaying incoming data.

To select a time interval:

1. In the [main menu](#), select the **Monitoring** section.
2. On the opened page, select the necessary time interval from the drop-down list. The following values are available by default:
 - 1, 5, 10, 15, and 30 minutes
 - 1, 3, 6, and 12 hours

- 1, 2, 15, and 30 days
- 3 and 6 months
- 1, 2, and 3 years


If necessary, the system administrator can [create, edit, or delete time intervals](#).

The graphs for the [selected preset](#) will display the tag values and inference results for the [selected ML model elements](#), for the chosen time interval.

Configuring how graphs are displayed in the Monitoring section

Kaspersky MLAD lets you configure how the graphic areas of presets are displayed in the **Monitoring** section.

To customize the appearance of preset graphic areas:

1. In the [main menu](#), select the **Monitoring** section.
2. On the opened page, click the  button in the upper part of the screen.
The **Graph display settings** pane appears on the right.
3. In the **Graph height** drop-down list, select one of the following values: 55 px, 110 px, 145 px, 190 px.
By default, the **Graph height** parameter is set to 55 px.
4. In the **To go to the History section, use** drop-down list, select the preset whose graphs should be displayed by default when you navigate to the **History** section.
5. Turn on the **Show observation graphs in selected color** toggle switch, and select a color in the **Color of observation graphs** field as needed.
6. Turn on the **Show prediction graphs in selected color** toggle switch, and select a color in the **Prediction graph color** field as needed.
7. Use the **Tag name and description** toggle switch to enable or disable display of the tags descriptions and names on the left of the graphs.
8. Use the **Predicted tag value** toggle switch to enable or disable display of the predicted tags values on graphs.
9. Use the **Individual tag error** toggle switch to turn on or off the display of individual tag value prediction errors on graphs.
10. Use the **Display indicators for all incidents** toggle switch to enable or disable display of the dot indicators for incidents registered by all ML models or [Limit Detector](#).
If this switch is disabled, only the dot indicators for incidents that were registered by the [selected ML model elements](#) will be shown.
11. If you need the graphs to display the defined technical limits for tags:
 - a. Turn on the **Blocking threshold** toggle switch.
 - b. If you need to always display the defined technical limits, turn on the **Always display blocking threshold** toggle switch.

If this switch is disabled, the technical limits will be displayed only if a tag value is approaching the corresponding limit in the graph area displayed on the screen.

12. Use the **Additional threshold lines** toggle switch to enable or disable the display of [additional threshold lines](#) on the graph.
13. Click the **Close** button to return to viewing graphs in the **Monitoring** section.

The defined settings for displaying graphic areas of presets in the **Monitoring** section will be applied.

Viewing data in the History section

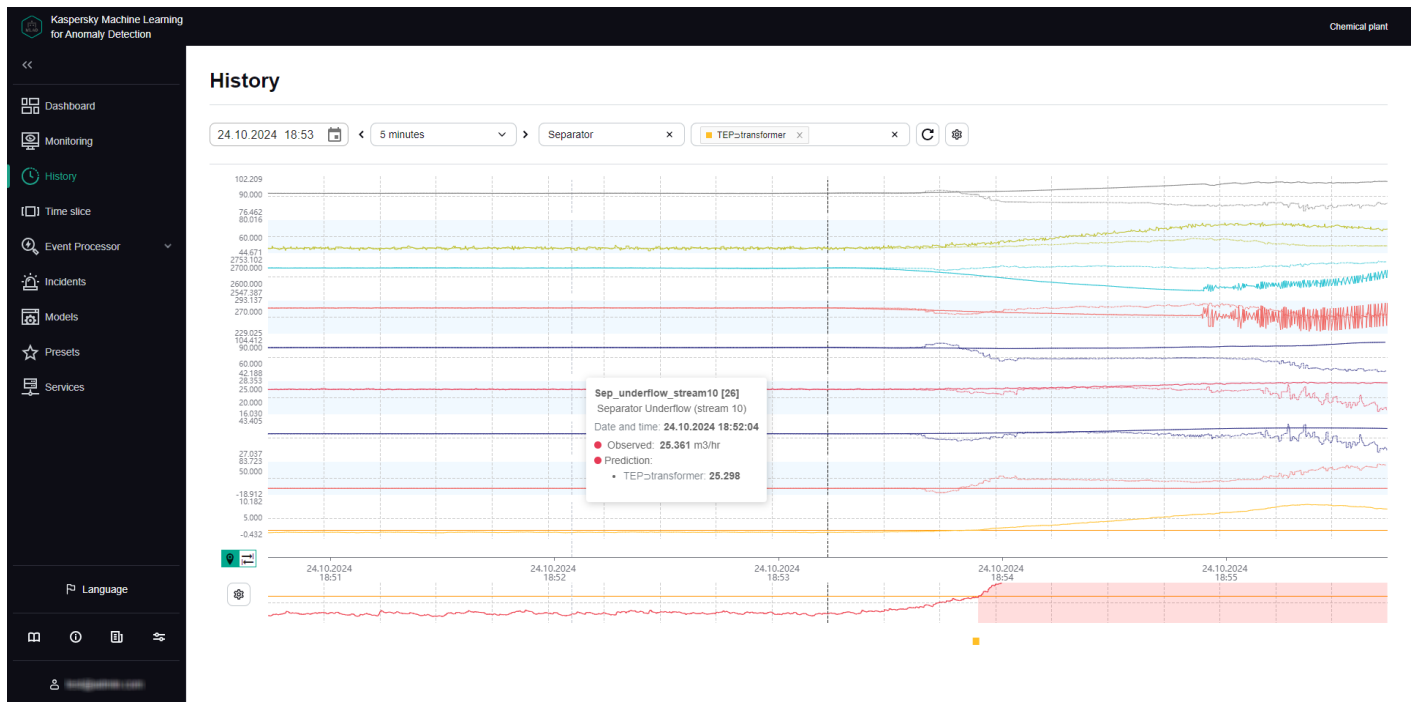
The **History** section provides access to the history of incoming data, the results of data processing by Kaspersky MLAD, generated ML model artifacts, and registered incidents information.

The central part of the **History** section consists of a set of horizontal segments designed to display graphs. Each such segment is called a *graphic area*. The graphic areas for the [selected preset](#) are displayed first. A single graphic area of a preset can display a graph of one tag or graphs of multiple tags superimposed over each other. The composition of tags whose data is shown in the graphic area can be determined when you [create a preset](#). The graphs display the values of preset tags received by Kaspersky MLAD from the monitoring object during the [selected time interval](#). You can [choose ML model elements](#) and [customize graph display](#) for the graphs for individual graphic areas to show [artifacts](#) linked to the tags associated with those areas and generated by the ML model elements that use these tags.

Graphic areas for each selected ML model element are displayed after the preset graphic areas. These graphic areas display graphs for [ML model element artifacts](#). The value of an ML model element artifact depends on the analytical algorithms used by the element. It is displayed as a colored line. The color of the line corresponds to the color selected for the **Color of incident dot indicators** setting when the corresponding element was created. Graphs also display an orange line that represents the threshold. When a value exceeds this threshold, the ML model element registers an incident.

At the bottom of the section, there is a graphic area that displays a graph of the ML model element artifact selected in the [ML model element artifact graph display settings](#) panel. The red line on the graph corresponds to the value of the ML model element artifact, while the orange line represents a threshold. When the value crosses this threshold, Kaspersky MLAD registers an incident. The area on the graph where the value of the ML model element artifact exceeds the specified threshold is colored red. Below the graph, color-coded dots that represent recorded incidents are displayed.

Depending on the selected time scale and the density of incidents, one dot indicator may correspond to one or multiple closely-spaced incidents that were registered by one or multiple ML model elements. The color of the indicator points relating to incidents recorded by a single ML model element is assigned when that element is created. Purple is reserved for indicator points that correspond to a group of incidents recorded by different elements. Red is reserved for indicator points that correspond to incidents recorded by Limit Detector.



History section

Viewing historical data for a specific preset

Kaspersky MLAD allows you to select custom presets for which historical data is displayed. If you want to view historical data for tags in the **Tags for incident #<incident ID>** dynamic preset, [click the incident registration date](#) under **Incidents**. The **Tags for incident #<incident ID>** dynamic preset contains tags that had the greatest influence on the generation of a registered incident.

To view historical data for a specific preset:

1. In the [main menu](#), select the **History** section.
2. On the opened page, select the relevant preset from the **Preset** drop-down list.

The page will display graphs for the tags included in the selected preset, according to the graphic areas settings specified when [that preset was created](#).

You can use the [time navigation](#) function to view the entire history of data. You can [edit the date and time interval](#) or [select ML model elements](#) to view their output, if needed. You can also change which tags are displayed by [editing the preset](#).

Selecting elements of the ML model in the History section

History provides the history of incoming data, the results of its processing by Kaspersky MLAD, artifacts generated by selected ML model elements, and registered incidents.

When multiple ML models are applied to processing data for a monitored object, Kaspersky MLAD gives you the option to select several components of these models to visualize their inference results: An ML model element is not created for the Limit Detector. The dot indicators of incidents registered using this detector are displayed if [use of the Limit Detector is enabled](#) and the [display of indicators for all incidents is enabled](#).

The functionality is available after [a license key is added](#).

To view the inference results of an ML model element:

1. In the [main menu](#), select the **History** section.
2. On the opened page, select one or several elements of the ML model from the **Model element** drop-down list.

Element names are displayed as < ML model name > > < element name >.

Graphic areas for the selected preset will display the values of tags received by Kaspersky MLAD for the selected time interval. When you [customize graph display](#), graphs for individual graphic areas will show artifacts linked to the tags associated with those areas and generated by the ML model elements that use these tags.

The central part of the section will display graphs for artifacts from the selected ML model elements. The values shown on the graphs depend on the analytical algorithms used by the elements to identify anomalies.

To hide the artifacts for a selected ML model element, click  next to the element.

3. To display a graph of a specific ML model element's artifact at the bottom of the section, do the following:

- a. Click the  button below the tag graphs on the left side of the page.

The **ML model element artifact graph display settings** pane appears on the right.

- b. From the **Model element** drop-down list, select the ML model element. You can select only one ML model element from the list.

- c. Click the **Close** button.


The graph will show the value of the selected ML model element's artifact as a red line. The graph area above the orange threshold line is highlighted in red to indicate above-threshold artifact values.

The lower part of the graph displays the dot indicators of incidents that were registered by the selected ML model elements. If the [display of indicators for all incidents is enabled](#), dot indicators for incidents that were registered by all ML models and [Limit Detector](#) will be displayed.

Selecting a date and time interval in the History section


Kaspersky MLAD lets you choose the date and a fixed time interval (scale) for displaying historical data or a user-defined time interval (for example, when an incident was detected).

To select the date for displaying historical data:

1. In the [main menu](#), select the **History** section.
2. Click the  button. In the opened window, select the date and time for which you need to display historical data on graphs.

3. Click the **Apply** button.

The vertical blue line on graphs will indicate the selected date and time (in the center of the graph).



4. If you need to select other date and time (point) on the graph, click the  button on the left of the time axis and select the relevant point.

The selected point will become the new center of the graph. The vertical blue dashed line will indicate the selected date and time.

To select a time interval for displaying historical data:

1. In the [main menu](#), select the **History** section.
2. On the opened page, do one of the following:
 - If you need to display data for a fixed time interval, select the relevant time interval from the drop-down list. The following time intervals are available by default:
 - 1, 5, 10, 15, and 30 minutes
 - 1, 3, 6, and 12 hours
 - 1, 2, 15, and 30 days
 - 3 and 6 months
 - 1, 2, and 3 years

If necessary, the system administrator can [create, edit, or delete time intervals](#).



- To display data for a custom time interval, click  on the left of the time axis, select an interval on the time axis, and click . If you need to change the scale again, repeat this step.

The graphs for the [selected preset](#) will display the tag values and inference results for the [selected ML model elements](#), for the chosen time interval.

Navigating through time in the History section

Kaspersky MLAD provides the capability to navigate through time for convenient viewing of historical data.

To use time navigation when viewing data:

1. In the [main menu](#), select the **History** section.
2. On the opened page, [select the time interval](#) for the data that you want to view.
3. Use the  and  buttons in the upper part of the page to move along the time axis to the right or left.

The time axis for viewing historical data on the graph will shift to the selected time interval.




Navigating through time

On graphs, a vertical blue dashed line indicates the midpoint of the selected time interval and matches the [selected date and time](#). If an interval of **1 day** is selected, the graph displays historical data for the 12-hour periods before and after the selected date and time relative to the dashed line. If necessary, you can [change the time interval](#).

Configuring how graphs are displayed in the History section

Kaspersky MLAD lets you configure the settings for displaying graphic areas of presets in the **History** section.

To customize the appearance of graphic areas:

1. In the [main menu](#), select the **History** section.
2. On the opened page, click the  button in the upper part of the screen.
The **Graph display settings** pane appears on the right.
3. In the **Graph height** drop-down list, select one of the following values: 55 px, 110 px, 145 px, 190 px.
By default, the **Graph height** parameter is set to 55 px.
4. Turn on the **Show observation graphs in selected color** toggle switch, and select a color in the **Color of observation graphs** field as needed.
5. Turn on the **Show prediction graphs in selected color** toggle switch, and select a color in the **Prediction graph color** field as needed.
6. Use the **Tag name and description** toggle switch to enable or disable display of the tags descriptions and names on the left of the graphs.
7. Use the **Predicted tag value** toggle switch to enable or disable display of the predicted tags values on graphs.
8. Use the **Individual tag error** toggle switch to turn on or off the display of individual tag value prediction errors on graphs.
9. Use the **Display indicators for all incidents** toggle switch to enable or disable display of the dot indicators for incidents registered by all ML models or [Limit Detector](#).
If this switch is disabled, only the dot indicators for incidents that were registered by the [selected ML model elements](#) will be shown.
10. If you need the graphs to display the defined technical limits for tags:
 - a. Turn on the **Blocking threshold** toggle switch.
 - b. If you need to always display the defined technical limits, turn on the **Always display blocking threshold** toggle switch.
If this switch is disabled, the technical limits will be displayed only if a tag value is approaching the corresponding limit in the graph area displayed on the screen.
11. Use the **Additional threshold lines** toggle switch to enable or disable the display of [additional threshold lines](#) on the graph.
12. Click the **Close** button to return to viewing graphs in the **History** section.

The defined settings for displaying graphic areas of presets in the **History** section will be applied.

Viewing data in the Time slice section

In the **Time slice** section, you can view the values of process parameters received from sensors of the monitored asset at the same point in time. The sensors must be of the same type (have the same dimension) and must be positioned linearly, like pressure sensors in an oil pipeline, for example.

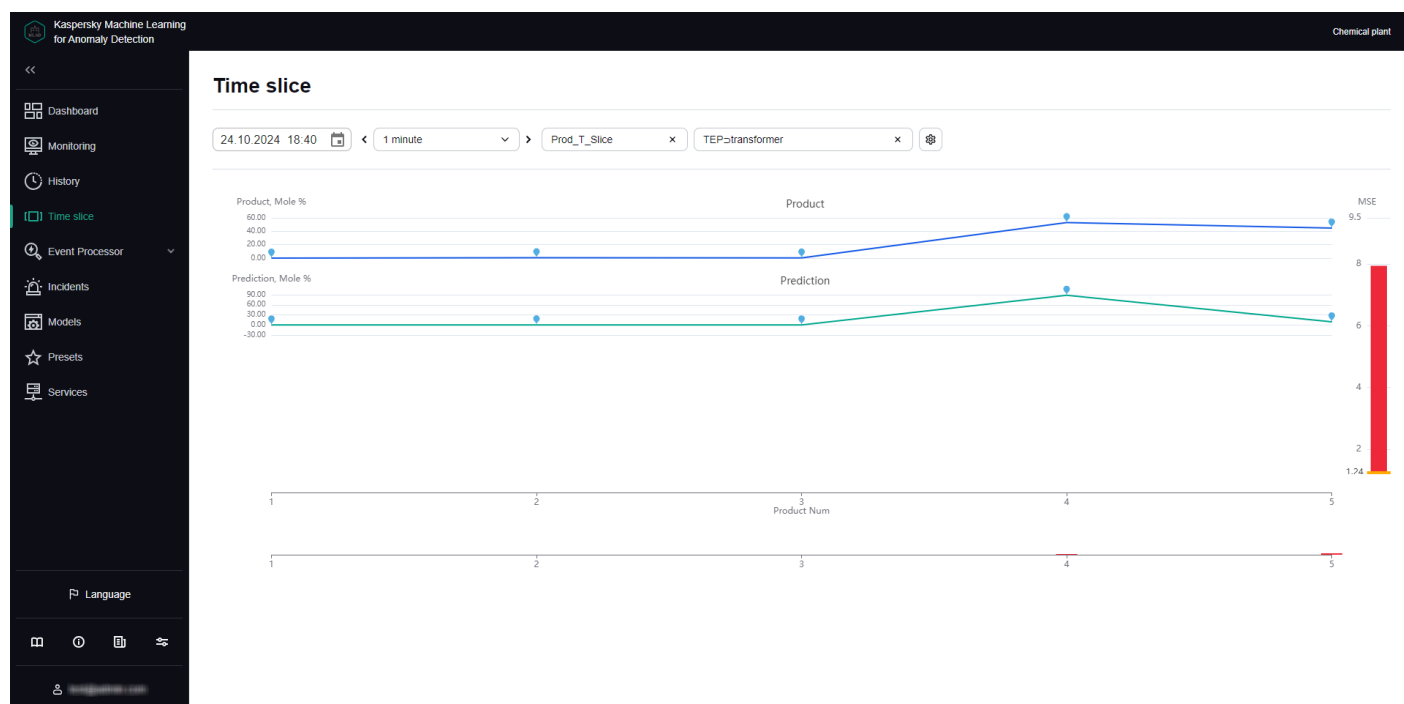
Data is presented in the form of graphs that display whether an incident was detected at the selected time and where the likely source of the incident is located.

The lower part of the page contains a section displaying the individual errors of tags. The data is presented as a bar graph. The error value for each tag is displayed when the mouse cursor hovers over the relevant column. The prediction error graph is located on the right of the preset tag graphs.

In the **Time slice** section, you can use the drop-down list to [select a preset](#) and the [date and time](#) when data was received. This list includes special presets that can be [created](#) in the **Presets** section. A special preset should contain only tags of the same type that have defined x-axis coordinates. You can additionally specify expressions dynamically calculated for each tag based on actual and predicted tag values, individual prediction errors, and tag coordinate values and constants defined in expressions.

You can also [customize the display of graphs](#), [select a time interval](#) for viewing data, and [select a specific element of the ML model](#) to view the individual errors of preset tags obtained as a result of data processing by the selected element of the ML model.

Data processing results can be displayed only for predictive ML model elements.



Time slice section

Viewing data for a specific preset in the Time slice section

To view data for a specific preset:

1. In the [main menu](#), select the **Time slice** section.
2. On the opened page, select the relevant preset from the **Preset** drop-down list.

The page displays graphs for tags that are included in the selected preset.

If necessary, you can [change the time interval](#) for displaying data, [customize the display of a graph](#), or [select a specific element of the ML model](#). You can also change which tags are displayed by [editing the preset](#).

Selecting a specific element of the ML model in the Time slice section

If the ML model used for a monitored asset has several elements for processing and predicting data, Kaspersky MLAD lets you select a specific element of the ML model to display the individual tag errors obtained as a result of this element in the **Time slice** section.

The functionality is available after [a license key is added](#).

Data processing results can be displayed only for predictive ML model elements.

To view the individual tag errors resulting from data processing by a specific ML model element:


1. In the [main menu](#), select the **Time slice** section.
2. On the opened page, select the relevant element of the ML model from the **Model element** drop-down list.
Element names are displayed as < ML model name > _ < element name >.

The bottom of the section displays the individual tag errors resulting from data processing by the selected element of the ML model.

Selecting a date and time interval in the Time slice section

Kaspersky MLAD lets you select a date and time interval (scale) for displaying incoming data.

To select the date for displaying incoming data:

1. In the [main menu](#), select the **Time slice** section.
2. Click the  button. In the opened window, select the date and time for which you need to display data.
3. Click the **Apply** button.

The graphs will display the tag values for the selected date and time.

To select a time interval for displaying incoming data:

1. In the [main menu](#), select the **Time slice** section.
2. Select the required time interval from the drop-down list in the upper part of the opened page. The following time intervals are available by default:
 - 1, 5, 10, 15, and 30 minutes
 - 1, 3, 6, and 12 hours

- 1, 2, 15, and 30 days
- 3 and 6 months
- 1, 2, and 3 years

If necessary, the system administrator can [create, edit, or delete time intervals](#).

The page will display graphs of the [defined preset](#) for the selected time interval.

Navigating through time in the Time slice section

Kaspersky MLAD provides the capability to navigate through time for convenient viewing of data.

To use time navigation when viewing data:

1. In the [main menu](#), select the **Time slice** section.
2. On the opened page, [select the time interval](#) for the data that you want to view.
3. Use the < and > buttons in the upper part of the page to move along the time axis to the right or left.

The time axis for viewing data on the graph will shift to the selected time interval.




Navigating through time

Configuring how graphs are displayed in the Time slice section

Kaspersky MLAD lets you configure the settings for displaying preset graphs in the **Time slice** section.

To configure the display settings for preset graphs:

1. In the [main menu](#), select the **Time slice** section.
2. On the opened page, click the  button located in the upper part of the screen.
The **Graph display settings** pane appears on the right.
3. In the **Graph height** drop-down list, select one of the following values: 55 px, 110 px, 145 px, 190 px.
By default, the **Graph height** parameter is set to 55 px.
4. Click the **Close** button to return to viewing the graphs.

The graph display settings will be applied.

Working with events and patterns

The **Event Processor** section provides data on [events](#) and the structure of [patterns](#) detected by the Event Processor service in the event stream received from external sources or from the Anomaly Detector service.

In the **Event Processor** section, you can [view the history of received events](#) and the [registration history of new and/or persistently recurring patterns](#). You can also configure the display of event parameters and can configure pattern registration settings. On the **Monitoring** tab, you can [monitor](#) specific events, patterns, or values of event parameters, and generalized events and patterns received by the Event Processor within the data stream from monitored assets.

The functionality is available after [a license key is added](#).

If restarted, Kaspersky MLAD restores the state of the Event Processor service and pauses the processing of data received from the CEF Connector. This data is temporarily stored in the internal queue of the application message broker. Until the Event Processor service is restored, the **Event Processor** section tabs will display a notification informing you that the Event Processor service has stopped. This service restoration process may take several minutes if there is a significantly large number of processed events or registered patterns.

Kaspersky Machine Learning
for Anomaly Detection

Chemical plant

Dashboard

Monitoring

History

Time slice

Event Processor

Monitoring

Event history

Patterns history

Incidents

Models

Presets

Services

Language

100%

100%

100%

100%

100%

100%

Event Processor

Configure filter display

Configure attention

Monitoring

Event history

Patterns history

Filters

History interval
01.10.2020 19:02 -- 24.10.2024 19:02

Process request

User_Host
Search by value

User_Name
Ivanov

Destination_Host
Search by value

Access_Result
Search by value

Incident_id
Search by value

Incident_group_name
Search by value

Incident_tag_id
Search by value

Incident_tag_name
Search by value

Incident_model_name
Search by value

Incident_detection_system
Search by value

Incident_triggered_tag_value
Search by value

Graph

Table

Event ID	System parameters	Event parameters
03ea4a40-b56e-4150-87f1-d4a2276c7658	Last detection in interval 25.10.2021 17:00:00.000 Activations count in interval 1 Parameter count 1 Last activation 25.10.2021 17:00:00.000	User_Host 192.168.2.0 User_Name Ivanov Access_Result Deny Destination_Host Teams
b9d43e53-1fb8-4262-b0c9-beb41b1b078c	Last detection in interval 25.10.2021 21:00:00.000 Activations count in interval 1	User_Host 192.168.2.0 User_Name Ivanov

Event Processor section

About Event Processor

The Kaspersky MLAD Event Processor is designed to detect regularities in the form of recurring [events](#) and [patterns](#) in the stream of events received from monitored assets and to detect new events and patterns. New events and patterns may indicate an [anomaly](#) in the monitored asset operation.

You can also focus the event processor attention on the overall behavior of the monitored asset. The event processor will register generalized events and patterns that lack generalized event parameters.

About events

Data received from monitored assets and from the Anomaly Detector service are processed as events by the Event Processor service. *Event* is a set of values taken from a predetermined list of parameters and indicating what happened on a monitored asset at a given moment. The set of event parameters depends on the monitored asset and is defined [in the configuration file for the Event Processor service](#).

The Event Processor is designed to work only with categorical values of the event parameters. Event parameter values are converted to string type. Kaspersky MLAD uses the Anomaly Detector service to work with numeric values of telemetry data when processing the event stream. The system administrator can enable the processing of incidents received from the Anomaly Detector service when [configuring the Event Processor service settings](#).

An event is a phenomenon distinct from other events. There may also be intervals of time during which no events have occurred. Event registration may be affected by such factors as the actions of personnel, changes in the asset operating mode at the facility, or the execution of ICS commands by a specialist.

[Examples of situations that may lead to event registration in Kaspersky MLAD](#) 

Event examples are provided for various monitored assets.

- *Employee login.*
 - Event time: 11/10/21 09:03
 - Event parameters:
 - Source: ACS
 - Employee: Smith
 - Station: engine room door, exterior side
 - Result: Passage.
- *Unit startup.*
 - Event time: 11/10/21 09:09
 - Event parameters:
 - Source: Operator workstation
 - User: Smith
 - Equipment: Unit 1
 - Command: Ignition switched on
 - Current: 44 A
 - Duration: 10 seconds.
- *Mode activation.*
 - Event time: 11/10/21 09:24
 - Event parameters:
 - Source: ICS
 - Equipment: Unit 1
 - Nominal mode: True.

An event is registered once by the Event Processor service. When an event stream is received, the Event Processor recognizes previously detected events. If events are found that do not match those previously detected, the Event Processor registers new events.

You can [view the received events](#) as a graph or a table. To view events, you need to upload them to **Event Processor** → **Event history**. Event parameters specified in the configuration file for the Event Processor service may not appear in all events received from the monitored asset. Thus, some parameters may be missing when you view the received events.

About patterns

The Event Processor detects regularities in the stream of events arriving from the monitored asset. These regularities are detected as a hierarchy of stable (persistently recurring) patterns, which can be either *simple patterns* (sequences of events) or *composite patterns* (sequences of patterns). The patterns that form a composite pattern are called subpatterns.

A sequence of events or patterns is considered recurrent if its constituent elements follow the same order, and the time intervals between similar elements in different sequences differ from each other by no more than a specific maximum range. The allowable range of intervals between the pattern elements is calculated considering the value of the [Coefficient defining the permitted dispersion of the pattern duration](#) parameter. Patterns are the result of the specific facility's adopted practices, prescribed procedures, or technical specifics of the industrial process.

The Event Processor presents the detected regularities as a layered hierarchy of nested elements (pattern structure) down to the event level. Events are the first layer elements, simple patterns are the second layer elements, and composite patterns are the third and higher layer elements. Event parameter values are elements of the null layer.

A pattern is registered once by the Event Processor service. When an event stream is received, the Event Processor recognizes previously detected patterns. If patterns are found that do not match previously detected regularities, the Event Processor registers new patterns.

New patterns also include the sequences of events or patterns with a deviation in the order or composition of subpatterns (for example, turning on an industrial unit before the operator has arrived at the workstation) or with significant changes in the intervals between events or subpatterns even though their sequence is preserved (for example, turning on an industrial unit immediately after or a lot later than the operator arrived at the workstation). Thus, the Event Processor registers patterns with a new structure.

New patterns may indicate an [anomaly](#) in the monitored asset operation. You can [view the structure of the new pattern](#) and examine its deviations from the structure of previously detected patterns.

If a newly identified sequence of events or patterns begins to repeat in a persistent manner, this sequence is converted to a stable pattern.

Event Processor can register patterns where the values of one or more event parameters, such as the name of the employee who turned on the machine, are irrelevant. These patterns are referred to as *generalized*. To register generalized patterns, set **Generalized attention** as the attention type when [configuring attention](#). You can also specify **Generalized parameter** as the condition type when configuring attention subject conditions. Generalized attention subject and condition parameters will not be displayed when [viewing the structure of generalized patterns](#) on the **Patterns history** tab.

About attention

The event stream from the monitored asset usually contains many unrelated events. The Event Processor service supports an attention mechanism to detect patterns based on a specific subset of events from the entire stream.

Attention is a special event processor configuration intended to track events and patterns for specific subsets of event history, and to detect commonalities in the behavior of the monitored asset.

Attention heads form the foundation of attention configuration. They define the attention subject parameter and attention subject condition parameters. The attention subject corresponds to the main event parameter that the event processor will use to register events and patterns. The conditions correspond to additional criteria for registering events and patterns for other event parameters. An attention head processes only those events in the entire incoming event stream that satisfy the specified attention subject and conditions. The event processor can process event streams for multiple attention heads simultaneously.

The event processor can register generalized events and patterns to track general behavior for different attention subject values. To do this, set **Generalized attention** as the attention type when configuring the attention subject. You can also specify **Generalized parameter** as the condition type when configuring attention subject conditions. Generalized attention subject and condition parameters will not be displayed when viewing generalized events or patterns. They will, however, influence the rules for extracting these generalized events and patterns from the stream.

You can [configure attention](#) in the **Event Processor** section.

About Event Processor operating modes

Kaspersky MLAD has the following operating modes of the Event Processor service:

- **Online mode.** In the online mode, the Event Processor processes the incoming stream as episodes. An *episode* is a sequence of events from the entire stream that is limited by a specific time period and/or the number of events. An episode is formed when one of the following conditions is fulfilled:
 - The episode accumulation time reached the limit defined by the [Interval for receiving batch events \(sec.\)](#) parameter of the Event Processor service.
 - The number of accumulated events reached the limit defined by the [Batch size in online mode \(number of events\)](#) parameter of the Event Processor service.

Based on an episode received in the event stream, the Event Processor service detects new and/or repeated (stable) events and patterns for each of the defined attention heads. You can [configure attention heads](#) in the **Event Processor** section.

When an event with the timestamp belonging to a previously processed episode is received, the Event Processor service does not revise the structure of patterns detected during the processing of that episode. The Event Processor service takes into account the events received by Kaspersky MLAD with a delay when detecting patterns during the event history reprocessing in the sleep mode.

- **Sleep mode.** To improve the quality and structure of the identified patterns, the Event Processor can switch to sleep mode according to the specified schedule. Processing of the event stream in the online mode is paused, and Kaspersky MLAD accumulates incoming events in the internal limited buffer on the server for subsequent processing after the application switches from the sleep mode back to online mode.

In sleep mode, the Event Processor re-analyzes sequences of events that were previously processed in online mode. To detect more complex pattern structures in the sleep mode, the Event Processor processes sequences of events during longer time intervals than the episode accumulation time in the online mode.

In the [Event Processor service settings](#), you can configure a schedule for the sleep mode (for example, at the time when the event stream is least intense) and define a time interval for the events analyzed in the online mode to be forwarded for reprocessing in the sleep mode.

About monitors

A *monitor* is the source of notifications about patterns, events, or values of event parameters detected by the Event Processor according to the defined monitoring criteria. The monitoring criteria define the attention head, additional filters for event parameter values, a sliding time window, and the number of consecutive monitor activations within that window.

You can [create monitors](#) for alerts about the following detections in the event stream:

- **Values of event parameters.** You can create a monitor for alerts about the identification of new or previously encountered values of a specific event parameter. For example, to track new users on a monitored asset, create a monitor with the **Parameter values** subscription type and configure it to detect new values for the **User** parameter.
- **Events.** You can create a monitor for alerts about the identification of new or previously encountered events. You can also focus the attention of the Event Processor on a specific parameter of events. For example, to track new actions of a specific user at the monitored asset, you need to create a monitor with the **Events** subscription type and specify the name of the user whose actions you want to track in the **User** event parameter.
- **Patterns.** You can create a monitor for alerts about the identification of new or previously encountered patterns. For example, to track regularities in the actions of a specific user at the monitored asset, create a monitor with the **Patterns** subscription type, focus the attention of the Event Processor on the **User** parameter, and set this parameter to the name of the user whose actions you want to track.
- **Similar generalized events or patterns.** You can create a monitor to receive alerts about similar generalized events or patterns. If you want to track overall patterns in the actions of different users on a monitored asset, then when creating a monitor, you need to select the **Similar generalized** subscription type, choose the generalized attention head for **User**, and select **Subscription to patterns** for **Subscription to events or patterns**.
- **Unique generalized events or patterns.** You can create a monitor to receive alerts about unique generalized events or patterns. For example, to track new overall patterns in the actions of any user, select the **Unique generalized** subscription type when creating a monitor. For **User**, select a generalized attention head with conditions for additional parameters that match your expectations of different users' behavior. Select **Subscription to patterns** for **Subscription to events or patterns**. For **Sliding window (sec.)**, specify a time interval for the event processor to wait for a similar generalized pattern for other users. If the event processor does not detect such a pattern, the monitor will send an activation alert.

You can set fuzzy filters in the monitoring criteria. For example, you can create a monitor to track situations when a user (monitoring all values of the **User** parameter) accessed the accounting server (the value of the **Server** parameter) more than ten times (the value of the **Activation threshold** field) in the last five minutes (the value of the sliding time interval).

When events, patterns and event parameter values matching the monitoring criteria are detected in the stream of incoming data, the Event Processor activates the monitor. Kaspersky MLAD displays information about the number of monitor activations when viewing a monitor, and sends to the external system alerts about the activation of monitors when the specified threshold is reached for a sliding window using the [CEF Connector](#).

The custom monitors are displayed in the **Event Processor** section on the **Monitoring** tab.

Configure display of event parameters

Before the Event Processor service can process events, you need to configure the way event parameters are displayed.

The functionality is available after [a license key is added](#).

To configure how event settings are displayed:

1. In the main menu, select the **Event Processor** → **Monitoring** section.

2. On the page that opens, click **Configure filter display**.

The **Configure display of event parameter filters** panel will appear on the right.

3. To configure the display of filters for the event parameters, in the **Filters** section on the **Event history** and **Patterns history** tabs, select the check boxes next to the names of the desired event parameters.

By default, the pane displays the [event parameters from the Anomaly Detector service](#). To display custom event parameters, [load the Event Processor service configuration file](#). All available event parameters are selected by default.

If the [Process incidents as events](#) function is enabled, the Event Processor receives events with the following parameters:

- **incident_detection_system** refers to the type of incident registered.
- **incident_model_name** – the name of the ML model used.
- **incident_tag_name** – the name of the tag whose behavior invoked registration of the incident.
- **incident_group_name** – the name of the incident group to which the registered incident belongs.
- **incident_triggered_tag_value** – the value of the tag whose behavior invoked registration of the incident.
- **incident_id** – the ID of the registered incident.
- **incident_tag_id** – the ID of the tag whose behavior invoked registration of the incident.

If necessary, in the **Filters** section you can change the display order for the event parameters. To do so, drag the event parameter to the required place in the **Configure display of event parameter filters** panel by holding the dots (::) on the left of the event parameter name.

4. Click the **Save** button.

Configure attention settings

Before events are processed by the Event Processor service, attention settings must be configured.

Attention heads form the foundation of attention configuration. They define the attention subject parameter and attention subject condition parameters. The attention subject corresponds to the main event parameter that the event processor will use to register events and patterns. The conditions correspond to criteria for registering events and patterns for other event parameters. An attention head processes only those events in the entire incoming event stream that satisfy the specified attention subject and conditions.

The event processor can register generalized events and patterns to track general behavior for different attention subject values. To do this, set **Generalized attention** as the attention type when configuring the attention subject. You can also specify **Generalized parameter** as the condition type when configuring attention subject conditions. Generalized attention subject and condition parameters will not be displayed within registered events or patterns. They will, however, influence the rules for extracting these generalized events and patterns from the stream.

All created attention heads and [information about these](#) are displayed in the **Attention heads** panel. To view information about attention heads in the **Attention heads** panel, click **Configure attention**.

- **Name** is the name of the attention head.
- **Attention subject parameter** is the name of the event parameter selected as the attention subject.
- **Attention type** is the type of attention according to which the event processor registers events and patterns.
- **State** indicates whether this attention head is in use.
- **Actions** are the buttons for editing or deleting attention heads.

Adding an attention head

You can create multiple attention heads and use different [attention heads](#) for different monitors simultaneously.

The functionality is available after [a license key is added](#).

A large number of attention heads can lead to reduced event processor performance and slow down the core Kaspersky MLAD services, such as data reception, anomaly detection, and the web interface. To clarify the number of attention heads, it is recommended to consult with Kaspersky experts or a certified integrator.

To add an attention head:

1. In the main menu, select the **Event Processor** → **Monitoring** section.
2. On the page that opens, click **Configure attention**.
The **Attention heads** panel appears on the right.
3. To add an attention head, click **Add attention head**.
The **Add attention head** panel appears on the right.
4. In the **Name** field, specify the attention head name.
5. To use the attention head when processing an event flow, set the **State** toggle switch to **Active**.
6. Under **Attention subject**, do the following:
 - a. From the **Event parameter** drop-down list, select the primary event parameter you want to register events and patterns for.

b. In the **Attention type** drop-down list, select one of the following values:

- **Attention.** When registering events and patterns, the event processor's attention will be directed to the selected event parameter based on selected value.
- **Generalized attention.** When registering events and patterns, the event processor will aggregate the selected values by selected event parameter.

When this attention type is selected, the event processor will register generic patterns that will not display the selected event parameter with the selected value when [viewed](#). The Event Processor will track each specified event parameter value separately.

c. Perform one of the following actions:

- To include or generalize all values of an event parameter in attention, select **All values** from the **Value type** drop-down list.

Selecting **All values** causes the event processor to track events and patterns for each specific event parameter value separately. To ensure stable event processor performance, we recommend defining specific values for the event subject.

- To include or generalize specific event parameter values in attention, select **Specific values** from the **Value type** drop-down list and enter the relevant value in the **Value** field. As you start typing a value, all matching parameter values are displayed in the list.

If you selected **Generalized attention** as the attention type, select at least two values for the event parameter.

- To include or generalize event parameter values according to a template in attention, from the **Value type** drop-down list, select **Regular expression** and enter the value template using a regular expression in **Value**.

You can use [special characters of regular expressions](#) to search for events and patterns based on regular expressions.

7. If you need to generalize other event parameters, set the **Generalize condition parameters** toggle switch to **Enabled**.

If generalized attention was selected as the attention type, then, when the switch is on, the event processor will generalize the remaining event parameters across all their values. In this case, the event processor will not register any event or pattern. To enable the Event Processor to generate events or patterns, you must define at least one event parameter in the **Conditions** block without generalization based on its values.

8. To refine the criteria for registering patterns using additional event parameters, do the following under **Conditions**:

a. Click the **Add condition** button.

b. From the **Event parameter** drop-down list, select an additional event parameter to refine the data sample for events and patterns registration.

c. In the **Condition type** drop-down list, select one of the following values:

- **Parameter.** When registering events and patterns, the event processor will consider the values of the selected event parameter while taking into account the data sample obtained for the main event parameter.
- **Generalized parameter.** When registering events and patterns, the event processor will aggregate the values of the selected parameter while considering the data sample obtained for the primary event

parameter.

When this condition type is selected, the event processor will register patterns that, when viewed, will not display the selected event parameter with the selected value.

This value is available if the **Generalized attention** type is selected for the attention subject.

d. Perform one of the following actions:

- To include or generalize the new values of an event parameter in attention, select **New values** from the **Value type** drop-down list.


New values is available in the following cases:

- The condition type is set to **Parameter**.
- The attention type is set to **Attention**, the **Generalize condition parameters** toggle switch is off, and the condition type is set to **Generalized parameter**.
- To include or generalize all values of an event parameter in attention, select **All values** from the **Value type** drop-down list.

All values is available in the following cases:

- The **Generalize condition parameters** toggle switch is on, and the condition type is set to **Parameter**.
- The **Generalize condition parameters** toggle switch is off, and the condition type is set to **Generalized parameter**.
- To include or generalize specific event parameter values in attention, select **Specific values** from the **Value type** drop-down list and enter the relevant value in the **Value** field. As you start typing a value, all matching parameter values are displayed in the list.
- To include or generalize event parameter values according to a template in attention, from the **Value type** drop-down list, select **Regular expression** and enter the value template using a regular expression in **Value**.

You can use [special characters of regular expressions](#) to search for events and patterns based on regular expressions.

You can set more than one condition for additional event parameters. You can delete a previously added condition by clicking  next to the condition.

The conditions will be additionally applied to the data sample obtained for the main event parameter set under **Attention subject**. For example, if the **Generalized attention** type is selected and the **Generalize condition parameters** toggle switch is on, the Event Processor will register patterns that will display only those event parameters that were specified under **Conditions** while considering their selected values. If the toggle switch is off, the event processor will register patterns that will not display the generalized parameter specified under **Attention subject**. In this case, the values of the event parameters specified under **Conditions** will be considered.

9. Click the **Save** button.

Information about the new attention head will be displayed in the table, in the **Attention heads** panel. You can [rename](#) the attention head, and enable or disable the use of the attention head for event processing.


Editing an attention head

You can enable or disable the use of the attention head when processing the flow of events.

You cannot modify attention subject or condition parameters. You can [remove attention heads](#) or [create new ones](#) if needed.

The functionality is available after [a license key is added](#).

To edit an attention head:

1. In the main menu, select the **Event Processor** → **Monitoring** section.
2. On the page that opens, click **Configure attention**.
The **Attention heads** panel appears on the right.
3. Click  next to the attention head you want to edit.
The **Edit attention head** panel appears on the right.
4. Rename the attention head as needed.
5. Perform one of the following actions:
 - To use the attention head when processing an event flow, set **State** to **Active**.
 - To disable the use of the attention head when processing an event flow, set **State** to **Inactive**.
6. Click the **Save** button.

Removing an attention head

The functionality is available after [a license key is added](#).

To delete an attention head:


1. In the main menu, select the **Event Processor** → **Monitoring** section.
2. On the page that opens, click **Configure attention**.
The **Attention heads** panel appears on the right.
3. Click  next to the attention head you want to delete.
4. In the window that opens, confirm that you want to delete the attention head.

Information about the attention head will be deleted from the table in the **Attention heads** panel. Patterns detected according to this attention head will also be removed from Kaspersky MLAD.

Managing monitors

The functionality is available after [a license key is added](#).

Under **Event Processor** → **Monitoring**, you can manage monitors to track specific events, patterns, event parameter values, and generalized events or patterns. You can view a summary of registered activations by monitor as a histogram.

You can manage monitors on the **Monitors** tab. To navigate to the tab, click  in the upper right corner of the section.

The tab displays all monitors created in the application, with the following brief information:

- Monitor name.
- Number of monitor activations on the sliding window.
- Monitor subscription type. The following values can be displayed for each monitor:
 - **Parameter values.** The monitor tracks the occurrence of certain event parameter values.
 - **Events.** The monitor tracks the occurrence of certain events.
 - **Patterns.** The monitor tracks the occurrence of patterns in the behavior of the monitored asset.
 - **Unique generalized.** The monitor tracks the occurrence of unique generalized events or patterns.
 - **Similar generalized.** The monitor tracks the occurrence of similar generalized events or patterns.
- **Activation threshold:** the number of monitor activations on the sliding window that causes the application to send monitor activation alert to the external system when reached.
- **Period:** the sliding window during which the number of monitor activations is tracked.

You can view [detailed information about each monitor](#)  if needed. To do so, click the monitor tile.

- **Name:** name of the monitor being viewed.
- **State:** parameter that determines the monitor state.
- **Monitor ID:** unique identifier of the monitor being viewed.
- **Activations count** is number of registered monitor activations on the sliding window.
- **Date and time of last activation:** date and time when the monitor was last activated.
- **Activation stack size** determines the number of most recent monitor activations displayed in the **Activation stack** table.
- **Subscription type** indicates what is being tracked by the viewed monitor: event parameter values, events, or patterns.
- **Sliding window** indicates the time interval from the current time back to the time sequence for which the number of activations is taken into account. This window shifts synchronously with the passage of time according to the timestamps in events.
- **Activation threshold** indicates the number of activations that must be registered by the monitor on the sliding window before sending an alert about the monitor activation to the external system via the CEF Connector.
- **Attention head** indicates the specific attention head that is the current focus of the Event Processor. This parameter is displayed only when the monitor is activated by a pattern, or unique or similar generalized event or pattern.
- **Attention subject parameter** indicates the specific parameter of the attention subject that is the current focus of the Event Processor. This parameter is displayed only when the monitor is activated by a pattern, or unique or similar generalized event or pattern.
- **Subscription to events** determines whether the monitor is tracking generalized events. This parameter is displayed only when the monitor is activated by a unique or similar generalized event or pattern.
- **Subscription to patterns** determines whether the monitor is tracking generalized patterns. This parameter is displayed only when the monitor is activated by a unique or similar generalized event or pattern.
- **Activation type** determines whether the monitor is tracking new values of event parameters, events, and patterns. This parameter is displayed only when the monitor is activated by an event parameter value, event or pattern.
- **Filters** is a table containing information about filters for event parameters observed by the current monitor to track event parameter values, events, and patterns. The following data is displayed for each element:
 - **Parameter name** refers to the name of the event parameter whose values are being observed by the viewed monitor.
Each monitored asset has its own specific incoming events and event parameters. The names of event parameters are defined in the [configuration file for the Event Processor service](#). The configuration file is created and uploaded by a qualified technical specialist of the Customer, a Kaspersky employee, or a certified integrator during [configuration of the Event Processor service](#).
 - **Filter type** determines the type of filter for event parameters that are observed by the current monitor to track event parameter values, events, and patterns.

- **Value type** defines which types of values are being tracked by the viewed monitor: values based on a template, specific values, new values, or all values.

- **Values** refers to the values of the event parameter that is being observed by the viewed monitor.

This table is displayed only when the monitor is activated by an event parameter value, event, or pattern.

- **Activation stack** is a table that contains information about the latest activations of the monitor:

- **Parameter value ID** is the ID of the event parameter value whose detection caused the monitor activation. This parameter is displayed only when the monitor is activated by an event parameter value.

- **Event ID** is the ID of the event whose detection caused the monitor activation. This parameter is displayed only when the monitor is activated by an event.

- **Pattern ID** is the ID of the pattern whose detection caused the monitor activation. This parameter is displayed only when the monitor is activated by a pattern.

- **System parameters** is a group of system settings containing the following information:

- **Event date and time** is the date and time when the event is detected in the event stream.

- **Interval from previous item** is the time interval between the current and the previous event in the event stream on the sliding window. Kaspersky MLAD displays the time intervals between events upon the first detection of the pattern containing the events. When a pattern is detected again, the Event Processor takes into account the [coefficient of allowed intervals dispersion](#) specified by the administrator for these events.

- **Total activations** is the number of event occurrences in the event stream on the sliding window.

- **Parameter count** is the number of event parameters for which the values were received from the monitored asset.

- **Last activation** is the date and time when the event was last detected in the event stream on the sliding window.

This group of parameters is displayed only when the monitor is activated by an event or an event parameter value.

- **Attention subject** is the attention subject parameter and its value whose detection activated the monitor. This parameter is displayed only when the monitor is activated by a pattern.

- **Activation date and time** is the date and time when the monitor was activated. This parameter is displayed only when the monitor is activated by a pattern.

- **Event parameter** is the value of the event parameter received from the monitored asset. This parameter is displayed only when the monitor is activated by an event parameter value.

- **Event parameters** are the values of the parameters of the event received from the monitored asset. This parameter is displayed only when the monitor is activated by an event.

- **Event count** is the number of events included in the pattern that caused the monitor activation. This parameter is displayed only when the monitor is activated by a pattern.

- **Total activations**: the number of pattern occurrences in the event stream on the sliding window. This parameter is displayed only when the monitor is activated by a pattern.

- **Statistics on generalized events** is a table that contains information about generalized events:
 - **Event ID** is the ID of the generalized event.
 - **Activations count** is the number of registered monitor activations on the sliding window.
 - **Number of attention subjects** is the number of attention subject parameter values whose detection activated the monitor.
 - **Event** is the detected generalized event.
 - **Attention subjects** are the attention subject parameter values whose detection activated the monitor.

This table is displayed only when the monitor is activated by generalized events.

- **Statistics on generalized patterns** is a table that contains information about generalized patterns:
 - **Pattern ID** is the ID of the generalized pattern.
 - **Activations count** is the number of registered monitor activations on the sliding window.
 - **Event count** is the number of events in the generalized pattern.
 - **Number of attention subjects** is the number of attention subject parameter values whose detection activated the monitor.
 - **Pattern duration** is the time interval between the first and the last event in a detected pattern. When a pattern is detected again, the Event Processor takes into account the [coefficient of allowed intervals dispersion](#) specified by the administrator for the events of a pattern.
 - **Pattern** is a detected generalized pattern.
 - **Attention subjects** are the attention subject parameter values whose detection activated the monitor.

This table is displayed only when the monitor is activated by generalized patterns.

You can view the histogram with a summary of activations on the **Histogram** tab, in the upper right corner of the section.

Creating a monitor

The functionality is available after [a license key is added](#).

To create a monitor:

1. In the [main menu](#), select the **Event Processor** → **Monitoring** section.
2. Click the **Create monitor** button.
The **Create monitor** pane appears on the right.

3. Specify the name of the monitor in the **Name** field.
4. To use the monitor to track parameter values, events, or patterns, set **State** to **Active**.
5. In the **Sliding window (sec.)** field, specify the interval (in seconds) from the current point in time back to the time sequence for which the monitor will process incoming values of parameters, events or patterns.
6. In the **Activation stack size** field, specify the number of monitor activations that must be displayed when [viewing information about the monitor](#).
7. Under **Subscription type**, select one of the following options:
 - To track the occurrence of certain event parameter values, select **Parameter values**.
 - To track the occurrence of certain events, select **Events**.
 - To track the occurrence of patterns in the behavior of the monitored asset, select **Patterns**.
 - To track unique generalized events or patterns, select **Unique generalized**.
 - To track similar generalized events or patterns, select **Similar generalized**.
8. In the **Activation parameters** block, do the following:
 - a. In the **Activation threshold** field, specify the number of monitor activations in the sliding window after which the monitor sends an alert to the external system.

This parameter is displayed if **Subscription type**, **Parameter values**, or **Events** is selected in the **Patterns** settings block.
 - b. To track new events, patterns, or event parameter values, set **Activation type** to **Track only new**.

This parameter is displayed if **Subscription type**, **Parameter values**, or **Events** is selected in the **Patterns** settings block.
 - c. In the **Attention head** drop-down list, select the attention head to focus the monitor's attention on the required directions in the behavior of the monitored asset.

This parameter is displayed if **Patterns**, **Unique generalized**, or **Similar generalized** is selected under **Subscription type**.
 - d. In the **Subscription to events or patterns** field, select one of the following options:
 - To track generalized events, select **Subscription to events**.
 - To track generalized patterns, select **Subscription to patterns**.

This parameter is displayed if **Unique generalized** or **Similar generalized** is selected under **Subscription type**.
9. To specify the conditions for activating the monitor when tracking event parameter values, events, or patterns, do the following under **Filters**:
 - a. Perform one of the following actions:
 - To track events for all specified values within a single monitor, set the toggle switch to **Track for all values simultaneously**.

- To create child monitors for each specified event parameter value selected from the **Event parameter** drop-down list, and track their occurrence separately, set the toggle switch to **Track for each value**.

The check box is displayed if **Events** is selected under **Subscription type**.

b. In the **Event parameter** drop-down list, select an event parameter to refine monitor activation conditions for.

c. In the **Filter type** drop-down list, select one of the following values:

- **Parameter**: to activate the monitor when tracking specific event parameter values.
- **Generalized parameter**: to activate the monitor when tracking generalized event parameter values.
This value can be selected if the monitor is tracking the occurrence of patterns.
- **Attention**: to focus the attention of the event processor on a specific event parameter.
This value can be selected if the monitor is tracking the occurrence of patterns.
- **Generalized attention**: to focus the generalized attention of the event processor on the selected parameter.
This value can be selected if the monitor is tracking the occurrence of patterns.


d. Perform one of the following actions:

- To include or generalize all values of an event parameter in attention, select **All values** from the **Value type** drop-down list.
- To include or generalize a specific event parameter value in attention, select **Specific values** from the **Value type** drop-down list and enter the relevant value in the **Value** field. As you start typing a value, all matching parameter values are displayed in the list.
- To include or generalize event parameter values according to a template in attention, from the **Value type** drop-down list, select **Regular expression** and enter the value template using a regular expression in **Value**.

You can use [special characters of regular expressions](#) to search for events and patterns based on regular expressions.

- To include or generalize the new values of an event parameter in attention, select **New values** from the **Value type** drop-down list.

This value type is available if the **Activation type** toggle switch is set to **Track only new**.

If necessary, you can specify more than one monitor activation condition. You can delete a previously added condition by clicking  next to the condition.

10. Click the **Save** button.

The new monitor is created and displayed on the **Monitoring** tab.

Editing a monitor

You can enable or disable the use of the monitor to track event parameter values, events, or patterns.

The functionality is available after [a license key is added](#).

To edit monitor:

1. In the [main menu](#), select the **Event Processor** → **Monitoring** section.
2. In the vertical menu ☰ of the monitor tile, select **Edit**.
The **Edit monitor** panel appears on the right.
3. Enter a new name for the monitor as needed.
4. Perform one of the following actions:
 - To start using the monitor to track event parameter values, events, or patterns, set **State** to **Active**.
 - To stop using the monitor to track event parameter values, events, or patterns, set **State** to **Inactive**.
5. Click the **Save** button.

Deleting a monitor

The functionality is available after [a license key is added](#).

To delete a monitor:

1. In the [main menu](#), select the **Event Processor** → **Monitoring** section.
2. In the vertical menu ☰ of the monitor tile, select **Delete**.
3. Confirm monitor deletion.

The monitor will be deleted.

Viewing the events history

Kaspersky MLAD lets you view the events that were received from external sources of events. To view events, you need to upload them to **Event Processor** → **Event history**.

The functionality is available after [a license key is added](#).



Kaspersky MLAD displays incoming events as a graph of relations between event parameters. The graph nodes correspond to the values of the event parameters, and the arcs between the nodes correspond to the links between the parameter values of incoming events. You can hover the mouse pointer over the event graph and view information about the event parameters and their values. You can also hover the mouse pointer over the event graph arc and view information about the number of links between the values of event parameters. The graph of event parameter relations is displayed on the **Graph** tab.

You can also view [information about the detected events](#)  as a table.

- **Event ID** is the ID of the detected event.
- **System parameters** contain the following information about the event:
 - **Last detection in interval** is the date and time when the event was last detected in the event stream during the specified period.
 - **Activations count in interval** is the number of event detections in the event stream during the specified period.
 - **Total activations** is the number of event occurrences in the event stream on the sliding window.
 - **Last activation** is the date and time when the event was last detected in the event stream.
 - **Parameter count** is the number of event parameters for which the values were received from the monitored asset.
- **Event parameters** are the values of the event parameters received from the monitored asset.

Each monitored asset has its own specific incoming events and event parameters. The list of event parameters is defined in the [configuration file for the Event Processor service](#). The configuration file is created and uploaded by a system administrator during [configuration of the Event Processor service](#).

To upload data for viewing incoming events:

1. In the [main menu](#), select the **Event Processor** → **Event history** section.
2. In the **Filters** section, click the  button to select the start and end date and time of the period for which you want to load and view events. To configure event parameters, do one of the following:
 - To load events based on the specific values of the event parameters, select the relevant event parameter value in the drop-down lists. As you start typing a value, all matching parameter values are displayed in the lists.
 - To load events based on a value template, click  in the event parameter cells, use the drop-down lists to enter the value template with the help of a regular expression, and select specified value template.
You can use [special characters of regular expressions](#) to perform a search based on regular expressions.

Each monitored asset has its own specific set and names of event parameters.

3. Click the **Process request** button.

Data on the events found by the application will be displayed as a graph in the central part of the page.

4. To view the received events as a table, select the **Table** tab.

The central part of the page displays a table that contains information on the detected events.

Viewing the pattern history


In the section **Event Processor** → **Patterns history**, you can find and view the structure of the new and/or persistently recurring patterns. The Event Processor generates patterns only for specific directions according to attention heads that are [defined in the attention configuration](#).


The functionality is available after [a license key is added](#).

You can also view the structure of the detected patterns down to the event level. The Event Processor represents patterns, events, and values of event parameters as a layered hierarchy of nested elements. For example, a fourth-layer pattern consists of subpatterns of the third layer. A third-layer pattern consists of second-layer patterns, and a second-layer pattern consists of events, which are first-layer elements. Event parameter values are elements of the null terminal layer.

Each monitored asset has its own specific incoming events and event parameters. The list of event parameters is defined in the [configuration file for the Event Processor service](#). The configuration file is created and uploaded by a system administrator during [configuration of the Event Processor service](#).

To view the registered patterns:

1. In the [main menu](#), select the **Event Processor** → **Patterns history** section.
2. In the **Filters** section, configure the following settings for displaying patterns on the page:
 - a. In the **History interval** drop-down list, click the  button to select the start and end date and time of the period for which you want to load and view patterns.
 - b. In the **Pattern type** drop-down list, select one of the following values:
 - **Stable** refers to patterns that were registered by the Event Processor service two or more times.
 - **New** refers to new patterns registered by the Event Processor service for the first time.
 - **All** includes all patterns that were registered by the Event Processor service.
 - c. From the **Attention head** drop-down list, select the specific attention head to examine for registered patterns.

You must select one of the attention heads that were defined when [configuring the attention settings](#).
 - d. To configure event parameters, do one of the following:
 - To view patterns based on specific values of the event parameters, select the event parameter values in the drop-down lists. As you start typing a value, all matching parameter values are displayed in the lists.
 - To view patterns based on a value template, click  in the event parameter cells, use the drop-down lists to enter the value template with the help of a regular expression, and select specified value template.

You can use [special characters of regular expressions](#) to perform a search based on regular expressions.

For the request to be processed correctly, enter the values for the event parameter that is receiving focused attention from the model. If an event parameter that is receiving focused attention has multiple values defined, the Event Processor will generate patterns for each value of the parameter.

Event parameters set as generalized in the selected attention head cannot be customized.

3. Click the **Process request** button.

The central part of the page displays a table containing [data on the registered patterns](#) .

- **Pattern ID** is the ID of the pattern. The number before the underscore at the beginning of a pattern identifier indicates the layer at which that pattern was detected.
- **Last detection in interval** is the date and time when the pattern was last detected in the event stream of the monitored asset during the specified period.
- **Activations count in interval** is the number of pattern detections in the event stream of the monitored asset during the specified period.
- **Event count** is the number of events in the pattern.
- **Last activation** is the date and time when the pattern was last detected in the event stream of the monitored asset or in the sleep mode.

4. To view the pattern structure, click the desired pattern row.

The page with [detailed information on the pattern](#)  opens.

- **Pattern ID** is the ID of the selected pattern. The number before the underscore at the beginning of a pattern identifier indicates the layer at which that pattern was detected.
- **Total activations** is the number of detections of the selected pattern in the event stream for the specified period.
- **Interval from previous item** is the time interval between the selected pattern and the pattern detected in the pattern sequence on the current layer before the selected pattern. Kaspersky MLAD displays the time intervals between the elements of the selected pattern when it is first detected. When a pattern is detected again, the Event Processor takes into account the [coefficient of allowed intervals dispersion](#) specified by the administrator for the elements of this pattern.
- **Event count** is the number of events in the pattern.
- **Pattern end time** is the end date and time of the selected pattern in the sequence of patterns on the current layer.
- **Last activation** is the date and time when the pattern was last detected in the event stream or in the sleep mode.
- **Patterns** is a tab that displays a table with information about the patterns included in the selected pattern. The following information is displayed on the **Patterns** tab:
 - **Pattern ID** is the ID of the subpattern. The number before the underscore at the beginning of a pattern identifier indicates the layer at which that pattern was detected.
 - **Pattern end time** is the end date and time of the subpattern in the sequence of patterns on the selected layer.
 - **Total activations** is the number of detections of the subpattern in the structure of the selected pattern.
 - **Event count** is the number of events in the subpattern.
 - **Interval from previous item** is the time interval between the subpattern and the previous pattern in the table. Kaspersky MLAD displays the time intervals between the elements of the subpattern when it is first detected. When a pattern is detected again, the Event Processor takes into account the [coefficient of allowed intervals dispersion](#) specified by the administrator for the elements of this pattern.
 - **Last activation** is the date and time when the subpattern was last detected in the sequence of patterns on the selected layer or in the sleep mode.
- **Events** is a tab that displays a table of events included in the selected pattern. The following data is displayed for each event:
 - **Event ID** is the ID of the event.
 - **System parameters** contain the following information about the event:
 - **Event date and time** is the date and time when the event is detected in the pattern structure.
 - **Interval from previous item** is the time interval between the current event and the previous event in the table. Kaspersky MLAD displays the time intervals between the events of the selected pattern when it is first detected. When a pattern is detected again, the Event

Processor takes into account the [coefficient of allowed intervals dispersion](#) specified by the administrator for the events of this pattern.

- **Total activations** is the number of the event repeated occurrences in the structure of the selected pattern during the specified period.
- **Parameter count** is the number of event parameters for which the values were received from the monitored asset.
- **Last activation** is the date and time when the event was last detected in the event stream.
- **Event parameters** are the values of the parameters of the event received from the monitored asset.

5. To view the structure of a pattern, do one of the following:

- To view the structure of a particular subpattern, on the **Patterns** tab in the **Nested elements** section, click the desired pattern.

You can return to viewing the top-level pattern structure by clicking the ID of the desired pattern above the **Pattern info** section.

- To view the events included in the pattern at the second nesting level, click the **Events** tab.

Kaspersky MLAD displays the pattern structure from the top nesting level.

Working with incidents and groups of incidents

The Kaspersky MLAD web interface provides the capability to investigate registered incidents. Depending on the [type of the registered incident source](#), information about the incident and the methods you can use to investigate it may differ.

The functionality is available after [a license key is added](#).

You can perform the following actions for any incident:

- [Analyze the incident details](#).
- [Find out if any similar incidents were detected previously](#).
- [Study the behavior of the monitored asset at the moment when the incident was detected](#).
- [Leave a note or expert opinion for a registered incident or incident group](#).

The **Incidents** section displays a column graph showing the incidents that match the filtering criteria specified under the graph. The graph displays statistics on the registered incidents for the period specified above the graph.

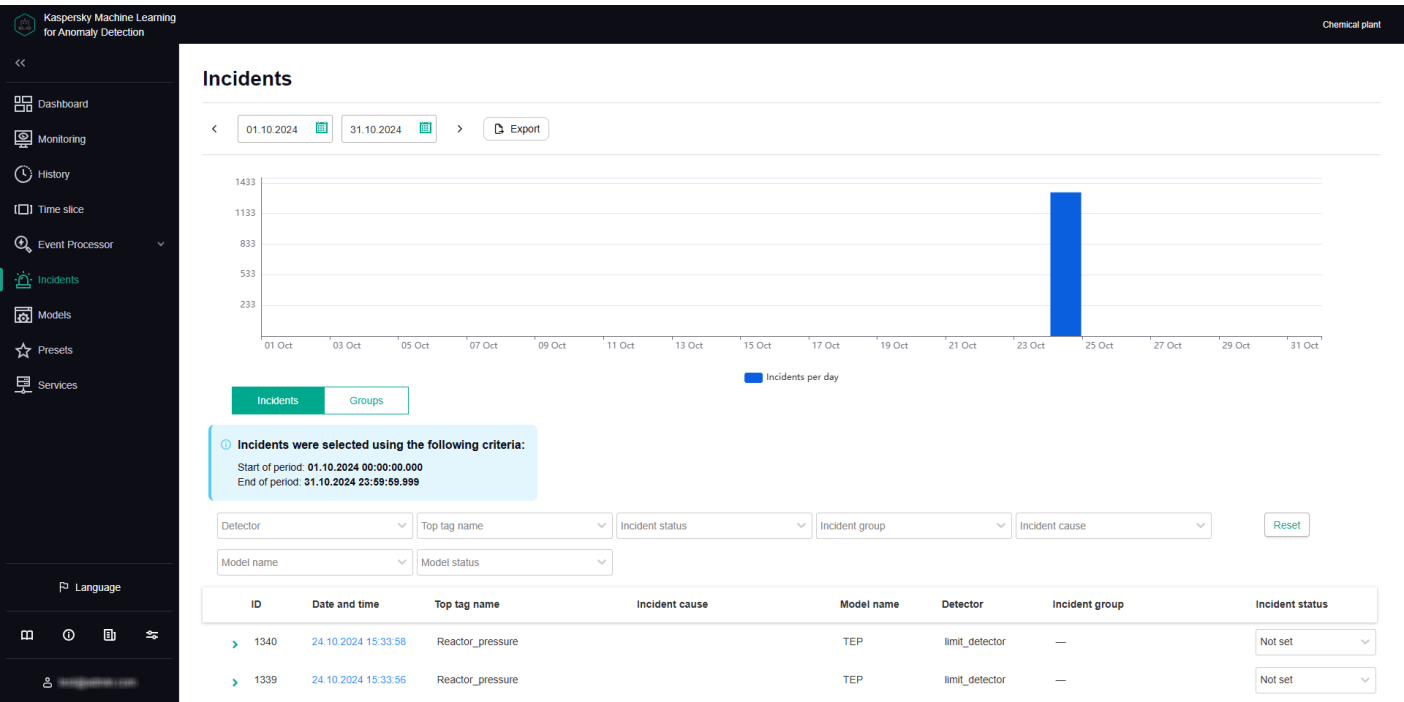
The graph can display up to 60 bars. If the specified period does not exceed 60 days, incidents on the graph are grouped by days. If the specified period is between 60 days and 60 weeks, incidents on the graph are grouped by weeks. If the specified period is longer than 60 weeks, incidents on the graph are grouped by months.

Hovering the mouse pointer over a bar of the graph displays a window showing the number of registered incidents per corresponding time period. Upon clicking a bar, the graph and in the table below display information about the incidents registered during the corresponding time period.

In this section, you can view individual incidents as well as groups of incidents.

Incidents tab

The **Incidents** tab shows a table of registered incidents. Incidents are sorted by date in descending order, with the newest incidents shown first.



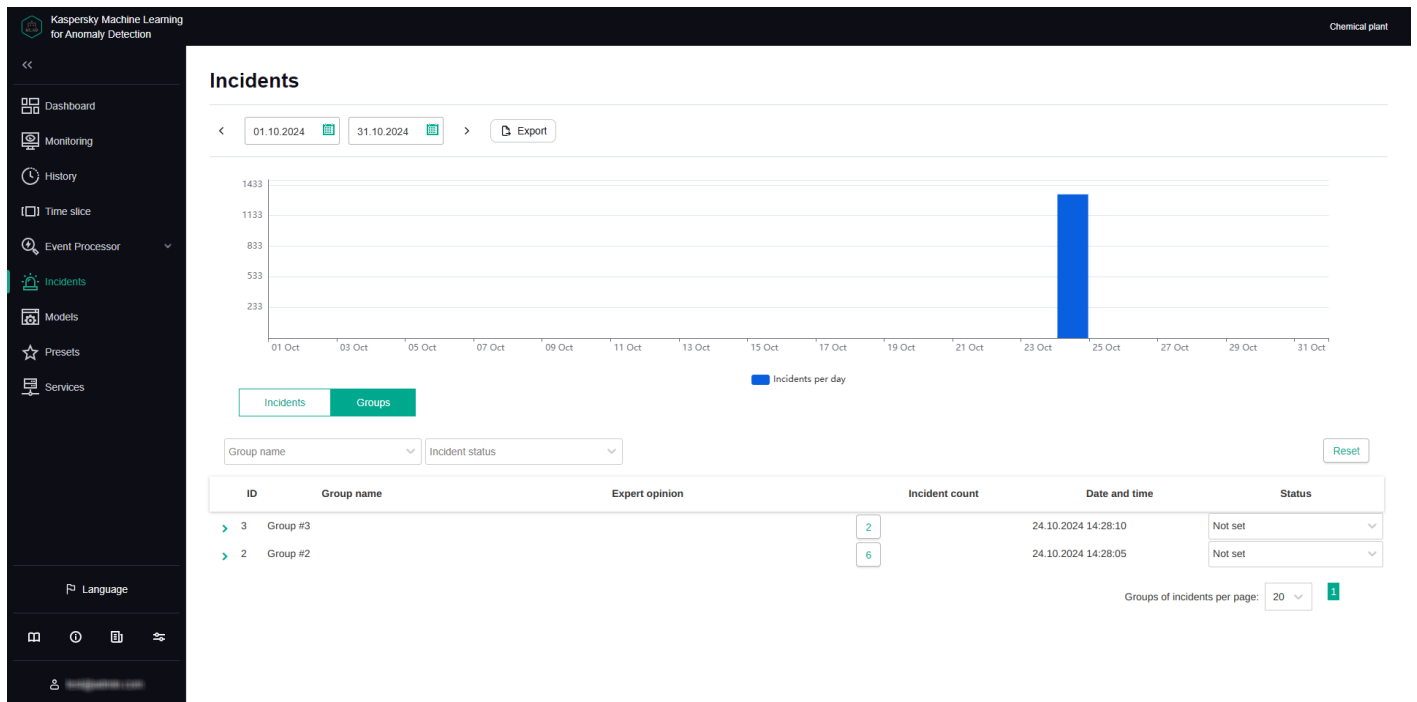
Incidents tab

You can go to the **History** section by clicking the date and time of the incident.

Groups tab

The **Groups** tab shows a table of incident groups. Kaspersky MLAD automatically generates groups of similar incidents.

You can change the group name that was assigned automatically and set the status of incidents that belong to this group. You can also [provide an expert opinion](#) that contains the recommended actions to take in response to new incidents in this group, for example.



Groups tab

About incidents

An *incident* is an identified deviation from the expected (normal) behavior of a monitored asset.

Kaspersky MLAD supports incident registration for the following sources:

- **ML model elements.** An incident is recorded when the value of an ML model element artifact reaches the incident registration threshold set for that specific element. Incidents can be detected by [predictive](#), [diagnostic rule-based](#), and [elliptic envelope-based](#) ML model elements.

The registration of an incident may be delayed after the specified threshold is reached, depending on the configuration of the ML model element. When a certain threshold for registering an incident is met, the ML model element starts watching for unusual activity in the monitored asset for a specific period defined in the element settings. If the monitored asset exhibits anomalous behavior for a certain specified proportion of this interval, the ML model element registers an incident.

Kaspersky MLAD can suppress the registration of consecutive incidents from the same ML model element if these occur shortly after the first incident in the series. This uses **Period of recurring alert suppression (sec)**, which you can define in the element settings.

An ML model element may continue to flag the same incident multiple times until the abnormal activity stops. You can control how often repeated incidents are recorded by adjusting the **Reminder period (sec)** and editing the element.

- [Limit Detector](#). The system will record an incident whenever a tag value reaches the upper or lower limit.
- [Stream Processor](#). An incident is recorded when the system notices that telemetry data is missing or when Kaspersky MLAD receives observations too early or too late.

When a deviation is detected, the corresponding source records the date, time and relevant deviation parameters, and saves this data as an entry in the [Incidents section](#). If incident notifications for users or external systems are created in Kaspersky MLAD, information about an incident is sent to the intended recipients via the corresponding services of Kaspersky MLAD.

About incidents detected by a predictive element of an ML model

An [ML model predictive element](#) has been trained on a particular subset of tags, and it can anticipate current tag behavior. In this case, an incident is any substantial discrepancy between the observed (actual) values of tags and the predicted values of tags resulting from operations of the ML model element. In the model element settings, you can view which tags are analyzed by the ML model element (**Input tags** parameter) and which tags' behavior is predicted (**Output tags** parameter).

An [ML model](#) can include one or more elements running in parallel. In the **History** and **Monitoring** sections, you can [select a specific element of the ML model](#) to display the incidents registered as a result of a specific model element operation. The graph of the ML model element artifact shows registered incidents as colored dots at the bottom.

The artifact graph also displays cumulative prediction error value for the selected ML model element. *Cumulative prediction error* is an indicator of the difference between predicted values from actual values, calculated cumulatively for all tags included in the selected element of the ML model. The higher the prediction error value, the more the behavior of tags will differ from the expected (normal) behavior. *The prediction error threshold* is the critical cumulative prediction error value that, when reached, causes the ML model predictive element to register an incident. The artifact graph displays the cumulative prediction error as a red line, and the prediction error threshold, as an orange line. The area on the graph where the forecast error exceeds the specified threshold is colored red.

The ML model artifact graph is displayed at the bottom of the **History** section (see figure below).



ML model element artifact graph under History

For each incident, the application automatically identifies tags with the greatest actual value deviations from those predicted by the ML model. These tags generate a **Tags for incident #<incident ID>** preset. This preset is displayed under **History** when you click the incident date and time in the [incidents table](#). Tags that are included in the **Tags for incident #<incident ID>** preset are sorted in descending order of their deviation from expected (normal) behavior. The tag with the greatest deviation from the predicted value is displayed in the incidents table under **Incidents**. The incidents table also indicates the prediction error threshold and the actual prediction error value at the moment when the incident was registered.

Information obtained when viewing the **Tags for incident #<incident ID>** preset is not actually diagnostic information for the purposes of identifying the causes of an incident, but you can still use this information when analyzing the values of tags with the largest deviations in behavior. The tag whose behavior was the first to deviate from the norm and caused subsequent deviations in other tags is referred to as the causal tag. In some cases, the causal tag may not be at the top of the list in the **Tags for incident #<incident ID>** preset and may even be entirely absent from this preset. This could happen due to the following reasons:

- Minor amplitude changes in the behavior of the causal tag had a multiplier effect and caused significant deviations in other tags that were included in the **Tags for incident #<incident ID>** preset.
- The causal tag is not analyzed by the ML model, and Kaspersky MLAD registers derivative changes in the behavior of tags caused by the deviation of the causal tag.
- Changes in the behavior of the causal tag had a delayed effect, and by the time an anomaly occurred in the operation of the monitored asset, the behavior of the causal tag returned to normal.

About incidents detected by an ML model element based on a diagnostic rule

An ML model can include one or more [elements based on diagnostic rules](#). Each diagnostic rule results in the following values being obtained that are calculated at each point in time:

- Value 0. The diagnostic rule was not triggered or applied at this moment.
- Value 1. The diagnostic rule was triggered at this moment.
- Intermediate values from 0 to 1 are possible in individual cases. The diagnostic rule was partially triggered at this moment.

Once the result surpasses the threshold set for the diagnostic rule, which is generally equal to one, the element based on the diagnostic rule records an event. For each incident registered by the diagnostic rule, Kaspersky MLAD automatically creates a **Tags for incident #<incident ID>** preset. This preset can be selected under **History** when you click the incident date and time in the [incidents table](#). This preset contains the value obtained as a result of the work of the diagnostic rule, as well as the tags included in this rule.

About incidents detected by an ML model element based on an elliptic envelope

An [ML model elliptic envelope](#) is trained on a specific subset of tags, and it can detect outliers (anomalies) in a dataset. The training of the ML model creates an elliptical region within the phase space. Any data points that fall within this ellipse are considered normal. When states are detected that are a distance from the center of the elliptical region equal to or greater than the predetermined threshold, the element based on the elliptic envelope registers an incident. In the model element parameters, you can view which tags are parsed by the element (**Input tags**).

The most relevant tags are automatically defined for every incident registered by an element based on an elliptic envelope. These are tags whose removal from the ML model causes the least deviation from the normal state. These tags generate a **Tags for incident #<incident ID>** preset. The preset can be selected under **History** when you click the incident date and time in the [incidents table](#). Tags that are included in the **Tags for incident #<incident ID>** preset are sorted in descending order of their deviation from expected (normal) behavior. The tag with the greatest impact on incident registration is displayed in the [incidents table](#) under **Incidents**.

An ML model may include one or more elements running in parallel. In the **History** and **Monitoring** sections, you can [select a specific element of the ML model](#) to display the incidents registered as a result of a specific model element operation. The graph of the ML model element artifact shows registered incidents as colored dots at the bottom.

About incidents detected by the Limit Detector

If the [Limit Detector is enabled](#), Kaspersky MLAD automatically monitors all tags having blocking thresholds specified for the tag when using any [ML model](#). Blocking thresholds can be defined in a [tag configuration imported into Kaspersky MLAD](#) at the start of operations. You can [edit the tag blocking thresholds](#) in the application's web interface.

To visually monitor the position of a tag graph relative to the blocking thresholds in individual graphic areas under [History](#) and [Monitoring](#), you can turn on **Blocking threshold** and **Always display blocking threshold**. If **Always display blocking threshold** is disabled, the upper or lower threshold line is displayed only if the tag values have reached the corresponding threshold during the time interval displayed on the screen. The Limit Detector identifies and registers events regardless of whether or not the **Always display blocking threshold** option is enabled.

When the tag value reaches its upper or lower technical limit, the Limit Detector registers an incident. This tag is displayed in the [incidents table](#) in the **Incidents** section. The incidents table also shows the blocking thresholds of the tag and the actual value of a tag that violated one of these limits. For each incident registered by Limit Detector, Kaspersky MLAD automatically creates a **Tags for incident #<incident ID>**. This preset can be selected under **History** when you click the incident date and time in the incidents table. This preset includes the only causal tag of the incident.

About incidents detected by the Stream Processor service

The *Stream Processor* service gathers real-time telemetry data received from the monitored asset at arbitrary points in time and converts this data to a uniform temporal grid (UTG). When analyzing incoming data, the Stream Processor service can detect losses of telemetry data and observations that were received by Kaspersky MLAD too early or too late. The Stream Processor service registers an incident in such cases.

Incidents detected by the Stream Processor service are displayed in the [incidents table](#) of the **Incidents** section. Each incident registered by the Stream Processor service is automatically assigned one of the following incident types:

- **No data** – input data stream for a specific tag was terminated or interrupted.
- **Clock malfunction** – observations received by Kaspersky MLAD too early are detected.
- **Late receipt of observation** – observations received by Kaspersky MLAD too late are detected.

The Stream Processor service transfers the UTG-converted data to the [ML model](#) of the Anomaly Detector service.

About anomalies

An *anomaly* is any deviation in a monitored asset's behavior that is abnormal, not provided for by the current work procedure, and not normally caused by the industrial process.

Kaspersky MLAD registers only [incidents](#). A specific incident can be identified as an anomaly only by an ICS specialist after [conducting an analysis of incidents registered by the application](#). An incident analysis may result in one of the following conclusions:

- The incident is an anomaly that requires certain actions from a responding ICS specialist.
- The incident is not actually an anomaly, but instead was a false positive by the ML model.

If an ML model consistently produces false positive results, you need to find out what is causing the decline in performance, adjust the settings of the ML model and/or its elements, or further train the elements.

- The ML model worked correctly, but the incident is not an anomaly.

The incident was a result of temporarily switching the monitored asset to a non-standard operating mode (preventative maintenance or testing) or was caused by short-term impacts from non-standard external factors (unusual weather conditions or startup of a neighboring unit). The ICS operator does not need to take any response action.

Incidents are analyzed and assessed by a subject-matter expert. In some cases, like when registering incidents detected by diagnostic rules or incidents that occur repeatedly, [similar incidents can be automatically grouped](#) and assessed.

The ML model might miss a real anomaly. In this case, the anomaly will not be correlated to any registered incidents and will not be reflected in the Kaspersky MLAD history. If observations from an expert, an ICS operator, or external sources reveal repeated instances of an ML model failing to activate, you need to identify the cause of the decline in performance, adjust the settings of the ML model and/or its elements, and further train the elements of the ML model.

New [events](#) [@](#) [patterns](#) [@](#) and values of the event parameters detected by the Event Processor service in the stream of incoming events can also indicate an anomaly in the operation of a monitored asset. When new events, patterns or values of event parameters are detected, the Event Processor service does not register incidents. To view new detections in the **Event Processor** section, you can [view the history of registered patterns](#), filtering them by the **New** type. You can also [create a monitor](#) for tracking new events, patterns, or values of event parameters. The Event Processor service activates the monitor when it detects events, patterns, or event parameter values that match the specified search criteria. When the specified threshold for the number of monitor activations in a sliding window is reached, the Event Processor service sends an alert about the monitor activation to the external system using the CEF Connector.

Scenario: analysis of incidents

This section describes the sequence of actions required when analyzing incidents registered by Kaspersky MLAD.

The functionality is available after [a license key is added](#).

The incident analysis scenario described in this section is not a precisely regulated procedure. The specific scope and sequence of actions taken to investigate an incident and identify its cause depend on the particular subject area, the knowledge level of the process engineer or ICS expert investigating the incident, and the availability of additional information on the monitored asset.

The incident analysis scenario consists of the following steps:

1 Viewing information about a registered incident

The [Incidents](#) section displays all incidents registered by Kaspersky MLAD, and provides detailed information about their registration time, the ML model that registered the incident, and an expert opinion if one was added. You can proceed to view incident information in one of the following ways:

- **Viewing the latest incidents in the Dashboard section**

If you want to view a recently detected incident, in the [Dashboard](#) section, click the date and time of the relevant incident in the **Latest incidents** table. In the [History](#) section that opens, in the lower part of the page, click the dot indicator in the artifact graph section to view a specific incident. The [Incidents](#) section opens showing only the incidents that were registered in the specific time interval represented by the selected dot indicator (the interval is displayed above the incidents table).

- **Viewing incidents in the Incidents section**

If you know the date and time when an incident was registered, [select the corresponding incident in the Incidents](#) section. You can change the time interval for the displayed incidents by using the bar graph or the date selection field in the upper part of the page.

- **Navigating from an incident notification received by email**

If an incident [notification was created](#) for you, you will receive the notification by email when an incident is registered. The email message contains the time when the incident began, the top tag, and a link to proceed to the [History](#) section in the Kaspersky MLAD web interface. You can use this link to proceed to the start of the incident in the **History** section. At the bottom of the **History** page, click the dot indicator that corresponds to the incident start time. The [Incidents](#) section opens showing only the incidents that were registered in the specific time interval represented by the selected dot indicator (the interval is displayed above the incidents table).

When you find a record about the required incident, click the [➤](#) button to [view detailed information about the incident](#).

2 Viewing information about similar incidents

When two or more similar incidents are detected, Kaspersky MLAD automatically combines them into a group. In the [incidents table](#) in the **Incidents** section, the group associated with the incident is displayed in the **Incident group** column. If nothing is indicated for the selected incident in this column, this means that Kaspersky MLAD has not yet detected incidents similar to this particular incident.

To [view all incidents in a group](#), select the **Groups** tab and click the [➤](#) button next to the relevant group. The table displays information about the incidents assigned to the selected group, as well as an expert opinion if it was added. Read the expert opinions for individual incidents and for the group.

3 Studying the behavior of the monitored asset at the moment when an incident was detected

[Study the behavior of the monitored asset](#) at the moment when the incident was detected.

4 Analyzing the incident

Analyze the incident while considering the specific details of incident registration depending on the type of the source that registered the incident:

- **Forecaster.** A predictive element of the ML model registers incidents when there is a significant discrepancy between observed (actual) tag values and predicted tag values. Based on information obtained when viewing the automatically generated **Tags for incident #<incident ID>** preset and considering the available expert knowledge on the monitored object, form a hypothesis regarding which tags could have caused the anomaly and select the appropriate preset after studying their behavior. Analyze the graph of the ML model element artifact, move back in time from the moment the prediction error threshold was reached, and examine the behavior of tags at the moment when the prediction error values started to grow.
- **Rule Detector.** For each incident registered by an ML model element based on a diagnostic rule, the application automatically creates the **Tags for incident #<incident ID>** preset, which includes the value obtained as a result of the diagnostic rule operation and which caused the incident registration.

- **Elliptic envelope.** An ML model elliptic envelope records incidents whenever it detects states that are a distance from the center of the normal state cluster equal to or greater than a predefined threshold. When registering an incident, the application generates a **Tags for incident #<incident ID>** preset that includes the tags whose exclusion from the ML model results in the smallest deviation of observations from the normal state. Analyze the graph of the ML model element artifact, move back in time from the moment the threshold was reached, and examine the behavior of tags at the moment when the deviation started to grow.
- **Limit Detector.** For each incident that was registered by the Limit Detector, the application automatically creates the **Tags for incident #<incident ID>** preset, which includes a single causal tag for the incident.
- **Stream Processor.** The Stream Processor service registers incidents up until telemetry data is transmitted to the ML model for processing. Incidents are registered if data loss is detected or if observations are received by Kaspersky MLAD too early or too late.

5 Adding a status, cause, expert opinion or note to an incident or its incident group

For each incident, [add an expert opinion or note](#) in which you can specify whether the incident is an [anomaly](#). An expert opinion and note for an incident are displayed only when viewing a specific incident. If necessary, you can specify the status and cause of an incident. The cause of an incident is displayed in the incidents table and when viewing a specific incident. You can also add or edit the status and expert opinion for a group of incidents.

If you know in advance the expert opinion, cause, and/or status of incidents registered by a specific ML model element, you can enter that information in the element parameters. The expert opinion, reason, and/or status will be automatically assigned to incidents at the time of their registration by the element.

Viewing incidents

The functionality is available after [a license key is added](#).

To view incidents that were registered during a specific period:

1. In the [main menu](#), select the **Incidents** section.
2. In the upper part of the opened page, select the start and end dates of the period.
By clicking a bar in the bar chart, you can also refine the time period for which incidents are displayed. The column can represent a month, week, or day, depending on the length of the period set above the chart.
3. If necessary, filter incidents according to the top tag names, incident groups, statuses as well as causes, names and statuses of the ML models that registered the incidents by selecting the values from the appropriate drop-down lists.

The table located in the central area of the page shows the incidents registered during a specific period according to the specified filtering criteria. When you click the **Reset** button, the table and the bar graph show all registered incidents.

The following information is displayed for each incident in the table:

- **ID** refers to the ID of the registered incident.
- **Date and time** refers to the date and time when the incident was registered.
Clicking the incident registration date and time opens the **History** section, where you can view information about the **Tags for incident #<incident ID>** preset generated for the registered incident.
- **Top tag name** is the name of the process parameter that had the greatest impact on incident registration.

- **Incident cause** refers to the cause of the registered incident as [entered by an expert](#) (ICS process engineer or operator) as a result of an incident analysis or assigned automatically according to the incident cause specified for the ML model element that registered the incident.
- **Model name** refers to the name of the ML model whose element registered the incident. This is absent if the incident was registered by Stream Processor.
- **Detector** refers to the type of the registered incident: [Elliptic Envelope](#), [Forecaster](#), [Limit Detector](#), [Rule Detector](#), or [Stream Processor](#).

- **Incident group** refers to the name of the incident group to which the registered incident belongs.


If two or more similar incidents are detected, they are combined into a group that is created automatically by using the [Similar Anomaly service](#). You can view incidents that belong to a particular group by selecting the group name from the **Incident group** drop-down list above the incidents table.

- **Incident status** refers to the status of the registered incident as [entered by an expert](#) (ICS process engineer or operator) as a result of an incident analysis or assigned automatically according to the incident status specified for the ML model element that registered the incident.

You can set the incident status based on analysis results by selecting the appropriate value from the drop-down list. After installation of Kaspersky MLAD, the following statuses of incidents and incident groups are available by default: **Under review**, **Decision pending**, **Instructions issued**, **Problem closed**, **Cause unknown**, **Ignore** and **False positive**. If necessary, the system administrator can [create, edit, or delete statuses of incidents](#).

Viewing the technical specifications of a registered incident

The functionality is available after [a license key is added](#).

In the **Incidents** section, you can view the technical specifications of registered incidents. To do so, click the  button near the relevant incident in the incidents table. The following technical specifications will be displayed for the selected incident:

- **Incident** is the section containing [information about the incident](#) .

- **Model name** refers to the name of the ML model whose element registered the incident. This is absent if the incident was registered by Stream Processor.
- **Model element** refers to the name of the ML model element that registered the incident. This is absent if the incident is registered by Limit Detector or Stream Processor.
- **Detector** refers to the type of the registered incident: [Elliptic Envelope](#), [Forecaster](#), [Limit Detector](#), [Rule Detector](#), or [Stream Processor](#).
- **ML model element artifact value** refers to deviation of the monitored asset's behavior from normal at the time of incident registration. This is absent if the incident is registered by Limit Detector or Stream Processor.
- **Threshold value** refers to the specific value at which the ML model element registered the incident. For any incident detected by Limit Detector, the specific threshold (upper or lower) reached by the tag is recorded.

- **Top tag** is a section that contains [information about the tag](#) that had the greatest impact on incident registration.

- **Top tag name (top tag ID)** is the name and ID of the tag that had the greatest impact on incident registration.

If the incident has been registered by a [predictive element of the ML model](#), the application displays the name of the tag for which the greatest deviation from the forecast was recorded. If the incident is registered by an [elliptic envelope](#), the application displays the name of the tag whose exclusion from the ML model results in the smallest deviation of the observation from the normal state. If the incident is registered by a [Limit Detector](#), the application displays the tag whose value exceeded the blocking threshold defined for this tag.

- **Top tag value** is the value of the top tag registered when the incident occurred.

- **Blocking threshold** refers to maximum permissible top tag values.

Limit Detector requires these settings to function correctly. Whenever the tag value reaches its upper or lower blocking threshold, the Limit Detector registers an incident.

- **Description** refers to a description of the top tag.

- **Measurement units** refer to the units for measuring the top tag values.

- **Stream Processor service incident parameters** is a section containing [information about the parameters of the incident registered by the Stream Processor service](#). This group of parameters is displayed if the current incident is registered by the Stream Processor service.

- **Incident type** is the type of incident registered by the Stream Processor service. The Stream Processor service registers incidents when it detects observations that were received too early or too late, or if the incoming data stream from a certain tag is terminated or interrupted.

- **Data date and time** is the date and time when the observation was generated according to the monitored asset time. This parameter is displayed only for the **Late receipt of observation** and **Clock malfunction** incident types.

- **Lag / Lead** is the amount of time by which the observation generation time lags behind or is ahead of the time the observation was received in Kaspersky MLAD. If data is received too early, the parameter value is displayed with a plus sign (+). If data is received too late, the parameter value is displayed with a minus sign (-). This parameter is displayed only for the **Late receipt of observation** and **Clock malfunction** incident types.

- **Incident cause** is the field for selecting the cause of the incident. This field is [completed by an expert](#) (process engineer or ICS specialist). If necessary, the system administrator can [create, edit, or delete causes of incidents](#).

An incident cause can be assigned automatically if a cause is specified in the parameters of the ML model element that registered the incident.

- **Expert opinion** is the field for adding an expert opinion based on an analysis of the registered incident. This field is [completed by an expert](#) (process engineer or ICS specialist).

An expert opinion can be assigned automatically if an opinion is specified in the parameters of the ML model element that registered the incident.

- **Note** is the field for entering a comment for the selected incident. If necessary, you can [provide a comment for the incident](#).

Viewing incident groups

When two or more similar incidents are detected, Kaspersky MLAD automatically combines them into a group (using the [Similar Anomaly service](#)). This lets you analyze incidents with consideration of prior history and expert opinions that were generated for similar incidents. In the [incidents table](#) in the **Incidents** section, the group associated with the incident is displayed in the **Incident group** column. If nothing is indicated for the incident in this column, this means that Kaspersky MLAD has not yet detected incidents similar to this particular incident. Incidents can be regrouped, and the expert opinions that were added to these incidents are migrated to the new group. The group name is automatically assigned in the format Group #N (N is replaced by the sequence number of the group). If necessary, you can [edit a group name](#).

The functionality is available after [a license key is added](#).

To view incident groups:

In the [main menu](#), select the **Incidents** section and select the **Groups** tab.

All incident groups for your monitored asset are displayed in the table located in the central part of the page.

The following information is displayed for each incident group in the table:


- **ID** is the incident group identifier.
- **Group name** refers to the name of the incident group.
- **Expert opinion** is a conclusion [added by an expert](#) (process engineer or ICS specialist) based on an analysis of the group of registered incidents.
- **Incident count** refers to the number of registered incidents included in the group.

You can proceed to view incidents of the group by clicking its incidents count.

- **Date and time** refers to the date and time when the incident group was created.
- **Status** refers to the status of the registered incidents in the group as [entered by an expert](#) (ICS process engineer or operator) as a result of an incident analysis or assigned to the incidents automatically according to the incident status specified for the ML model elements that registered the incidents.

You can set the incident group status based on analysis results by selecting the appropriate value from the drop-down list. After installation of Kaspersky MLAD, the following statuses of incidents and incident groups are available by default: **Under review**, **Decision pending**, **Instructions issued**, **Problem closed**, **Cause unknown**, **Ignore** and **False positive**. If necessary, the system administrator can [create, edit, or delete statuses of incidents](#).

To view detailed information about an incident group:

1. Click the  button near the incident group.

A list of incidents in this group is displayed. The following technical specifications are displayed for each incident of the group:

- **Incident date** is the date and time when the incident was registered.

You can go to the **History** section by clicking the incident registration date.

- **Top tag name** is the name of the tag that had the greatest impact on [incident registration](#).
 - **Top tag value** is the value of the tag that had the greatest impact on incident registration.
 - **Relevant tags** refers to a table that contains the identifiers of tags that influenced the identification of similar incidents and merging of these incidents into a group.
2. If you need to view the degree of influence a tag had on the formation of similar incidents, click the **Relevant tags** table cell containing the identifier of the relevant tag.

All table cells containing the selected tag ID are highlighted in green. The closer the green-highlighted cells containing the ID of the selected tag are to the first table column, the more impact that tag has when identifying and grouping similar incidents.

You can also [add a status and expert opinion for the incident group](#).

Studying the behavior of the monitored asset at the moment when an incident was detected

This section describes the sequence of actions required when studying the behavior of a monitored asset at the moment when an incident was detected.

The functionality is available after [a license key is added](#).

Studying the behavior of a monitored asset consists of the following steps:

1 Viewing the history of tags received for a monitored asset in the History section

You can proceed to view incident information in one of the following ways:

- If you want to view a recently detected incident, in the [Dashboard](#) section, click the date and time of the relevant incident in the **Latest incidents** table.
- In the **Incidents** section, click the date and time of the relevant incident in the [incidents table](#).
- If an incident [notification was created](#) for you, you can proceed to view the incident by clicking the link from the email notification. The email message contains the time when the incident began, the top tag, and a link to proceed to the [History](#) section in the Kaspersky MLAD web interface.

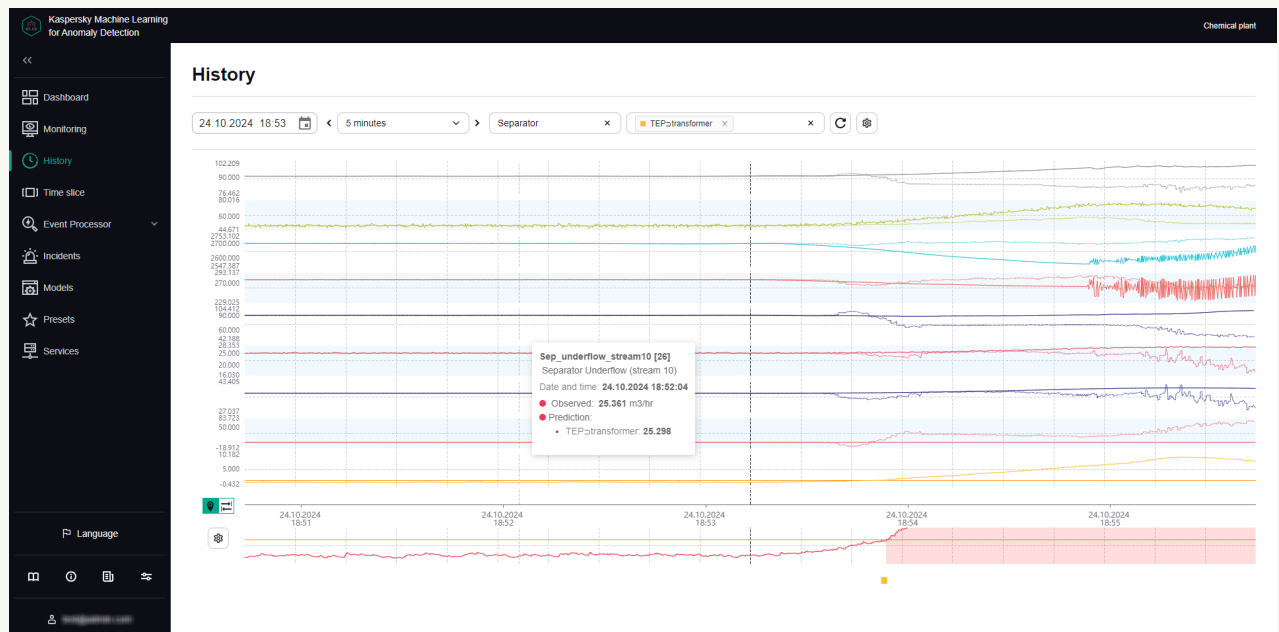
In the **History** section, Kaspersky MLAD displays graphs of tags received from the monitored asset for which the selected incident was registered. The graphs display data for the **Tags for incident #<incident identifier>** preset, generated for the date and time when the selected incident was registered. This preset includes the tags whose behavior led to incident registration. Depending on the type of the source that registered an incident, this may involve the following tags:

- The tags for which the actual values showed the greatest deviations from the ML model's forecast, given that the incident was registered by the ML model predictive element.
- Tags included in a diagnostic rule and the value obtained as a result of the operation of this rule, if the incident was registered by the ML model element based on the diagnostic rule
- The tags whose removal from the ML model results in the least deviations of observations from the normal state, given that the incident was detected by the ML model element based on the elliptic envelope.
- A tag whose value was outside of the set blocking thresholds, if the incident was registered by the Limit Detector.

If necessary, you can [select a different preset](#) for displaying data received from the monitored asset at the moment when the incident was registered. The graph uses a vertical blue dashed line to indicate the date and time when the incident was registered.

Example tag graphs for a registered incident under History.

The tag graphs are displayed in the upper part of the **History** section. The graph of the ML model element artifact is displayed in the lower part of the **History** section.



Tag graphs in the History section

2 Configuring how data is displayed on graphs in the History section

Under **History**, you can [turn on the display of predicted tag values](#) generated by the predictive elements of the ML model. This lets you assess the difference between actual tag values and predicted tag values. Hovering over a tag graph displays tag details, such as the name, description, date and time when it was observed, value, and unit of measurement. You can also [enable display of the tag name and description](#) on the left of each tag graph.

3 Configuring the time settings for displaying data in the History section

When studying the behavior of tags, you can [change the scale of the time axis](#) or [move forward or backward in time through graphs](#). When displaying shorter time intervals on tag graphs, the **History** section may show more details of the behavior of tags that had been averaged when tag graphs for a longer period were displayed.

4 Changing the vertical boundaries for displaying data in the History section

When displaying single graphic areas, the default vertical scale of the graph is automatically determined according to the minimum and maximum tag values within the displayed area. If minimum and maximum permissible values (blocking thresholds) are defined for a tag, you can control graph scale along the vertical axis by enabling [Always display blocking threshold](#). If a tag value is within the permissible range, the vertical scale of the graph will be fixed by limit lines derived from the lower and upper thresholds of the tag graph. If the tag values go beyond the specified blocking thresholds, the vertical scale will be automatically changed to display the tag values exceeding the thresholds.

If graphic areas are displayed for several tags, you can adjust their vertical scale by using the parameters of the corresponding graphic area, which you can set when [editing the selected preset](#).

Adding a status, cause, expert opinion or note to an incident or incident group

Kaspersky MLAD lets you add an expert opinion or note to a registered incident.

The functionality is available after [a license key is added](#).

An expert opinion is normally added by an expert (process engineer or ICS specialist) and may contain an incident analysis or recommendations on resolving a problem that is indicated by an identified incident. An expert opinion can be added to an individual incident or to a group of incidents. If expert opinions were previously added to incidents that are later put into a group, these opinions will also be displayed in the group (linked to each specific incident). When incidents are regrouped, the expert opinion for an incident migrates together with the incident to the new group.

Notes are intended to aid discussions between experts or operators of facilities regarding recommended actions for analysis, investigation, and remediation of an incident. Each note includes information stating who added the note and when it was added.

You can also add the cause of the incident and the incident status determined by the expert based on the incident analysis results. A status can be assigned to an individual incident or to a group of incidents. When changing the status of a group of incidents, Kaspersky MLAD changes the status of the incidents that are part of this group. The status of an incident also affects whether a dot indicator for it will be displayed under **Monitoring** and **History** and whether an incident notification with this status will be sent. If the **Notify about an incident** check box is cleared for the incident status, the incident dot indicators to which this status was assigned automatically will not be displayed under **Monitoring** or **History**, and no email notifications about incidents will be sent. An incident status can be assigned automatically in one of the following cases:

- If the incident was automatically assigned to a group with that status.
- If the incident is registered by an ML model element that sets that incident status by default.

For the **Problem closed** and **Ignore** statuses, the **Notify about an incident** check box is cleared by default. If during registration, incidents are automatically assigned one of these statuses in accordance with the status specified for the ML model element that registered this incident, notifications about these incidents will not be sent.

If you know in advance the expert opinion, cause, and/or status of incidents registered by a specific ML model element, you can enter that information in the element parameters. The expert opinion, reason, and/or status will be automatically assigned to incidents at the time of their registration by the element.

Before adding a cause, status, note or expert opinion, you must conduct an [analysis of the registered incident](#).


To add an expert opinion, status, cause, or note to an incident:

1. In the [main menu](#), select the **Incidents** section.
2. If necessary, change the incident status by selecting one of the following statuses from the **Incident status** drop-down list: **Under review**, **Decision pending**, **Instructions issued**, **Problem closed**, **Cause unknown**, **Ignore**, or **False positive**.

By default, an incident is assigned the **Unknown** status. If necessary, the system administrator can [create, edit, or delete statuses of incidents](#).

3. To display detailed technical specifications of an incident, click the ► button near the relevant incident. In the details area that opens, you can do the following:

- If you need to add the cause of an incident, use the **Incident cause** field to select the cause of the incident. If necessary, the system administrator can [create, edit, or delete causes of incidents](#).

- If you need to add an expert opinion based on an analysis of a registered incident, click the  button on the right of the **Expert opinion** field, enter the opinion in the opened field and press **ENTER**.

The expert opinion will be added to the selected incident and will appear in the incidents table in the **Incidents** section.

- If you need to add a note to an incident, enter your message in the **Note** field and click the **Add note** button.
You can provide a message up to 512 characters long.

The status, cause, expert opinion, and note will be added to the incident and will be available to other users when viewing this incident.

When two or more similar incidents are detected, Kaspersky MLAD automatically combines them into a group. The group name is also automatically assigned in the format Group #N (N is replaced by the sequence number of the group). You can edit the group name, change the status of an incident group, and edit the expert opinion containing recommendations for analyzing similar events, for example.

To add a status and expert opinion to a group of incidents:

1. In the [main menu](#), select the **Incidents** section and click **Groups**.
2. If necessary, change the incident group status by selecting one of the following statuses from the **Status** drop-down list: **Under review**, **Decision pending**, **Instructions issued**, **Problem closed**, **Cause unknown**, **Ignore**, or **False positive**.

When changing the status of a group of incidents, Kaspersky MLAD changes the status of the incidents that are part of this group. By default, a group of incidents is assigned the **Unknown** status.

If necessary, the system administrator can [create, edit, or delete statuses of incidents](#).

3. In the [incident groups table](#), double-click the row of the incident group.

The **Edit group** window opens.

4. To change the name of the incident group, enter a new name for the group in the **Group name** field.
5. In the **Expert opinion** field, enter the text of the expert opinion (for example, recommendations for analyzing similar incidents).
6. Click the **Save** button.

The status and expert opinion will be changed for the incident group and can now be viewed by other users in the **Groups** table in the **Incidents** section.

Exporting incidents to a file

Incidents registered for a specific period in Kaspersky MLAD can be exported to an XLSX file.

The functionality is available after [a license key is added](#).

To save incidents registered for a specific period to a file:

1. In the [main menu](#), select the **Incidents** section.
2. In the upper part of the opened page, select the start and end dates of the period.


3. Click the **Export** button.
4. Select a directory to save on your local drive, and save the file.

Incidents registered for the selected period in Kaspersky MLAD will be saved to an XLSX file on the local drive. The XLSX file can be opened in Microsoft® Excel®.

Managing ML models

This section provides instructions on working with ML models, ML model templates and markups.



The functionality is available after [a license key is added](#).

ML models, templates of ML models and markups are functional elements of the [monitored asset hierarchical structure](#). The hierarchical structure is displayed as an [asset tree](#) .

In Kaspersky MLAD, ML models can be [imported](#), [created manually](#), [copied](#), or [created based on a template](#). If you created the ML model manually, cloned a manually created model, or created the model from a template based on a manually created model, you can add [predictive elements](#), [elliptic envelope-based elements](#), and/or [diagnostic rule-based elements](#) to the new model.

After training the ML model elements and [checking the results of their training](#), you can [run historical or streaming inference](#) on the ML model. As a result of inference, ML model elements register incidents and also generate artifacts that can be viewed under [Monitoring](#) and [History](#).

You can [publish](#) the ML model if needed. You can run historical or streaming inference on a published ML model.

In the **Models** section, you can [create markups](#) for generating [learning indicators](#)  or [inference indicators](#) . If necessary, you can [edit](#), [clone](#), or [delete](#) markups.

About ML-models

An *ML model* is an algorithm based on machine learning methods tasked with analyzing the telemetry of the monitored asset and detecting [anomalies](#).

An ML model is created for a specific monitored asset while taking into account the specifications of the asset and the characteristics of telemetry data. The general structure of the algorithm (architecture) is formed during creation of the ML model. Then the ML model is trained based on historical telemetry data and is thereby adjusted to the behavior of a specific object.

An ML model consists of one or more elements, with each separately analyzing telemetry data to detect anomalies. Normally, the more complex the industrial processes of the monitored asset are, the more elements the ML model will contain. An ML model can include the following elements operating in parallel:

- [Predictive element](#)
- [Element based on a diagnostic rule](#)
- [Elliptic envelope-based element](#)

[Predictive elements](#) and [elements based on elliptic envelopes](#) need to be trained on a dataset. A predictive element learning process may consist of one or several epochs. An *epoch* is a cycle during which an element is trained on the entire training dataset. The number of training epochs is specified in the element training settings. Elements based on a diagnostic rule do not need to be trained, so they are considered to be pretrained.

The process of using an ML model to analyze telemetry data and detect anomalies is known as *inference*. In Kaspersky MLAD, ML model inference can be performed on historical data (*historical inference*) and on telemetry data received in real time (*streaming inference*). If [historical inference is started](#) for multiple ML models, Kaspersky MLAD runs the inference of these ML models in the order of their startup queue. The duration of historical inference is determined by the time interval of the data analyzed by the ML model. If streaming inference is started for multiple ML models, Kaspersky MLAD runs the inference of these ML models simultaneously. Historical inference and streaming inference run in parallel and independently of each other. During the inference process, the ML model registers [incidents](#) that can be viewed in the **Incidents** section.

In addition to incidents, an ML model inference process also generates *artifacts*. An artifact is a time series of numerical data. An ML model can generate the following artifacts:

- Artifacts associated with tags. An ML model element generates these artifacts for each of its output tags. These artifacts are generated only by the predictive elements of the ML model and represent a predicted tag value and prediction error.
- Artifacts of ML model elements. Each ML model element generates this type of artifact as its primary output. The mathematical nature of an artifact is determined by the analytical algorithms employed by the element. In this context, an artifact for an ML model of any type is uniformly interpreted as the degree to which the behavior of the monitored asset deviates from the expected (normal) behavior. Every artifact has a critical threshold. If this threshold is reached, an incident is recorded.

Any user can view generated artifacts under [Monitoring](#) and [History](#).

ML models can be created by Kaspersky specialists or by a certified integrator as part of the *Kaspersky MLAD Model-building and Deployment Service*. To use such ML models, you must [import them to Kaspersky MLAD](#). You can also [create ML models](#) independently and add the necessary elements to them using the model builder.

About predictive ML model elements

Predictive ML model elements predict the behavior of an object from data on its recent behavior. Predictive ML model elements include neural network elements and linear regression-based elements.

Kaspersky MLAD model builder supports the following architectures for ML model predictive elements:

- *Dense*. Neural network element of an ML model with a fully connected architecture. When creating an ML model element, you must specify the multipliers for calculating the number of neurons on inner layers and the activation functions on them.
- *TCN*. Neural network element of an ML model with a hierarchical time-based convolutional architecture. When creating an ML model element, you must specify the filter size and number, extensions on layers, activation functions on them and the number of layers in the residual block.
- *CNN*. Neural network element of an ML model with a convolutional architecture. When creating an ML model element, you must specify the number of neurons on the layers of encoder, the size and number of filters on layers, and the size of the maximum sampling window (MaxPooling).
- *RNN*. Neural network element of an ML model with a recurrent architecture. When creating an ML model element, you must specify the number of GRU neurons on layers and the number of time-distributed neurons on the layers of the decoder.

- *Transformer*. Neural network element of an ML model with a transformer architecture. When creating an element of the ML model, the number of attention heads and the number of transformer encoders are specified.
- *Linear regression*. Element of an ML model based on linear regression.

A predictive element of an ML model generates the following artifacts as a result of inference:

- Predicted tag values. These are displayed in the central part of the [Monitoring](#) and [History](#) sections on individual graphic areas of the selected preset.
- Individual prediction errors are the differences between the predicted and actual values for each tag. These are displayed in the central part of the [Monitoring](#) and [History](#) sections on individual graphic areas of the selected preset.
- The total prediction error (cumulative prediction error) is the total discrepancy between the predicted and actual values. Cumulative prediction error and the cumulative prediction error threshold are displayed in the graphic area in the central part of the [Monitoring](#) and [History](#) sections after the graphic areas of the selected preset and on the ML model element artifact graph located at the bottom of the sections.

If the cumulative prediction error exceeds the cumulative prediction error threshold, predictive element of the ML model considers this a deviation in the behavior of the monitored asset and registers an incident.

About elements of an ML model based on a diagnostic rule

Diagnostic rules describe previously known behavioral traits of the monitored asset that are considered anomalies. Diagnostic rules must be formalized and calculated based on available telemetry data for the object.

Examples of diagnostic rules:

- The level of tag A has changed abruptly (criterion for the behavior of the [Step change](#) tag).
- Over the past 12 hours, tag B has trended upward, tag C has trended downward, and tag D has not shown any clear dynamics.
- The value of tag X fell below 2800 after it previously rose higher than 2900.

About elliptic envelope-based ML model elements

Elliptic envelopes are used to detect abnormal states of a monitored asset.

Unlike a predictive element, an elliptic envelope does not attempt to determine how the behavior of the ML model's input tags affects the behavior of its output tags. An elliptic envelope uses the assumption that the set of tags included in the ML model describes the state of the monitored asset at any given moment, and the observable states have a normal distribution (also known as a Gaussian distribution) in the phase space.



During training, the elliptic envelope adjusts the parameters of this normal distribution while considering that the training sample may contain a certain percentage of anomalous states. During the training of an ML model, an elliptical region is formed in the phase space. States that fall within this region are classified as normal, while all other states are categorized as outliers (anomalies). The farther a state is from the boundaries of the ellipse, the more anomalous it is. The tag whose value as part of the anomalous state contributed the most to the deviation from the ellipse is considered the top tag.

An elliptic envelope is simpler to construct than a predictive element, learns more quickly, and requires fewer resources for inference. However, an elliptic envelope only demonstrates good performance when applied to stationary equipment operating modes that do not involve multiple operating ranges or abrupt changes in tag values.






About statuses and states of ML models and their elements

The statuses and states of ML models and their elements signify sequences of steps completed by the user under **Models**.

ML model elements can take the following statuses:




- **Not trained.** This status is assigned to an ML model element if training has not started or completed with an error. This status is also displayed for elements that were trained previously, but whose settings have been changed. The asset tree displays the  icon to the left of the untrained element names.
- **Trained.** This status is assigned to an ML model element if training has been successfully completed or if the element is not subject to training. The asset tree displays the  icon to the left of the trained element names.

An ML model can be assigned one of the following statuses:

- **Not activated.** The ML model is [imported](#) but is not [activated](#). The asset tree displays the  icon to the left of the ML model name.
- **Not trained.** The ML model activated or [created manually](#). The ML model contains untrained elements. The asset tree displays the  icon to the left of the ML model name.
- **Trained.** All the elements in the ML model have been trained or no training is required. [Inference can be run](#) on a trained ML model. The asset tree displays the  icon to the left of the ML model name.
- **Ready for publication.** The ML model is [ready for publishing](#) and cannot be modified. The asset tree displays the  icon to the left of the ML model name.
- **Published.** The ML model is [published](#). Inference can be run on a published ML model. The asset tree displays the  icon to the left of the ML model name.

The states of ML models and their elements are displayed to the right of the ML model name when viewing a specific ML model, and in the asset tree. The table below lists the states of ML models and their elements in Kaspersky MLAD.

Statuses of ML models and elements of ML models

Name of the state when viewing the ML model	Symbol of the state in the asset tree	ML model	ML model element	Description
Not used	—		—	This state is assigned to an ML model unless inferred or trained earlier, or if the model was viewed after inference or training finished. The asset tree does not display this status.
Training completed with error	TRN ERR			This state is assigned to ML model elements whose training finished with an error.

				An ML model is also assigned this state if containing at least one element whose training finished with an error. After viewing the training results , the ML model is assigned a state of Not used .
Training in progress	TRN	✓	✓	This status is assigned to an ML model element that is currently undergoing training. The model itself is assigned this status if it contains elements that are undergoing training and unless it contains any elements whose training completed with an error.
Queued for training	TRN Q	✓	✓	An ML model element is assigned this state if the model has started learning, but the training is running for a different element. An ML model is assigned this state if all of its elements are queued for training.
Training successfully completed	TRN DONE	✓	✓	This state is assigned to a successfully trained ML model element. An ML model is assigned this state if all of its elements have been successfully trained. After viewing the training results , the ML model is assigned a state of Not used .
Historical inference in progress	INFR HIST	✓	—	This state is assigned to an ML model running historical inference. You can view the inference results under History . Incidents logged during inference are displayed under Incidents .
Queued for inference	INFR Q	✓	—	This status is assigned to an ML model currently running historical inference, while a different model is currently running historical inference.
Historical inference completed	INFR DONE	✓	—	This state is assigned to an ML model that has finished historical inference. You can view the inference results under History . Incidents logged during inference are displayed under Incidents . An ML model is assigned a state of Not used after being viewed.
Streaming inference in progress	INFR STRM	✓	—	This state is assigned to the ML model running streaming inference. You can view the inference results under Monitoring . Incidents logged during inference are displayed under Incidents .

About ML model templates

ML model templates are created on the basis of ML models previously added to Kaspersky MLAD or created using the model builder functionality. ML model templates preserve the algorithm structure, set of elements, and the state of the ML model used to create the template. The training state of the created ML model will match the training state of the source ML model when the template was created.

Using templates, you can add ML models of the same type to Kaspersky MLAD. These models will analyze data received from equipment of the same type with a similar set of tags. When creating an ML model from a template, you can configure the use of other tags in the ML model by specifying tag IDs that differ from the ones in the source ML model.

About markups

Markup is the tool for selecting time intervals. Markups are used to generate learning indicators and [inference](#) of the ML model. Markups that form part of learning indicators define the data time intervals from which the ML model takes data for training. Markups that form part of inference indicators define the time intervals during which the ML model performs the inference.

A markup may utilize two types of criteria: conditions on the behavior of specific tags (time intervals are selected where these conditions are met) and a time filter (time intervals are selected independently of tag behavior).

Markup is a functional element of the [hierarchical structure](#). Markups can be [created manually](#) or imported into Kaspersky MLAD together with an ML model.

About conditions included in markups and diagnostic rules

The selection of data time intervals for learning or inference indicators in the markup, and the execution of a diagnostic rule in the ML model are governed by the conditions that are set when [creating a markup](#) and/or [ML model element based on a diagnostic rule](#). While creating a markup or an ML model element based on a diagnostic rule, you can specify the following condition types:

- Time filter.

The time filter defines a sequence of recurring calendar intervals, such as an interval that considers the number of business days in a week and work hours, or a set of intervals with precisely defined start and end times.

In the absence of defined tag behavior conditions, the filtered intervals will be a product of the markup or a diagnostic rule. The rule will be considered fulfilled at all UTG nodes within the selected intervals.

- Tag behavior conditions.

The tag behavior conditions are checked at the UTG nodes that fall within the time intervals selected by the filter. Without time filtering enabled, tag behavior conditions are evaluated at all UTG nodes.

Tag behavior criteria are described in condition blocks and linked by the logical operators **AND** and/or **OR**. The operator **AND** tracks the simultaneous fulfillment of all related criteria, and the operator **OR** tracks the fulfillment of at least one linked criterion. The negation operator **NOT** can be applied individually to the criteria in the condition block to track behavior opposite to that described in the criterion. Condition blocks themselves can also be linked with the logical operators **AND** and/or **OR**.

Evaluating any condition yields one of three possible outcomes:

- Positive (TRUE) if the condition is met. If the criteria are linked by the logical operator **OR** and the evaluation of the criteria resulted in TRUE and UNDEFINED, then the evaluation of the entire condition block will yield a positive result.
- Negative (FALSE) if the condition is not met.
- Undefined (UNDEFINED) if it is impossible to check if the condition was met (for example, when there is not enough data). Evaluation of the entire block of conditions produces an undefined result in the following cases:
 - If the criteria within the block of conditions are connected by the logical operator **OR** and the evaluation of individual criteria produced FALSE and UNDEFINED.
 - If the criteria within the block of conditions are connected by the logical operator **AND** and the evaluation of individual criteria produced UNDEFINED and/or FALSE.

You can link two condition blocks with the temporal operators **Wait** or **If ahead** if required. The condition block that precedes the temporal operator is called a *precondition*. The condition block that follows the temporal operator is called a *post-condition*. Unlike the logical operators **AND** and **OR**, which require simultaneous evaluation of conditions, the temporal operator connects blocks of conditions that are evaluated at different points in time.

A precondition is evaluated at one UTG node. A postcondition is evaluated at one or more consecutive UTG nodes. The interval between the pre-condition check node and the node where the post-condition is checked corresponds to the waiting interval. The post-condition check is controlled with the following settings:

- Minimum waiting interval is the interval between the pre-condition check node and the UTG node where the post-condition check will start.
- Maximum waiting interval is the interval between the pre-condition check node and the UTG node where the post-condition check will finish.
- A group operator that specifies whether the postcondition needs to hold at every postcondition check node or at just one.

A FALSE or UNDEFINED result of a precondition check causes the entire temporal operator to return the same value. If the pre-condition check evaluates to TRUE, then the post-condition check is performed at each UTG node between the minimum and maximum wait intervals. The result of applying the temporal operator is determined by the results of the post-condition checks and considers the value of the group operator.

If more than one condition check is performed with the temporal operator, then the output of the previous temporal operator is a precondition for each subsequent temporal operator.

The result of applying the temporal operator **If ahead** is generated at the precondition check node. The temporal operator **If ahead** can only serve as a training indicator component, as it requires future data, which is not yet available during the inference process.

The result of applying the temporal operator **Wait** is generated in the last UTG node of the post-condition check. Since all nodes involved in the operator are in the past at this point, the **Wait** operator can be used as part of both a training indicator and an inference indicator.

The overall result of evaluating all markup conditions or a diagnostic rule can be either TRUE or FALSE. If a UTG node produces an UNDEFINED result when evaluating all specified tag behavior conditions, the overall outcome of applying markup or diagnostic rule for that node is determined by the **Treat inconclusive result as positive** setting.

Scenario: working with ML models

This section describes the sequence of actions required to work with ML models.

The functionality is available after [a license key is added](#).

The scenario for working with ML models consists of the following steps:

1 Adding markups

If you need to select specific time intervals for the data that ML models must use for training or inference, [create markups](#).

2 Adding an ML model

You can add an ML model to Kaspersky MLAD in one of the following ways:

- [Import an ML model](#) created by Kaspersky specialists or by a certified integrator as part of the *Kaspersky MLAD Model-building and Deployment Service*. If the ML model uses markups, they will be incorporated into the same asset as the model itself. After an ML model is imported, it must be [activated](#).
- [Manually create an ML model](#). [Add predictive elements](#), [elliptic envelope-based elements](#), and/or [diagnostic rules-based elements](#) to the new ML model.
- [Create an ML model from a template](#). [Create a template based on the relevant ML model](#) in advance. If the original ML model used for the template was created manually, you can add predictive elements, elliptic envelope-based elements, and/or diagnostic rule-based elements to the new ML model.
- [Clone a previously added ML model](#). After cloning an ML model that was created manually or from a template based on a manually created ML model, you can add predictive elements, elliptic envelope-based elements, and/or diagnostic rule-based elements to the new ML model.

3 Training ML model elements

The ML model needs to be trained before you can run inference on it. To do this, all [predictive elements](#) and elliptic envelope-based elements within the ML model must be [pretrained](#). ML model elements based on diagnostic rules do not need to be trained, so they are considered to be pretrained.

An ML model imported to Kaspersky MLAD has been previously trained by Kaspersky Lab experts or a certified integrator. ML models that are created from a template of an imported ML model or by cloning an imported ML model are also considered to be already trained. If necessary, you can change their training settings and retrain the elements.

To generate a learning indicator, specify the created markup in the element training settings.

After training the elements, [examine the training results](#), adjust the training settings and retrain the elements, if necessary.

4 ML model inference

[Run a historical or streaming inference on the ML model](#). Examine the artifacts under [History](#) and [Monitoring](#), and [incidents](#) inferred by the ML model.

For better ML model performance, adjust the parameters of the model and/or markups. Re-train the elements of the ML model as needed. Run a repeat inference on the ML model. When restarting an inference on previously inferred data, previous inference results will be deleted.

5 Preparing an ML model for publication

If you need to save the parameters of an ML model and its elements, [prepare the ML model for publication](#) after completing training and checking the inference results.

6 Publishing an ML model

After preparing the ML model for publication, notify the officer responsible for [publishing the ML model](#) that the ML model is ready, or publish the ML model if you have the required permissions. If necessary, the system administrator can [create a role](#) that has the right to publish ML models and [assign this role to the relevant employee](#).

7 Inferencing a published ML model

[Start inference](#) of the ML model. During the inference process, published ML model analyzes telemetry data and log [incidents](#). Recorded incidents, unlike those inferred by unpublished ML models, necessitate actions and reporting in production.

Search and filter objects in the Models section

In the **Models** section of the asset tree, you can search for and filter the following objects:

- ML models
- ML model templates
- ML model elements
- Markups
- Assets
- Tags

The search is done by object name.

To find objects in the asset tree,

under **Models**, enter your search query into the **Search** field. The search is performed as you type characters in the search field.

Matching objects will be displayed in the tree, along with the asset tree sections where they were found.

To reset the search query, click  in the search bar.

You can filter objects within the asset tree. You can select [object statuses and states](#) for ML models and their elements.

Filtering is applied to objects that are found according to the search query. If no search query is defined, filtering is applied to all objects in the asset tree.

To filter objects within the asset tree:

1. Under **Models**, click  above the asset tree.

The filter options will be displayed on the right.

2. From the **Section type** drop-down list, select one or more asset tree section types.

You can select the following section types: **Models**, **Model templates**, **Model elements**, **Markups**, **Assets**, and **Tags**.


3. If you have selected **Models**, do the following as needed:

- From the **Model status** drop-down list, select ML model statuses.
- From the **Model state** drop-down list, select the ML model states.

4. If you have selected the **Model elements** section type, do the following as needed:


- From the **Model element type** drop-down list, select one or more ML model element types.
You can choose the following types of ML model elements: **Predictive element**, **Rule**, and **Elliptic envelope**.
- From the **Model element status** drop-down list, select ML model element statuses.
- In the **Model state** drop-down list, select the states of ML model elements.

5. To reset the filter settings, do one of the following:

- To reset a specific filter setting, click  next to the setting.
- To reset all filter settings, click **Clear filters** in the upper right corner of the window.

6. To hide the filter settings, click  next to the **Search** field.

Object filtering is performed as you select the filter criteria. If any objects match your filter criteria, the tree view will display those items, along with the corresponding categories where they were found.



If any filters are applied to the asset tree,  will appear above.

Working with markups

This section provides information on working with markups.

The functionality is available after [a license key is added](#).

In the **Models** section, you can [create](#), [modify](#), and [delete markups](#). If required, you can [view the graph to see the data time intervals](#) that the ML model will use for training and/or inference.

Markups are used as training or inference indicators to point to data time intervals that the ML model can use for training or inference. When [creating](#) or [changing the parameters of an ML model](#), you can generate an [inference indicator](#)  by selecting one or several previously created markups. When configuring the training parameters for ML model elements, you can generate a [learning indicator](#)  by selecting one or more previously created markups.

Creating markup

You can use markup to generate learning indicators or inference of the ML model.

The functionality is available after [a license key is added](#).

To create markup:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, next to the name of the asset for which you want to create a markup, open the vertical menu **...** and select **Create markup**.
A list of options appears on the right.
3. Specify the name of the markup in the **Name** field.
4. Enter a description for the markup in the **Description** field.
5. In the **Grid step (sec)** field, specify a UTG period for markup in seconds expressed as a decimal.
6. In the **Markup color** field, select a color that will be used to highlight data intervals selected by the markup.

7. If necessary, turn on the **Treat inconclusive result as positive** toggle switch.

If Kaspersky MLAD cannot unequivocally evaluate the fulfillment of criteria specified in the **Time filter** and **Tag conditions** settings blocks, for example, due to the absence of observations for tags, the application will consider specified criteria to be fulfilled when this option is enabled.

8. In the **Time filter** settings block, do one the following:

- To add an interval, click **Add interval** and select one of the following time interval types from the **Interval type** drop-down list:
 - **Fixed.** If you select this type of interval, specify the days of the week and the time interval during which the input data must be validated according to the specified criteria.
 - **Recurrent.** If you select this type of interval, specify the years, dates, days of the week, and daily time interval for periodically validating input data according to the specified criteria.
- To delete an interval, click **×** to the right of the interval.

You can add one or more time intervals.

9. To add tag behavior criteria, do the following:

a. In the **Tag conditions** settings block, click the **Condition** button.

b. In the **Tag** drop-down list, select the tag for which to add a tag behavior criterion.

If you need to check the behavior directly opposite of the selected behavior criterion from the condition block, click the **NOT** button on the left of the selected tag. The **NOT** caption in the button will be highlighted in bold.

For example, click the **NOT** button if you need to add a condition that contains no steps with the specified settings.

c. In the **Behavior** drop-down list, select one of the following tag behaviors that must be tracked:

- **Over:** the tag value exceeds the specified threshold.
- **Below:** the tag value falls below the specified threshold.
- **Rising:** the trendline of tag values is increasing.
- **Falling:** the trendline of tag values is decreasing.
- **Level:** there are no pronounced changes in the trendline of tag values.
- **Step change:** the trendline of the selected tag is displaying abrupt upward or downward shifts.
- **Flat:** the selected tag is transmitting the same value.
- **Spread:** abrupt changes in the spread of values are being observed around the trendline of the selected tag.

d. In the **Window** field, specify an interval for analyzing the behavior of tags in the UTG steps.

e. Depending on the value selected for **Behavior**, do one of the following:

- If you selected **Over** or **Below**, specify a tag threshold value in the **Threshold** field and specify the minimum number of times the threshold value can be breached within a window in the **Minimum**

violations field.

- If you selected **Rising**, **Falling**, or **Level**, use the **Threshold slope** field to specify the trend slope percentage value that must be exceeded for the trend to be considered as growing or falling, and specify the time interval between adjacent trend estimates in the **Evaluation period** field.

By default, the **Threshold slope** setting is not defined. If the setting is not defined, Kaspersky MLAD will determine the trend direction automatically.

By default, the **Evaluation period** setting has a value of **1**. With this value, the trend is estimated at each UTG node.

- If you selected **Spread**, use the **Minimum change** field to specify the minimum value by which the tag value spread around the trendline can change, and select one of the following spread change directions in the **Direction** drop-down list: **Any**, **Flare**, or **Shrink**.

By default, the **Minimum change** setting is not defined. If the setting is not defined, Kaspersky MLAD will determine it automatically.

The tag behavior criterion is met when the tag spread around the trendline increases and/or decreases.

- If you selected **Step change**, use the **Minimum change** field to specify the minimum shift value for the tag trendline, and select one of the following tag value change directions from the **Direction** drop-down list: **Any**, **Up** or **Down**.

By default, the **Minimum change** setting is not defined. If the setting is not defined, Kaspersky MLAD will determine it automatically.

- If you selected **Flat**, use the **Value** field to specify the value that the tag should transmit, and specify the maximum tag value spread in the **Spread** field.

The **Spread** parameter is set to zero by default. With this value, any repeating tag value triggers the criterion.

f. To add a tag behavior criterion to a condition block, click the plus sign at the bottom of the condition block and repeat steps 9b through 9e.

g. If the block contains more than one tag behavior criterion, select one of the following logical operators between the criterion rows by clicking logical operator button:

- **AND** if you require all of the block criteria to be fulfilled at the same time.
- **OR** if at least one of the block criteria must be fulfilled.

h. To delete a tag behavior criterion from a condition block, click **x** in the row that contains the criterion.

10. If you need to check whether the fulfillment of a pre-condition triggered the fulfillment of a post-condition, do the following:

a. Add one of the following [temporal operators](#):

- **Wait** if you need to generate the result of the criteria check in the last node of the maximum waiting interval.
- **If ahead** if you need to generate the result of the criteria check at the time of a pre-condition check.

The **Wait** and **If ahead** buttons are available after adding at least one condition.

Markup with an **If ahead** temporal operator can be used in learning indicators only.

b. In the **Recess (steps)** field, specify the following time intervals:

- **from** is the interval between the pre-condition check node and the UTG node where the post-condition check will start (minimum waiting interval).
- **to** is the interval between the pre-condition check node and the UTG node where the post-condition check will finish (maximum waiting interval).

The post-condition is checked in the UTG nodes between the minimum and maximum waiting intervals.

c. In the **Check** drop-down list, select one of the following group operators:

- If you require fulfillment of tag behavior criteria from the post-conditions in all UTG nodes between the minimum and maximum waiting intervals, select the **All steps** group operator.
- To require fulfillment of tag behavior criteria from the post-conditions in at least one UTG node between the minimum and maximum waiting intervals, select the **Any step** group operator.

If the **Wait** temporal operator is added, the criteria check result is determined in the last node of the maximum waiting interval. If more than one condition check is performed using the **Wait** temporal operator, the result of the previous temporal condition check is the precondition for each subsequent check of the **Wait** temporal condition.

If the **If ahead** temporal operator is added, the criteria check result is generated at the time of the precondition check.

11. Select one of the following logical operators between markup blocks by clicking the logical operator button:

- **AND** if you require the criteria of both condition blocks to be fulfilled.
- **OR** if the criterion of at least one of the condition blocks must be fulfilled.

12. In the upper-right corner of the window, click the **Save** button.

The new markup will be displayed in the **Markups** group of the asset tree. The **Markups** group is created automatically and displayed as part of the selected section of the asset tree.


Viewing the markup chart

After [creating markup](#), you can view data time intervals selected by the markup on the graph.


The functionality is available after [a license key is added](#).

To view the markup chart:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the markup whose chart you want to view.
A list of options appears on the right.
3. Click the **On graph** button.
A pane with the markup chart appears on the right.
4. Select the relevant preset from the **Preset** drop-down list.

5. If necessary, in the **Markups** field, select the markups for displaying data intervals.
6. If you need to select a date and time for displaying the data, do one of the following:
 - In the **Graph center** field, select the date and time for which you want to display data in the chart.
The vertical black dotted line will indicate the selected date and time (in the center of the chart).
 - Click  to the left of the time axis, and select the point on the time axis.
The selected point will become the new center of the graph. The vertical black dashed line will indicate the new date and time.
7. If you need to select a time interval for displaying data on the chart, do one of the following:
 - If you need to display data for a fixed time interval, select the relevant time interval from the **Scale** drop-down list. The following time intervals are available by default:
 - 1, 5, 10, 15, and 30 minutes
 - 1, 3, 6, and 12 hours
 - 1, 2, 15, and 30 days
 - 3 and 6 months
 - 1, 2, and 3 years

If necessary, the system administrator can [create, edit, or delete time intervals](#).

 - To display data for a custom time interval, click the  button icon to the left of the time axis, select the required interval on the time axis, and click the **Apply** button. If you need to change the scale again, repeat this step.

The chart will show the data intervals in the colors specified for the selected markups.

Copying a markup

You can create a markup by copying a previously created one. Copying will create a markup whose settings match those of the original at the time of copying.

The functionality is available after [a license key is added](#).

To copy a markup:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, next to the name of markup that you want to copy, open the vertical menu **...** and select **Copy markup**.
The **Copy markup** pane appears on the right.
3. Specify the name of the markup in the **Name** field.
By default, the markup is assigned a name in the following format: < name of the original markup >_Cloned_< date and time of cloning >.

4. In the **Asset** drop-down list, select the asset to which you want to assign the markup.

5. Click the **Save** button.

The new markup will be displayed in the **Markups** group of the asset tree. The **Markups** group is created automatically and displayed as part of the selected section of the asset tree.

Modifying the markup

You can edit the markup settings.

Markup settings cannot be edited for imported ML models and ML models that were created by cloning imported ML models or based on a template of imported ML models.

The functionality is available after [a license key is added](#).

To edit markup:


1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the markup that you want to edit.
A list of options appears on the right.
3. Click the **Edit** button.
4. Adjust the markup settings, if needed. For a description of the settings, see the [instructions on creating markup](#).
5. In the upper-right corner of the window, click the **Save** button.

Removing markup

You can delete markup if it is not used for training or inference of any ML model.

The functionality is available after [a license key is added](#).

To delete markup:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the markup that you want to delete.
A list of options appears on the right.
3. In the upper-right corner of the window, click .
4. In the window that opens, confirm the deletion of the markup.

Working with imported ML models

This section provides information about working with imported ML models and their elements.

The functionality is available after [a license key is added](#).

ML models can be provided by Kaspersky specialists or certified integrators within the *Kaspersky MLAD Model-building and Deployment Service*. Such ML model must be [imported](#) to Kaspersky MLAD and [activated](#). You cannot create new elements for an imported ML model, or delete existing elements.

Upon importing into Kaspersky MLAD the ML model is already trained. You can [train the predictive elements](#) and [elliptic envelope-based elements](#) as part of the imported ML model before [running inference](#) and/or [publishing it](#).

ML model importing

If the ML model was created by Kaspersky specialists or a certified integrator, you can import this ML model into Kaspersky MLAD.

Kaspersky MLAD may slow down its operation when importing an ML model whose size exceeds 1 GB.

System administrators and users who have the **Upload models** permission from the [Manage ML models](#) group of rights can import ML models. The functionality is available after [a license key is added](#).

To import an ML model:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, next to the name of the asset for which the ML model is to be imported, open the vertical menu **...** and select **Import model**.
3. In the opened window, select the ML model file.

An ML model file is provided as a TAR archive with a maximum size of 1.5 GB.

The ML model will be imported to Kaspersky MLAD. The new ML model displays in the **Models** group of the asset tree. The **Models** group is created automatically and displayed as part of the selected section of the asset tree. If the imported ML model contains predictive elements, elliptic envelope-based elements, and/or diagnostic rule-based elements, the **Models** group will display the **Predictive elements**, **Elliptic envelopes**, and/or **Rules** subgroups, respectively.

After being imported, the ML model is assigned the [Not activated status](#). The [ML model must be activated](#). If you import an ML model that was previously activated and then deleted, you do not need to reactivate the ML model.

Activating an imported ML model

After an ML model prepared by Kaspersky specialists or a certified integrator has been imported into Kaspersky MLAD, it must be activated.

If the ML model activation code is lost, send a request to Kaspersky to receive a new code.

System administrators and users who have the **Activate models** permission from the [Manage ML models](#) group of rights can activate imported ML models. The functionality is available after [a license key is added](#).

To activate an imported ML model:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the imported ML model.
The details area appears on the right.
3. In the **Model activation code** field, enter the code received from Kaspersky personnel, and click the **Activate** button in the upper right part of the window.

ML model is activated. It will be assigned the [Trained status](#). If necessary, you can train the ML model again. For example, you can train it again on new data.

You can to [start ML model inference](#) to begin the analysis of telemetry data received from the monitored asset.

Changing the parameters of an element of an imported ML model

You can change some parameters of an element of an imported ML model.

Parameters cannot be changed if the ML model is assigned the **Ready for publication** or **Published** status.

System administrators and users who have the **Edit untrained models** permission from the [Manage ML models](#) group of rights can edit the settings of elements of imported ML models. The functionality is available after [a license key is added](#).

To change the parameters of an imported ML model element:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model element that you want to change.
A list of options appears on the right.
3. In the upper-right corner of the window, click the **Edit** button.
4. Adjust the following element settings, if needed:

- Name and description of the ML model element
- Reminder period

This parameter is unavailable for editing if the ML model is in the **Historical inference in progress** or **Streaming inference in progress** state.

Modifying this setting changes anomaly detection sensitivity.

- Period of recurring alert suppression

This parameter is unavailable for editing if the ML model is in the **Historical inference in progress** or **Streaming inference in progress** state.

Modifying this setting changes anomaly detection sensitivity.

- Anomaly observation period

This parameter is unavailable for editing if the ML model is in the **Historical inference in progress** or **Streaming inference in progress** state.

Modifying this setting changes anomaly detection sensitivity.

- Anomaly duration share in interval

This parameter is unavailable for editing if the ML model is in the **Historical inference in progress** or **Streaming inference in progress** state.

Modifying this setting changes anomaly detection sensitivity.

- Color of incident dot indicators
- Incident status and cause
- Detection threshold

This parameter is unavailable for editing if the ML model is in the **Historical inference in progress** or **Streaming inference in progress** state.

The detection threshold value was set after training an element of the imported ML model. Modifying this setting changes anomaly detection sensitivity.

- Expert opinion

5. In the upper-right corner of the window, click the **Save** button.

Working with manually created ML models

This section provides information about working with manually created ML models and their elements.

The functionality is available after [a license key is added](#).

When [creating an ML model manually](#), you can [add predictive ML model elements](#), [elliptic envelope-based elements](#), and/or [diagnostic rule-based elements](#), and edit or [delete these](#).

The ML model needs to be trained before you can run [inference](#) on it. To do this, all predictive elements and elliptic envelope-based elements within the ML model must be pretrained. If necessary, you can [view the training results of the elements](#). Elements based on diagnostic rules do not need to be trained, so they are considered to be pretrained.

Creating an ML model

System administrators and users who have the **Create models** permission from the [Manage ML models](#) group of rights can create ML models. The functionality is available after [a license key is added](#).

To create an ML model:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, next to the name of the asset for which you want to create an ML model, open the vertical menu **...** and select **Create model**.
A list of options appears on the right.
3. In the **Name** field, specify the ML model name.
The ML model name must not be longer than 100 characters.
4. In the **Description** field, specify the ML model description.
5. If you need to apply markups when selecting data for ML model inference, select the required markups under **Inference indicator**.
6. To view the data that will be selected by the markups, click **On graph**.
Markups are displayed in the colors selected when they were [created](#).
7. In the upper-right corner of the window, click the **Save** button.

The new ML model displays in the **Models** group of the asset tree. The **Models** group is created automatically and displayed as part of the selected section of the asset tree.

Adding a predictive element to an ML model

System administrators and users who have the **Create models** permission from the [Manage ML models](#) group of rights can add ML model elements. The functionality is available after [a license key is added](#).

To add a predictive element to an ML model:

1. In the [main menu](#), select the **Models** section.
2. To add a predictive element, do the following:
 - a. In the asset tree, next to the name of the ML model to which you want to add a predictive element, open the vertical menu **...** and select **Create element**.

b. In the window that opens, select the element type **Predictive element**.

c. Click the **Create** button.

A list of options appears on the right.

3. In the **Name** field, specify the name of the ML model element.

4. Enter a description for the ML model element in the **Description** field.

5. In the **General element settings** block, do the following:

a. In the **Reminder period (sec)** field, specify the period in seconds, upon reaching which the ML model will generate a repeated incident if anomalous behavior is retained in each UTG node.

The default value of this setting is 0, which corresponds to no reminders.

b. In the **Period of recurring alert suppression (sec)** field, specify the period in seconds during which the ML model does not log repeated incidents for the same element.

The default value of this setting is 0 (repeat incidents not suppressed).

c. In the **Anomaly observation interval (sec)** field, enter the period (in seconds) during which the anomalous behavior of the tag is monitored to make a decision regarding incident registration.

d. In **Anomaly duration share in interval**, enter as a decimal fraction the proportion of the period in **Anomaly observation interval (sec)** that must elapse for the ML model element to register an incident.

You can specify a value in the range of 0 to 1.

e. In the **Color of incident dot indicators** field, select the color of the indicator points of the incidents logged by the ML model element on the graphs in the **Monitoring** and **History** sections. This color will also be used to display the graph of the artifact generated by this element.

f. If necessary, in the **Incident status** drop-down list, select a status to be automatically [assigned to incidents](#) logged by the ML model element.

g. If necessary, in the **Incident cause** drop-down list, select the cause to be automatically [set for incidents](#) logged by the ML model element if this cause is known in advance.

h. In the **Detection threshold** field, specify a prediction error threshold value upon reaching which an incident is logged.

The value of this parameter will be automatically adjusted after training the ML model element. If necessary, you can [change the value of this parameter](#).

i. If required, in the **Expert opinion** field, specify the expert opinion that will be automatically generated for incidents registered by the ML model element if the contents of this opinion are known in advance.

6. Select one of the following [ML model predictive element architectures](#): **Dense**, **RNN**, **CNN**, **TCN**, **Transformer**, or **LR**.

7. If necessary, turn on the **Advanced neural network settings** toggle switch.

The toggle switch is only available for elements with a **Dense**, **RNN**, **CNN**, **TCN**, or **Transformer** architecture.

8. In the **Main settings** block, do the following:

a. In the **Grid step (sec)** field, specify the element's UTG period (in seconds) expressed as an integer or decimal.

- b. In the **Input tags** drop-down list, select one or more tags that serve as the source data for predicting the values of the output tags.
 - c. In the **Output tags** drop-down list, select one or several tags whose behavior is predicted by the model element.
 - d. In the **Smoothing factor** field, specify the cumulative prediction error smoothing factor in decimal format.
The higher the coefficient, the less smoothing is applied to the data.
 - e. In the **Prediction error power exponent** field, specify the power to which the prediction error value is raised at each UTG node before calculating the cumulative error.
9. In the **Window settings** block, do the following:
- a. In the **Input window (steps)** field, specify the size of the input value window, from which the ML model element predicts the output values.
The window size is indicated in the number of UTG steps.
 - b. In the **Output window offset** field, specify the number of UTG steps by which the beginning of the output window will be shifted relative to the beginning of the input window.
 - c. In the **Output window (steps)** field, specify an output tag prediction length calculated from the input tags on the input window.
10. If extended setup mode is enabled and you are adding an element with a Dense architecture, do the following:
- a. In the **Multipliers for calculating number of neurons per layer** field, provide the multipliers, separated by a comma without spaces, by which to multiply the number of input tags to calculate the number of neurons in the ML model element layers.
The default value of this parameter is 8,4,8.
 - b. In the **Activation function per layer** field, specify one of the following activation functions on each layer of an ML model element separated by a comma without spaces:
 - **relu**: A non-linear activation function that converts an input value to a value between 0 and positive infinity.
 - **selu**: A monotonically increasing function that enables normalization based on the central limit theorem.
 - **linear**: A linear function that is a straight line proportional to the input data.
 - **sigmoid**: A non-linear function that converts input values to values between 0 and 1.
 - **tanh**: A hyperbolic tangent function that converts input values to values between -1 and 1.
 - **softmax**: A function that converts a vector of values to a probability distribution that adds up to 1.

The default value of this setting is **relu,relu,relu**.
 - c. In the **Regularization** field, specify the regularization coefficient in decimal format to prevent overfitting of the ML model element.
The default value of this parameter is 0.
11. If extended setup mode is enabled and you are adding an element with an RNN architecture, do the following:

- a. In the **GRU neurons per layer** field, specify the number of GRU neurons on layers separated by a comma without spaces.

The default value of this parameter is 40,40.

- b. In the **Number of neurons in TimeDistributed layer** field, specify the number of neurons distributed in time on the layers of the decoder separated by a comma without spaces.

The default value of this parameter is 40,20.

- c. If you need to restore data received as input to the network, turn on **Use autoencoder** toggle switch.

- d. In the **Regularization** field, specify the regularization coefficient in decimal format to prevent overfitting of the ML model element.

The default value of this parameter is 0.

12. If extended setup mode is enabled and you are adding an element with an CNN architecture, do the following:

- a. In the **Filter size per layer** field, specify the size of the filters for each layer of the element separated by a comma without spaces.

The default value of this parameter is 2,2,2.

- b. In the **Number of filters per layer** field, specify the number of filters for each layer of the ML model element separated by a comma without spaces.

The default value of this parameter is 50,50,50.

- c. In the **Regularization** field, specify the regularization coefficient in decimal format to prevent overfitting of the ML model element.

The default value of this parameter is 0.

- d. In the **MaxPooling window size per layer** field, specify the maximum sampling window size on each layer separated by a comma without spaces.

The default value of this parameter is 2,2,2.

- e. In the **Number of neurons in decoder** field, specify the number of neurons on the layers of the decoder.

- f. If you need to restore data received as input to the network, turn on **Use autoencoder** toggle switch.

13. If extended setup mode is enabled and you are adding an element with an TCN architecture, do the following:

- a. In the **Regularization** field, specify the regularization coefficient in decimal format to prevent overfitting of the ML model element.

The default value of this parameter is 0.

- b. In the **Size of filters** field, specify the size of the filters for the ML model element.

The default value of this parameter is 3.

- c. In the **Number of layers in residual block** field, specify the number of residual block layers.

The default value of this parameter is 1.

- d. In the **Number of filters per layer** field, specify the number of filters for each ML model element layer.

The default value of this parameter is 64.

- e. In the **Dilation per layer** field, specify the exponential expansion values of the output data on the layers as a comma-separated list.

The default value of this parameter is 1, 2, 4, 8, 16.

f. In the **Decoder layer type** field, select one of the following types of layer to precede the output layer:

- **TimeDistributedDense** (default): A fully connected architecture layer.
- **GRU**: A layer with a recurrent architecture.

g. In the **Activation function** drop-down list, select one of the following activation functions:

- **linear**: A linear activation function whose result is proportional to the input value.
- **relu**: A non-linear activation function that converts an input value to a value between zero and positive infinity. If the input value is less than or equal to zero, the function returns a value of zero; otherwise, the function returns the input value.

The default value of this parameter is **linear**.

14. If extended setup mode is enabled and you are adding an element with a Transformer architecture, do the following:

a. In the **Encoder regularization** field, specify the regularization coefficient in the encoder in decimal format.

The default value of this parameter is 0.01.

b. In the **Number of attention heads** field, specify the number of attention heads.

The default value of this parameter is 1.

c. In the **Number of encoders** field, specify the number of encoders.

The default value of this parameter is 1.

d. In the **Multipliers for calculating number of neurons per layer** field, provide the factors, separated by a comma without spaces, by which to multiply the number of input tags to calculate the number of neurons in the decoding layers.

The default value of this parameter is 10, 5, 10.

15. In the upper-right corner of the window, click the **Save** button.

When the first item in the ML model is created, a **Predictive elements** group will be automatically created in the asset tree. The newly created element appears in this group.

The ML model element will be assigned the **Not trained** status, and the ML model to which the added element belongs will be assigned the **Not trained status**. To [run inference on the ML model](#), all of its [predictive elements](#) and [elliptic envelope-based elements must be trained](#).

Modifying an ML model predictive element

You can edit the settings of an ML model predictive element.

Parameters cannot be changed if the ML model is assigned the **Ready for publication** or **Published** status.

System administrators and users who have the **Edit untrained models** permission from the [Manage ML models](#) group of rights can edit elements of ML models. The functionality is available after [a license key is added](#).

To edit an ML model predictive element:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the predictive element that you want to edit.
A list of options appears on the right.
3. In the upper-right corner of the window, click the **Edit** button.
4. Adjust the settings of the predictive ML model element, if needed. For a description of the settings, see the [instructions on adding a predictive ML model element](#).

Editing the **Reminder period (sec)**, **Period of recurring alert suppression (sec)**, **Anomaly observation interval (sec)**, **Anomaly duration share in interval**, **Detection threshold**, and/or **Smoothing factor** settings changes anomaly detection sensitivity. These parameters are unavailable for editing if the ML model is in the **Historical inference in progress** or **Streaming inference in progress** state.

5. In the upper-right corner of the window, click the **Save** button.
6. If you have edited the neural network element architecture settings, and the options in **Main settings** and/or **Window settings**, confirm that you want to save the changes.
After changes are made to these parameters, the ML model element must be [retrained](#).

The element will be assigned the **Not trained** status.

Adding an ML model element based on a diagnostic rule

System administrators and users who have the **Create models** permission from the [Manage ML models](#) group of rights can add ML model elements. The functionality is available after [a license key is added](#).

To add an ML model element based on a diagnostic rule:

1. In the [main menu](#), select the **Models** section.
2. To add a diagnostic rule, do the following:
 - a. In the asset tree, next to the name of the ML model to which you want to add a diagnostic rule, open the vertical menu **...** and select **Create element**.
 - b. In the window that opens, select the **Rule** element type.
 - c. Click the **Create** button.
A list of options appears on the right.
3. In the **Name** field, specify a name for the diagnostic rule.

4. In the **Description** field, specify the diagnostic rule description.

5. In the **General element settings** block, do the following:

- a. In the **Reminder period (sec)** field, specify the period in seconds, upon reaching which the ML model will generate a repeated incident if anomalous behavior is retained in each UTG node.

The default value of this setting is 0, which corresponds to no reminders.

- b. In the **Period of recurring alert suppression (sec)** field, specify the period in seconds during which the ML model does not log repeated incidents for the same element.

The default value of this setting is 0 (repeat incidents not suppressed).

- c. In the **Anomaly observation interval (sec)** field, enter the period (in seconds) during which the anomalous behavior of the tag is monitored to make a decision regarding incident registration.

- d. In **Anomaly duration share in interval**, enter as a decimal fraction the proportion of the period in **Anomaly observation interval (sec)** that must elapse for the ML model element to register an incident.

You can specify a value in the range of 0 to 1.

- e. In the **Color of incident dot indicators** field, select the color of the indicator points of the incidents logged by the ML model element on the graphs in the **Monitoring** and **History** sections. This color will also be used to display the graph of the artifact generated by this element.

- f. If necessary, in the **Incident status** drop-down list, select a status to be automatically [assigned to incidents](#) logged by the ML model element.

- g. If necessary, in the **Incident cause** drop-down list, select the cause to be automatically [set for incidents](#) logged by the ML model element if this cause is known in advance.

- h. If required, in the **Expert opinion** field, specify the expert opinion that will be automatically generated for incidents registered by the ML model element if the contents of this opinion are known in advance.

6. In the **Rule settings** block, do the following:

- a. In the **Grid step (sec)** field, specify the element's UTG period (in seconds) expressed as an integer or in decimal format.

- b. If necessary, turn on the **Treat inconclusive result as positive** toggle switch.

If Kaspersky MLAD cannot unequivocally evaluate the fulfillment of criteria specified in the **Time filter** and **Tag conditions** settings blocks, for example, due to the absence of observations for tags, the application will consider a rule to be triggered when this option is enabled.

7. In the **Time filter** settings block, do the following:

- a. Click the **Add interval** button.

- b. In the **Interval type** drop-down list, select one of the following time interval types:

- **Fixed.** If you select this type of interval, specify the days of the week and the time interval during which the input data must be validated according to the specified criteria.
- **Recurrent.** If you select this type of interval, specify the years, dates, days of the week, and daily time interval for periodically validating input data according to the specified criteria.

- c. If you want to add one more interval, click the **Add interval** button and complete step 7b.

d. To delete an interval, click **x** to the right of the interval.

You can add one or more time intervals. If no time interval is specified, the diagnostic rule is applied in each UTG node.

8. To add tag behavior criteria, do the following:

a. In the **Tag conditions** settings block, click the **Condition** button.

b. In the **Tag** drop-down list, select the tag for which to add a tag behavior criterion.

If you need to check the behavior directly opposite of the selected behavior criterion from the condition block, click the **NOT** button on the left of the selected tag. The **NOT** caption in the button will be highlighted in bold.

For example, click the **NOT** button if you need to add a condition that contains no steps with the specified settings.

c. In the **Behavior** drop-down list, select one of the following tag behaviors that must be tracked:

- **Over**: the tag value exceeds the specified threshold.
- **Below**: the tag value falls below the specified threshold.
- **Rising**: the trendline of tag values is increasing.
- **Falling**: the trendline of tag values is decreasing.
- **Level**: there are no pronounced changes in the trendline of tag values.
- **Step change**: the trendline of the selected tag is displaying abrupt upward or downward shifts.
- **Flat**: the selected tag is transmitting the same value.
- **Spread**: abrupt changes in the spread of values are being observed around the trendline of the selected tag.

d. In the **Window** field, specify the number of UTG steps.

e. Depending on the value selected for **Behavior**, do one of the following:

- If you selected **Over** or **Below**, specify a tag threshold value in the **Threshold** field and specify the minimum number of times the threshold value can be breached within a window in the **Minimum violations** field.
- If you selected **Rising**, **Falling**, or **Level**, use the **Threshold slope** field to specify the trend slope percentage value that must be exceeded for the trend to be considered as growing or falling, and specify the time interval between adjacent trend estimates in the **Evaluation period** field.

By default, the **Threshold slope** setting is not defined. If the setting is not defined, Kaspersky MLAD will determine the trend direction automatically.

By default, the **Evaluation period** setting has a value of **1**. With this value, the trend is estimated at each UTG node.
- If you selected **Spread**, use the **Minimum change** field to specify the minimum value by which the tag value spread around the trendline can change, and select one of the following spread change directions in the **Direction** drop-down list: **Any**, **Flare**, or **Shrink**.

By default, the **Minimum change** setting is not defined. If the setting is not defined, Kaspersky MLAD will determine it automatically.

The tag behavior criterion is met when the tag spread around the trendline increases and/or decreases.

- If you selected **Step change**, use the **Minimum change** field to specify the minimum shift value for the tag trendline, and select one of the following tag value change directions from the **Direction** drop-down list: **Any**, **Up** or **Down**.

By default, the **Minimum change** setting is not defined. If the setting is not defined, Kaspersky MLAD will determine it automatically.

- If you selected **Flat**, use the **Value** field to specify the value that the tag should transmit, and specify the maximum tag value spread in the **Spread** field.

The **Spread** parameter is set to zero by default. With this value, any repeating tag value triggers the criterion.

f. To add a tag behavior criterion to a condition block, click the plus sign at the bottom of the condition block and repeat steps 8b through 8e.

g. If the block contains more than one tag behavior criterion, select one of the following logical operators between the criterion rows by clicking logical operator button:

- **AND** if you require all of the block criteria to be fulfilled at the same time.
- **OR** if at least one of the block criteria must be fulfilled.

9. If you need to check whether the fulfillment of a pre-condition caused the fulfillment of a post-condition in a future UTG node, add a [temporal operator](#):

a. In the **Tag conditions** settings block, click the **Wait** button.

The **Wait** button is available after at least one condition has been added.

b. In the **Recess (steps)** field, specify the following time intervals:

- **from** is the interval between the pre-condition check node and the UTG node where the post-condition check will start (minimum waiting interval).
- **to** is the interval between the pre-condition check node and the UTG node where the post-condition check will finish (maximum waiting interval).

The post-condition is checked in the UTG nodes between the minimum and maximum waiting intervals.

c. In the **Check** drop-down list, select one of the following group operators:

- If you require fulfillment of tag behavior criteria from the post-conditions in all UTG nodes between the minimum and maximum waiting intervals, select the **All steps** group operator.
- To require fulfillment of tag behavior criteria from the post-conditions in at least one UTG node between the minimum and maximum waiting intervals, select the **Any step** group operator.

The criteria check result is determined in the last node of the maximum waiting interval.

If more than one condition check is performed using the temporal operator, then the result of the check of the previous temporal condition is a precondition for each subsequent check of the temporal condition.

10. Select one of the following logical operators between rule blocks by clicking the logical operator button:

- **AND** if you require the criteria of both condition blocks to be fulfilled.
- **OR** if the criterion of at least one of the condition blocks must be fulfilled.

11. In the upper-right corner of the window, click the **Save** button.

When the first ML model element is created, a **Rules** group will be automatically created in the asset tree. The newly created element appears in this group.

If an ML model contains only elements based on diagnostic rules, the model is assigned the [Trained status](#). You can [start inference](#) for such an ML model. If the ML model contains untrained predictive elements and/or elliptic envelope-based elements, these must be trained before starting the inference.

Changing an ML model element based on a diagnostic rule

You can change the settings of an ML model element based on a diagnostic rule.

Parameters cannot be changed if the ML model is assigned the **Ready for publication** or **Published** status.

System administrators and users who have the **Edit untrained models** permission from the [Manage ML models](#) group of rights can edit elements of ML models. The functionality is available after [a license key is added](#).

To change an element of an ML model based on a diagnostic rule:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the element based on a diagnostic rule that you want to edit.
A list of options appears on the right.
3. In the upper-right corner of the window, click the **Edit** button.
4. Adjust the diagnostic rule settings, if needed. For a description of the settings, see the [instructions on adding a diagnostic rule-based ML model element](#).

Editing the **Reminder period (sec)**, **Period of recurring alert suppression (sec)**, **Anomaly observation interval (sec)**, and/or **Anomaly duration share in interval** settings changes anomaly detection sensitivity. These parameters are unavailable for editing if the ML model is in the **Historical inference in progress** or **Streaming inference in progress** state.

5. In the upper-right corner of the window, click the **Save** button.
6. If **Grid step (sec)** has been edited, confirm the changes.

Adding an elliptic envelope-based ML model element

System administrators and users who have the **Create models** permission from the [Manage ML models](#) group of rights can add ML model elements. The functionality is available after [a license key is added](#).

To add an elliptic envelope-based ML model element:

1. In the [main menu](#), select the **Models** section.
2. To add an elliptic envelope, do the following:
 - a. In the asset tree, next to the name of the ML model you want to add an elliptic envelope to, open the vertical menu **...** and select **Create element**.
 - b. In the window that opens, select the **Elliptic envelope** item type.
 - c. Click the **Create** button.

A list of options appears on the right.

3. In the **Name** field, specify the name of the ML model element.
4. Enter a description for the ML model element in the **Description** field.
5. In the **General element settings** block, do the following:
 - a. In the **Reminder period (sec)** field, specify the period in seconds, upon reaching which the ML model will generate a repeated incident if anomalous behavior is retained in each UTG node.
The default value of this setting is **0**, which corresponds to no reminders.
 - b. In the **Period of recurring alert suppression (sec)** field, specify the period in seconds during which the ML model does not log repeated incidents for the same element.
The default value of this setting is **0** (repeat incidents not suppressed).
 - c. In the **Anomaly observation interval (sec)** field, enter the period (in seconds) during which the anomalous behavior of the tag is monitored to make a decision regarding incident registration.
 - d. In **Anomaly duration share in interval**, enter as a decimal fraction the proportion of the period in **Anomaly observation interval (sec)** that must elapse for the ML model element to register an incident.
You can specify a value in the range of **0** to **1**.
 - e. In the **Color of incident dot indicators** field, select the color of the indicator points of the incidents logged by the ML model element on the graphs in the **Monitoring** and **History** sections. This color will also be used to display the graph of the artifact generated by this element.
 - f. If necessary, in the **Incident status** drop-down list, select a status to be automatically [assigned to incidents](#) logged by the ML model element.
 - g. If necessary, in the **Incident cause** drop-down list, select the cause to be automatically [set for incidents](#) logged by the ML model element if this cause is known in advance.
 - h. In the **Detection threshold** field, specify the threshold value upon reaching which an incident is registered.
The value of this parameter will be automatically adjusted after training the ML model element. If necessary, you can [change the value of this parameter](#).

- i. If required, in the **Expert opinion** field, specify the expert opinion that will be automatically generated for incidents registered by the ML model element if the contents of this opinion are known in advance.
6. In the **Grid step (sec)** field, specify the element's UTG period (in seconds) expressed as an integer or decimal.
7. In the **Input tags** drop-down list, select one or several tags to include in the ML model.
8. In the upper-right corner of the window, click the **Save** button.

When creating the first ML model element, an **Elliptic envelopes** group will be automatically created in the asset tree. The newly created element appears in this group.

The ML model element will be assigned the **Not trained** status, and the ML model to which the added element belongs will be assigned the **Not trained status**. To [run inference on the ML model](#), all of its [predictive elements](#) and [elliptic envelope-based elements must be trained](#).

Editing an elliptic envelope-based ML model element

You can edit the settings of an elliptic envelope-based ML model element.

Parameters cannot be changed if the ML model is assigned the **Ready for publication** or **Published** status.

System administrators and users who have the **Edit untrained models** permission from the [Manage ML models](#) group of rights can edit elements of ML models. The functionality is available after [a license key is added](#).

To edit an elliptic envelope-based ML model element:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the elliptic envelope-based element that you want to edit.
A list of options appears on the right.
3. In the upper-right corner of the window, click the **Edit** button.
4. Adjust the elliptic envelope settings, if needed. For a description of the settings, see the [instructions on adding an elliptic envelope-based ML model element](#).

Editing the **Reminder period (sec)**, **Period of recurring alert suppression (sec)**, **Anomaly observation interval (sec)**, **Anomaly duration share in interval**, and/or **Detection threshold** settings changes anomaly detection sensitivity. These parameters are unavailable for editing if the ML model is in the **Historical inference in progress** or **Streaming inference in progress** state.

5. In the upper-right corner of the window, click the **Save** button.
6. If you have edited **Grid step (sec)** and/or **Input tags**, confirm that you want to save the changes.
After changes are made to these parameters, the ML model element must be [retrained](#).

The element will be assigned the **Not trained** status.

Cloning of the ML model element

You can create an ML model element by cloning an element of any ML model. Copying creates an ML model element whose status and state, architecture settings, and learning settings match those of the original at the time of copying. If the original element of the ML model is trained at the time of its copying, the new element of the ML model will display the learning results of the original element.

The functionality is available after [a license key is added](#).

An element used in an imported ML model cannot be copied.

To copy an ML model element:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, next to the name of the ML model element that you want to copy, open the vertical menu ... and select **Copy element**.
The **Copy element** pane appears on the right.
3. Specify the name of the element in the **Name** field.
By default, an ML model element is assigned a name in the following format: < name of the original ML model element>_Cloned_ <date and time of cloning>.
4. In the **Model** drop-down list, select the ML model to copy the selected item to.
5. Click the **Save** button.

The new ML model element will be displayed within the corresponding group of elements of the selected ML model.

Removing an ML model element


When removing an ML model element, Kaspersky MLAD also deletes the inference results of the selected ML model element.

System administrators and users who have the **Remove models** permission from the [Manage ML models](#) group of rights can remove elements of ML models. The functionality becomes available after adding a [license key](#).
You cannot delete an element used in an imported ML model.

To remove an ML model element:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model element that you want to delete.

A list of options appears on the right.

3. In the upper-right corner of the window, click .
4. In the window that opens, confirm the deletion of the ML model element.

Cloning an ML model

System administrators and users who have the **Copy models** permission from the [Manage ML models](#) group of rights can clone ML models.

The functionality is available after [a license key is added](#).

You can create an ML model by cloning a previously added ML model. When cloning, a new ML model is created. The new ML model contains the same elements, statuses and settings of the ML model and its elements, as well as the training state of the elements as the ones of the ML model being cloned at the time of its cloning.

After cloning an ML model that was created manually or from a template based on a manually created ML model, you can [add predictive elements](#), [elliptic envelope-based elements](#), and/or [diagnostic rule-based elements](#) to the cloned ML model, and edit or [delete them](#).

After cloning an ML model that was imported into the application or created using a template based on an imported ML model, you cannot change the set of elements of the cloned ML model.

Before [running inference](#), you can change the training settings and retrain the elements of the copied ML model.

To clone an ML model:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model that you want to copy.

A list of options appears on the right.

3. In the upper-right corner of the window, click .

The **Model copying** pane appears on the right.

4. In the **Name** field, specify the ML model name.

The ML model name must not be longer than 100 characters.

By default, an ML model is assigned a name in the following format: < name of the original ML model>_Cloned_ <date and time of cloning>.

5. In the **Asset** drop-down list, select the asset to which you want to assign the new ML model.
6. Click the **Save** button.

The new ML model displays in the **Models** group of the asset tree. The **Models** group is created automatically and displayed as part of the selected section of the asset tree. If the cloned ML model contains predictive elements, elliptic envelope-based elements, and/or diagnostic rule-based elements, the **Models** group will display the **Predictive elements**, **Elliptic envelopes**, and/or **Rules** subgroups, respectively.

Working with ML model templates

This section provides instructions on working with ML model templates.

The functionality is available after [a license key is added](#).

You can [create a template](#) of an existing ML model to reuse its algorithm structure, set of elements, and training state at the time of the template creation. You can use a created template to [add new ML models](#).

If the original ML model used as a template was [created manually](#), you can add [predictive elements](#), [elliptic envelope-based elements](#), and/or [elements based on diagnostic rules](#) to the ML model created based on such template, as well as modify or [delete them](#).

If the original ML model used to create a template was [imported](#) to Kaspersky MLAD, the set of elements of the ML model created based on such a template cannot be changed.

Before [run inference on the ML model](#), [train all of its predictive](#) and [elliptic envelope-based elements](#).

Creating a template based on an ML model

System administrators and users who have the **Create model templates** permission from the [Manage ML models](#) group of rights can create templates based on ML models. The functionality is available after [a license key is added](#).

You can create an ML model template based on a previously added ML model. The created templates retain the algorithm structure, set of elements, tag composition, and the training state of the source ML model.

You can generate a template from an existing ML model if all predictive and elliptic envelope-based elements have been trained, and conditions have been set for all diagnostic rule-based elements.

To create a template based on an ML model:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, next to the name of the ML model based on which you want to create a template, open the vertical menu **...** and select **Create template**.
A list of options appears on the right.
3. Specify the name of the template in the **Name** field.
You can enter up to 100 characters.
By default, a template is assigned a name in the format `Template_<ML model name>_<date and time of template creation>`.
4. To change the names of the template tags, in the **Template tag name** column specify the new names for the relevant tags.

The template tags are automatically assigned the names of the tags employed in the ML model that was used to create the template. You can specify any other names for template tags. For example, you can use the functional descriptions of the tag roles. The names of the template tags do not have to match the names of the tags of the ML model that was used to create the template.

5. Click the **Save** button.

The new ML model template appears in the **Templates** group of the asset tree. The **Templates** group is created automatically and displayed as part of the selected section of the asset tree.

Editing an ML model template

You can edit the settings of a created ML model template.

System administrators and users who have the **Edit model templates** permission from the [Manage ML models](#) group of rights can edit ML model templates. The functionality is available after [a license key is added](#).

To edit an ML model template:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the template that you want to edit.
A list of options appears on the right.
3. In the upper-right corner of the window, click the **Edit** button.
4. Adjust the settings of the ML model template, if needed. For a description of the settings, see [instructions on creating an ML model template](#).
5. Click the **Save** button.

Creating an ML model based on a template

System administrators and users who have the **Create models** permission from the [Manage ML models](#) group of rights can create ML models based on templates. The functionality is available after [a license key is added](#).

You can create a new ML model based on available templates. When creating an ML model, you can specify the IDs of tags that should be used in the new ML model.

To create an ML model based on a template:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, next to the name of the template that you want to use to create an ML model, open the vertical menu **...** and select **Create model**.
The **Creating a model** pane opens on the right.
3. Enter a name for the new ML model in the **Model name** field.

The ML model name must not be longer than 100 characters.

4. In the **Model tag name** column, for each template tag, in the asset tree select the tag that will be used by an ML model that is created from the template.
5. Click the **Save** button.

The new ML model displays in the **Models** group of the asset tree. The **Models** group is created automatically and displayed as part of the selected section of the asset tree. If the ML model contains predictive elements, elliptic envelope-based elements, and/or diagnostic rule-based elements, the **Models** group will display the **Predictive elements**, **Elliptic envelopes** and/or **Rules** subgroups, respectively.


The state of the created ML model will match the training state of the source ML model when the template was created.

Removing an ML model template

System administrators and users who have the **Delete model templates** permission from the [Manage ML models](#) group of rights can remove ML model templates. The functionality is available after [a license key is added](#).

You can remove an ML model template from Kaspersky MLAD. Deleting a template does not remove ML models based on this template.

To remove an ML model template:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model template that you want to delete.
A list of options appears on the right.
3. In the upper-right corner of the window, click .
4. Confirm deletion of the ML model template.

The selected ML model template will be removed from Kaspersky MLAD.

Changing the parameters of an ML model

You can change the settings of an ML model that was created manually, imported into Kaspersky MLAD, created from a template, or copied.

Markup editing is not available for imported ML models and ML models that were created by cloning imported ML models or based on a template of imported ML models.

System administrators and users who have the **Edit untrained models** permission from the [Manage ML models](#) group of rights can edit the settings of ML model elements. The functionality is available after [a license key is added](#).

To change the parameters of an ML model:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model whose settings you want to edit
A list of options appears on the right.
3. In the upper-right corner of the window, click the **Edit** button.
4. Adjust the ML model settings, if needed. For a description of the settings, see the [instructions on creating an ML model](#).
5. In the upper-right corner of the window, click the **Save** button.

Training an ML model predictive element

With Kaspersky MLAD, you can train a predictive element for an ML model that was created manually, imported into Kaspersky MLAD, created from a template, or copied.

System administrators and users who have the **Train models** permission from the [Manage ML models](#) group of rights can train elements of ML models. The functionality is available after [a license key is added](#).

To train an ML model element:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the predictive element that you want to train.
A list of options appears on the right.
3. Open the **Training** tab and click the **Edit** button in the upper-right corner of the window.
4. In the **Data selection interval** field, specify the data time interval on which you want to train the ML model.
5. To apply [markups](#) when selecting data for training the ML model within a selected interval, select one or several markups in the **Markups** field.
The selected markups will form a [learning indicator](#).
6. To view the data that will be selected by the markups, click **On graph**.
Markups are displayed in the colors that were specified when they were [created](#).
You can select a preset when viewing the data on the graph.
7. If you need to configure extended training settings, turn on **Advanced training settings** toggle switch.
8. In **Maximum training duration (sec)**, specify a maximum time in seconds the Kaspersky MLAD server can spend training an ML model.
9. In the **Validation split** field, use a decimal value to specify the share of the validation sample as a percentage of the entire dataset used to train the ML model.
You can specify a value in the range of 0 to 1.
The default value of this parameter is 0.2.

10. In the **Maximum epoch count** field, specify the maximum number of epochs for training the ML model.
The default value of this parameter is **500**.
11. In the **Patience** field, specify the number of epochs with no improvement in training quality to wait before stopping the ML model training process early.
Stopping the ML model training early avoids overfitting of the model. Training in this case is considered to be completed successfully.
The default value of this parameter is **15**.
12. In the **Resolution of training results graphs** field, use a decimal value to specify the graph resolution for displaying training results on the **Training results** tab.
You can specify a value in the range of **0** to **1**.
13. In the **Batch size** field, specify the number of selection items that must be sent for training within the iteration.
The default value of this parameter is **16**.

This parameter is not available for an element with TCN architecture.

14. In **Block count**, specify the number of blocks to split the ML model training dataset into.
The default value of this parameter is **4**.
15. In the **Inference mode** drop-down list, select one of the following values:
- If you want to load all datasets for training into RAM, select **Fast inference**.
This inference mode allows you to perform inference faster.
 - If you want to load datasets into RAM one at a time, select **Memory saving mode**.
This inference mode allows inference to be performed with minimal expenditure of RAM, but it will take place slower than in **Fast inference** mode.
16. In the **Training mode** drop-down list, select one of the following values:
- To load the entire training dataset into RAM, select **Load all data to RAM**.
 - If you want to load one data block at a time into RAM and generate validation blocks from the end of the dataset, select **Validate at the end of the dataset**.
 - If you want to load one data block at a time into RAM without generating validation blocks, select **Run validation in each training data block**.
Validation data is generated from each training data block.
17. In the **Memory allocation mode** drop-down list, select one of the following settings:
- **Reserve minimum amount of free RAM**. If this setting is selected, the Trainer service will make sure that the minimum amount of memory specified in the **Amount of RAM, MB** field remains free when training the ML model.
 - **Reserve maximum available amount of RAM for model training**. If this setting is selected, the Trainer service will use the maximum amount of RAM specified in the **Amount of RAM, MB** field when training the ML model.

18. To consider previous training results while training an ML model on new data, enable the option to **Initialize model weights with values from previous training results**.
19. If you want to shuffle the data to improve the quality of ML model training, enable the **Shuffle data** option.
20. In **Initialization of pseudorandom number generator**, set a value for generating a pseudorandom number sequence.
21. In **Learning rate coefficient**, set the coefficient to be used for adjusting ML model element weights in each training iteration.
The default value of this parameter is **0.0001**.
22. In the **Training optimization algorithm** drop-down list, select one of the following algorithms:
- **Adadelata** is an adaptive learning rate-based algorithm for each dimension.
 - **Adagrad** is the algorithm in which the learning rate depends on the update rate of settings during learning.
 - **Adam** is the algorithm based on adaptive computing of the first-order and second-order momentum of setting distribution.
 - **RMSprop** is the algorithm that uses a moving average of the squared gradient to adaptively normalize the learning rate at each step.
 - **SGD** is the Stochastic Gradient Descent algorithm.
23. In the **Loss function** drop-down list, select one of the following functions:
- **MSE**: for calculating the root mean square error.
 - **MSLE**: for calculating the logarithmic mean error.
 - **MAE**: for calculating the mean absolute error.
 - **MAPE**: for calculating the mean absolute percentage error.
24. In the upper-right corner of the window, click the **Save** button.
If you change the learning parameters for a previously trained element, you have to confirm the changes.
25. In the information block located above the training settings, click the **Train element** button.
The information block will show the number of the current training epoch of the element.

After the training is complete, you can [view the training results of an ML model element](#) in the **Training results** tab.


After all predictive elements and elliptic envelope-based elements that are part of the ML model have been successfully trained, the model will be assigned a [status of Trained](#). If required, you can retrain the ML model element by clicking **Restart training**.

Training an elliptic envelope-based ML model element

With Kaspersky MLAD, you can train an elliptic envelope-based element for an ML model that was created manually, imported into Kaspersky MLAD, created from a template, or copied.

System administrators and users who have the **Train models** permission from the [Manage ML models](#) group of rights can train elements of ML models. The functionality is available after [a license key is added](#).

To train an ML model element:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the elliptic envelope-based element that you want to train.
A list of options appears on the right.
3. Open the **Training** tab and click the **Edit** button in the upper-right corner of the window.
4. In the **Data selection interval** field, specify the data time interval on which you want to train the ML model.
5. To apply [markups](#) when selecting data for training the ML model within a selected interval, select one or several markups in the **Markups** field.
The selected markups will form a [learning indicator](#) .
6. To view the data that will be selected by the markups, click **On graph**.
Markups are displayed in the colors that were specified when they were [created](#).
You can select a preset when viewing the data on the graph.
7. If you need to configure extended training settings, turn on **Advanced training settings** toggle switch.
8. In **Sample fraction for estimating the mean and covariance**, enter as a decimal fraction the proportion of the training sample the covariance and mean are being calculated on.
You can specify a value in the range of 0 to 0.5 inclusive.
9. In **Outliers in sample**, enter as a decimal fraction the proportion of outliers (anomalies) in the training sample.
You can specify a value in the range of 0 to 1. This setting automatically overrides the **Detection threshold** as set when [the element was created](#). As the percentage of outliers in the training data increases, the threshold for registering an incident decreases. After training the element, you can [adjust](#) the incident registration threshold manually.
10. In **Initialization of pseudorandom number generator**, set a value for generating a pseudorandom number sequence.
11. In the **Resolution of training results graphs** field, use a decimal value to specify the graph resolution for displaying training results on the **Training results** tab.
You can specify a value in the range of 0 to 1. The higher the value, the better the quality of the graphs.
12. If you assume that the tag values are centered and their mean is equal to zero, turn on **Data is centered** toggle switch.
13. In the upper-right corner of the window, click the **Save** button.
If you change the learning parameters for a previously trained element, you have to confirm the changes.
14. In the information block located above the training settings, click the **Train element** button.

After the training is complete, you can [view the training results of an ML model element](#) in the **Training results** tab.

After all predictive elements and elliptic envelope-based elements that are part of the ML model have been successfully trained, the model will be assigned a [status of Trained](#). If required, you can retrain the ML model element by clicking **Restart training**.

Viewing the training results of an ML model element

You can view the results of training predictive elements and elliptic envelope-based elements.

System administrators and users who have the **Train models** permission from the [Manage ML models](#) group of rights can view the results of training ML model elements. The functionality is available after [a license key is added](#).

To view the training results of an ML model element:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model element whose training result you want to view.
A panel with the settings of the selected element will appear on the right.
3. Select the **Training results** tab.

If the ML model element has been successfully trained, the following information about the training results is displayed in the **Training results** tab:

- Message about successful completion of training of an ML model element.
If you want to view the training settings for an element that were specified during its [creation](#), click the **Training settings** button.
- **User**: The name of the user who started training the ML model element.
- **Start of training**: The date and time when the Trainer service began training the ML model element.
- **End of training**: The date and time that training of the ML model element finished. ML model element weights have been updated by the Trainer service.
- **Training interval**: The time spent by the Kaspersky MLAD server for training the ML model element.
- **Total training duration**: The duration of data time intervals considering the markups in the training dataset.
- **Number of UTG nodes**: The number of UTG nodes included in the training set.
- Graphs with learning results for ML model predictive elements:
 - **Training and validation errors**: A graph showing the training and validation errors for each training epoch.
 - **Model prediction**: Graphs showing model predictions for the output tags and the overall prediction error.
- Graphs with learning results for ML model elliptic envelopes:
 - **Tag deviation**—a graph showing the distance of a point, representing the state of a monitored asset at every moment in time within the phase space, from the center of the elliptical region of normal states. The

orange horizontal line marks the threshold. It indicates the farthest point at which a condition can still be considered normal.

- **Tag values:** graphs showing the values of each tag during training.
- **Tag value distribution:** histograms that show the distribution of values for each tag during training.
- **Tag correlation:** matrix that shows relationships between tags used when training an ML model element.

Starting and stopping ML model inference

You can start or stop the inference of an ML model in a [status of Trained or Published](#) on historical or incoming telemetry data.

The functionality is available after [a license key is added](#).

To start the ML model inference:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model whose inference you want to run.
A list of options appears on the right.
3. Select the **Inference** tab.
4. In the **Inference type** drop-down list, select one of the following values:
 - **Historical** to run ML model inference on historical telemetry data. If you select this value, specify the data time interval for running the ML model.
 - **Real-time** to run ML model inference on telemetry data that is being received in real time.
5. Click the **Start** button.
When starting a historical inference on previously inferred data, previous inference results will be deleted.
If historical inference was started, Kaspersky MLAD will add the ML model to the inference queue.

To stop the ML model inference:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model whose inference you want to stop.
A list of options appears on the right.
3. Select the **Inference** tab.
4. Click the **Stop** button.
Kaspersky MLAD will stop inference for the selected ML model.

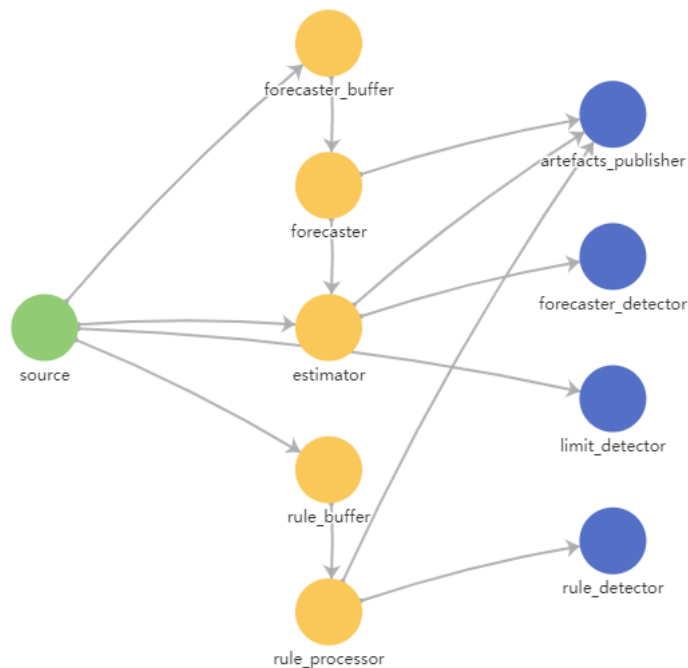
Viewing the data flow graph of an ML model

You can view the data flow graph in ML models.

The functionality is available after [a license key is added](#).

To view the data flow graph in an ML model:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select an ML model, for which you want to view the data flow graph.
A list of options appears on the right.
3. Select the **Data flow graph** tab.
The ML model data flow graph is displayed on the right.
4. If you need to view the settings of a graph element, move the mouse cursor over it.
A window listing the values of settings of the selected element will be displayed.



ML model data flow graph

Preparing an ML model for publication

You can prepare the model for publication after training it and checking the inference results. An ML model ready for publishing cannot be modified.

System administrators and users who have the **Edit untrained models** permission from the [Manage ML models](#) group of rights can prepare an ML model for publication. The functionality is available after [a license key is added](#).

To prepare an ML model for publication:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model you want to prepare for publication.
A list of options appears on the right.
3. Click the **Prepare to publish** button.

The ML model is assigned the [Ready for publication status](#). Notify the officer responsible for [publishing the ML model](#) that it is ready, or, if you have the required permissions, publish the ML model.

To make changes to the ML model before publishing, click the **Back to edit mode** button. The ML model will revert to a status of **Trained**.

Publishing an ML model

You can publish an ML model. The ML model will register incidents detected in real-time data from the monitored asset once [inference begins](#). Recorded incidents, unlike those inferred by unpublished ML models, necessitate actions and reporting in production.

System administrators and users who have the **Edit untrained models** permission from the [Manage ML models](#) group of rights can publish ML models. The functionality is available after [a license key is added](#).

To publish an ML model:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model that you want to publish.
A list of options appears on the right.
3. Click **Publish**.

The ML model is assigned the [Published status](#).


Removing an ML model

You can delete previously created and/or imported ML models from Kaspersky MLAD. You can reload a previously imported and subsequently deleted [ML model](#), if needed.

After the ML model is removed, its artifacts, such as predictions, individual errors, prediction errors, or rule progress indicators, as well as incidents registered by the ML model, will be deleted.

System administrators and users who have the **Remove models** permission from the [Manage ML models](#) group of rights can remove ML models. The functionality is available after [a license key is added](#).

To remove an ML model:

1. In the [main menu](#), select the **Models** section.
2. In the asset tree, select the ML model to be deleted.
A list of options appears on the right.
3. In the upper-right corner of the window, click .
4. Confirm deletion of the ML model.

The selected ML model will be removed from Kaspersky MLAD.



Managing presets

A *preset* is a set of tags generated by a user in arbitrary order or created automatically when an incident is registered. A set of tags in a custom preset can correspond to a certain aspect of the technological process or a section of the monitored asset.

You can [create](#) custom presets under **Presets**. The presets created by you are displayed only for your user account.

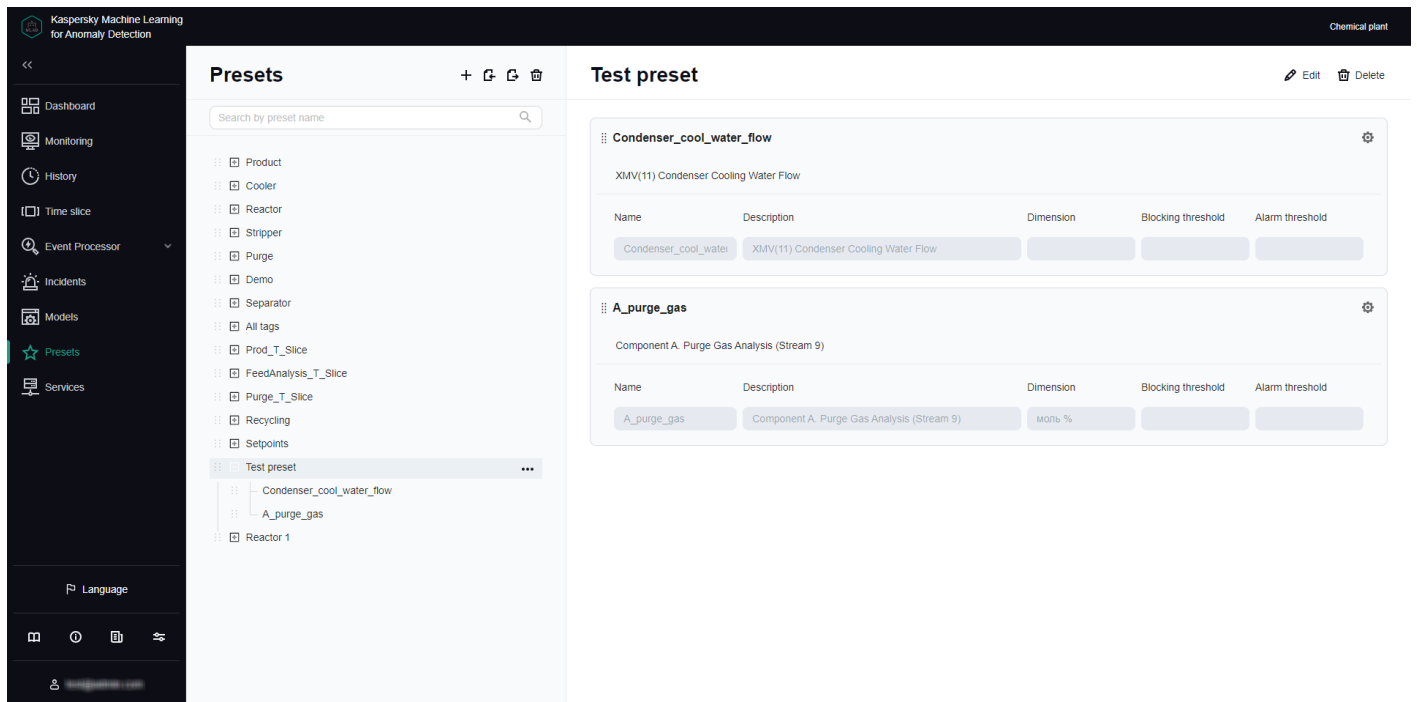
When creating presets, you can do the following:

- Select tags whose data to display on the graphs under [History](#) and [Monitoring](#).
- Manage graphic areas within a preset. A *graphic area* is a collection of tags whose data is displayed together by overlapping on a single graph in **History** and **Monitoring** sections. A graphic area can display data for one or more tags in a preset.
- Control the display of preset tags in [Time slice](#) section. To do this, you can define expressions with basic arithmetic operations, such as addition, subtraction, multiplication, or division, to calculate the values of the tags within the preset.

Custom presets and their tags are displayed as a tree on the left side of the window. You can use the  and  buttons to the left of the preset names to display or hide tags included in the presets.

To view incoming data on the graphs under **History** and **Monitoring**, [upload presets](#) to Kaspersky MLAD from a JSON file.

You can [edit](#) presets, [save](#) them to JSON, or [delete](#).




Presets section

Viewing a preset

You can view presets you [created](#) and [uploaded](#) to Kaspersky MLAD for your monitored asset.

To view a preset:

1. In the [main menu](#), select the **Presets** section.
2. On the page that opens, select the relevant preset from the preset tree on the left.
A list of the graphic areas included in the selected preset will appear on the right.
3. To view the details of the tags in a preset, do the following:
 - a. Click the  button on the left of the preset name.
A list of included tags is displayed under the selected preset.
 - b. To view tag settings, click the tag in the preset.
A list of tag settings defined when the tag was created appears on the right.

If necessary, you can [change the preset](#) or [create a new preset](#).

Creating a preset

Kaspersky MLAD allows you to create presets and configure how the results are displayed in the **Time slice** view.

You can define graphic areas. Graphic areas define the composition of tags whose data is displayed on the same graph and the place where the tag graph is displayed under **History** and **Monitoring**. Graphic areas that contain a single tag are referred to as *single-tag* graphic areas. Single-tag graphic areas are created automatically after you select tags for the preset.

If necessary, you can specify an expression to use for calculating the values of tags in the preset to display these values on the graph in the **Time slice** section. For example, you can use the specified expressions to view individual tag errors, predicted tag values, and the values of tags received from the monitored asset's sensors at the same time. You can use the following variables in your expressions:

- *\$tagValue* is the received tag value (based on the results of monitoring).
- *\$tagError* is the individual tag error.
- *\$tagPrediction* is the predicted tag value.
- *\$tagX* is the X coordinate of the monitored asset's sensor location specified when creating the tag.
- *\$tagY* is the Y coordinate of the monitored asset's sensor location specified when creating the tag.
- *\$tagZ* is the Z coordinate of the monitored asset's sensor location specified when creating the tag.

To create a new preset:

1. In the [main menu](#), select the **Presets** section and click the **+** button.

The **Create preset** pane appears on the right. The asset tree appears on the left side of the pane.

2. Specify the name of the preset in the **Preset name** field.

3. In the asset tree, select the check boxes next to the tags that you want added to the preset.


To include all tags associated with a particular preset, choose that preset from **Select preset**. Find and add a tag by typing its name into the **Search by tag name** field, then select the relevant tag.

If you need to delete tags from a preset, clear the check boxes next to the tags you want to delete in the asset tree.

4. Click **Save selection**.


Graphic areas automatically generated for each selected tag appear on the right. The number of graphic areas corresponds to the number of tags in the preset.

5. To change the preset name, do one of the following:

- Click **Delete all areas** under **Graphic areas** to delete all graphic areas associated with the preset.
- Click  in the upper right corner of the graphic area section and confirm your choice.




Deleting a graphic area does not remove the tags associated with it from the preset. You can add tags from deleted graphic areas to other areas.


6. To edit an existing graphic area, do one the following:

- To add a tag to a graphic area, click the plus sign in the graphic area section and select a tag.
To copy the name of a selected tag, click the  button on the right of the relevant name. You can use the copied tag name, for example, as the name of a graphic area.
- To remove a tag from a graphic area, click the **x** button next to the tag in the graphic area section.

7. To add a graphic area, do the following:

- a. Click **Empty area**.

- b. Select a tag to add to the graphic area.
- If the selected tag is used within a single graphic area, the browser window displays a corresponding informational message.
- To copy the name of a selected tag, click the  button on the right of the relevant name. You can use the copied tag name, for example, as the name of a graphic area.
- c. To add a further tag or several tags to the graphic area, click the plus sign and select tags.
- To copy the name of a selected tag, click the  button on the right of the relevant name. You can use the copied tag name, for example, as the name of a graphic area.
8. To add single-tag graphic areas for tags unused in other graphic areas in the preset, click **Single-tag graphs**.
- On the right, single graphical areas will be displayed for tags that were not used in previously created graphic areas.
9. If you need to change the position of a graphic area within a preset, drag the graphic area up or down by holding the dots on the left (:::) of its name.
10. To change the settings for displaying the graphic area under **History** and **Monitoring**, do the following:
- Click the  button in the upper-right corner of the graphic area section.
 - A panel with the graphic area display settings appears on the right.
 - In **Graphic area name**, provide a name for the graphic area.
 - Enter a new graphic area description in **Description**.
 - In the **Axis scale mode** drop-down list, select one of the following modes:
 - **Single axis mode**: uses one Y-axis to display tag data.
 - **Cast mode**: scales data along the Y-axis for each tag individually, irrespective of data from other tags in the graphic area.
 - If you have selected single axis mode, do one of the following:
 - Turn on the **Automatic** toggle button to automatically scale the graph according to the minimum and maximum data values for all tags in the graphic area.
 - Turn off the **Automatic** toggle button and provide an upper and lower display boundary for tags in the graphic area.

If tag values go beyond the defined boundaries, they will not be displayed in the graphic area. The permissible boundaries for displaying tag values take priority over the display of blocking thresholds, even if the [Always display blocking threshold](#) function is enabled.
 - To add further horizontal threshold lines for tags on the graph, click **Add threshold line**, and provide a threshold value and line color to display on the graph.
- Additional threshold lines help to visualize tag value fluctuations within certain limits. You can add multiple threshold lines.
- If you need to delete a previously added threshold line, click the  button next to the specified threshold value and the color of the relevant line.
- Click the **Save** button.
11. To configure the display of preset in the **Time slice** section:

a. Click **Display preset in the Time slice section**.

The **Display preset in the Time slice section** panel appears on the right.

b. Turn on the **Display preset in the Time slice section** toggle button.

c. In the **X-axis caption** field, enter the caption to be displayed on the x-axis.

d. To display on the graph the values of preset tags calculated according to the expression, click **Add graph** and specify the following values:


- In the **Name** field, enter the name of the expression to be used for calculating the tag values.
- In the **Y-axis caption** field, enter the caption to be displayed on the y-axis.
- In the **Expression for calculation** field, enter an expression for calculating tag values.

You can define expressions with simple arithmetic operations (such as addition, subtraction, multiplication, and division). For example, if the sensors are reporting temperature in Fahrenheit, you can use the following expression to display the temperature in Celsius:

```
5/9 * ($tagValue - 32)
```

- In the **Graph color** field, select the color of the graph that will be displayed for the preset in the **Time slice** section.

If necessary, you can add multiple expressions for the **Time slice** section.

e. If you want to delete an expression from a preset for the **Time slice** section, click the  button on the right of the relevant expression.

f. Click the **Save** button.

12. Click the **Save** button.

If any preset tags have not been added to at least one graphic area, the browser window displays an informational message. Tags that are not being used in graphic areas will not be displayed on the graphs in the **History** and **Monitoring** sections.


The new preset is displayed under **Presets** in the presets tree on the left and in the presets drop-down list under **History** and **Monitoring**. The preset for which step 11 of these instructions was performed will also be displayed in the drop-down list of presets in the **Time slice** section.

To change the position of presets in the presets tree, drag the preset up or down the tree by clicking and holding the dots (:::) to the left of the preset icon.

Loading presets from a file

You can load presets to Kaspersky MLAD from a [JSON file](#).

To import presets into Kaspersky MLAD from a file:

1. In the [main menu](#), select the **Presets** section.
2. In the upper part of the opened page, click the  button.

3. Select the JSON file containing the preset description on your local drive.

The selected file will be imported into Kaspersky MLAD. The new presets will appear in the preset list in addition to the previously [created presets](#). If the name of the preset imported from a file matches the name of an existing preset, the new preset will be assigned a name set in the JSON file.

Editing a preset

You can edit the presets you [created](#) or [uploaded](#).

You can remove tags from a preset using the vertical menu in the preset tree, if needed. To do this, open the vertical menu **...** to the right of the tag, select **Delete tag**, and confirm your choice.

To edit a preset:

1. In the [main menu](#), select the **Presets** section.

2. Perform one of the following actions:

- In the preset tree on the left, select the preset and click **Edit** in the upper right corner of the page.
- In the vertical menu **...** to the right of the preset, select **Edit preset**.

The **Edit preset** pane appears on the right. The asset tree appears on the left side of the pane.

3. Edit the following preset settings if needed:

- Preset name.
- The composition of tags in the preset.
When deleting a tag from a preset, Kaspersky MLAD also deletes it from all graphic areas that have used the tag. A single graphic area is automatically created for each tag added to the preset.
- The composition of graphic areas in the preset.
Deleting a graphic area does not remove the tags associated with it from the preset. You can add tags from deleted graphic areas to other areas.
- The composition of tags in the graphic areas.
- The location of the graphic area within the preset.
- Settings for displaying the graphic area in **History** and **Monitoring** sections.
- Settings for displaying the preset in the **Time slice** section

4. Click the **Save** button.

If any preset tags have not been added to at least one graphic area, the browser window displays an informational message. Tags that are not being used in graphic areas will not be displayed on the graphs in the **History** and **Monitoring** sections.


The changed preset will be updated in the presets tree under **Presets** and in the presets drop-down list under **History** and **Monitoring**. The modified preset whose display settings in the **Time slice** section have been defined will also be displayed in the preset drop-down list in **Time slice**.

To change the position of presets in the presets tree, drag the preset up or down the tree by clicking and holding the dots (::) to the left of the preset icon.

Saving presets to a file

You can save the presets you [created](#) and [uploaded](#) to Kaspersky MLAD as a JSON file.

To save the presets you created and uploaded to Kaspersky MLAD to a file:

1. In the [main menu](#), select the **Presets** section.
2. In the upper part of the opened page, click the  button.

The presets that were created and uploaded to Kaspersky MLAD will be saved to a JSON file on the local drive.

Delete presets


You can delete the presets you created or uploaded.

To delete a preset:

1. In the [main menu](#), select the **Presets** section.
2. Perform one of the following actions:
 - In the preset tree, select the preset and click **Delete** in the upper right corner of the page.
 - In the vertical menu ... to the right of the preset, select **Delete preset**.
3. Confirm preset deletion.

The preset will be deleted from the list of presets.

To delete all presets:

1. In the [main menu](#), select **Presets**, and click  above the preset tree.
2. Confirm preset deletion.

All presets will be deleted.

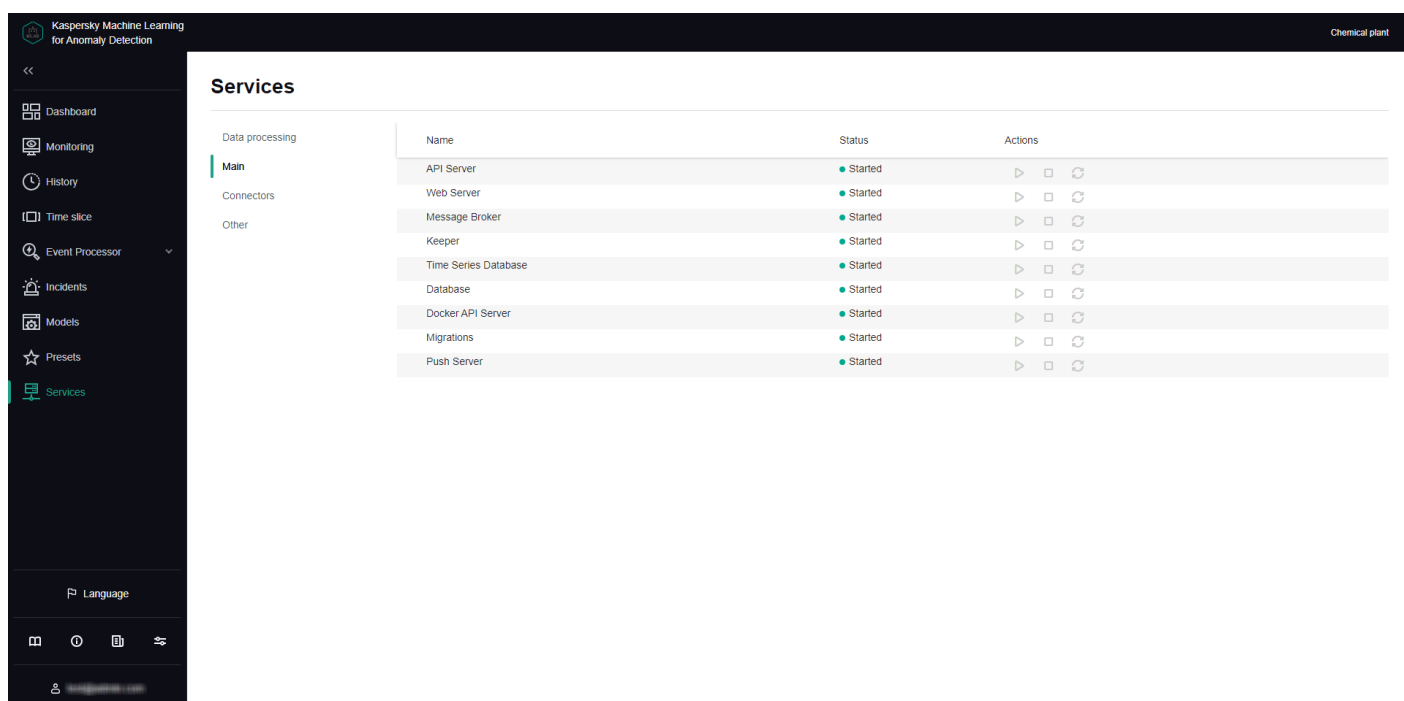
Managing services

The **Services** section displays a table containing information about [services](#) and their statuses. In the Kaspersky MLAD web interface, services are grouped by their functional scope, and the following information is displayed for each service:

- **Name** is the name of the service.
- **Status** refers to the current status of the service. Kaspersky MLAD provides the following service statuses:

- **Updating:** the service has been started, restarted, or stopped with one of the buttons in the **Actions** column.
- **Started:** the service is running.
- **Starting:** the service is starting.
- **Error when starting:** an error occurred when starting the service.
- **Stopped:** the service has been stopped.
- **Restarting:** the service is restarting.
- **Not initialized:** the service could not be started because Kaspersky MLAD had been started incorrectly.
- **Actions** are the available [actions](#) (start, stop, and restart).

Service statuses cannot be managed under **Main** in the Kaspersky MLAD web interface.



Services section

Viewing the statuses of services

You can view the statuses of [services](#) to make sure that the services were successfully started or stopped.

System administrators and users who have the **View statuses of application services** permission from the [Working with application services](#) group of rights can view the statuses of services.

Kaspersky MLAD checks the statuses of services every 30 seconds.

To view the statuses of services:

In the [main menu](#), select the **Services** section.

The **Services** section opens, displaying a table with available services, their statuses, and buttons for starting, stopping, and restarting services.

Starting, stopping, and restarting services

Kaspersky MLAD lets you start, stop and restart [services](#).

System administrators and users who have the **Manage statuses of application services** permission from the [Working with application services](#) group of rights can start, stop, and restart services. Management of service statuses under **Data processing** becomes available after adding a [license key](#).




Service statuses cannot be managed under **Main** in the Kaspersky MLAD web interface.

Under **Services**, you can manage the statuses of the following services:

- *Anomaly Detector*. This service must be enabled and [configured](#) if you need to analyze telemetry data and detect anomalies in the behavior of the monitored asset.
- *Trainer*. This service must be enabled if you need to train ML model elements on telemetry data.
- *Similar Anomaly*. This service must be enabled and [configured](#) if you need to group similar incidents.
- *Event Processor*. This service must be enabled and [configured](#) if you need to detect patterns and abnormal sequences of events.
- *Stream Processor*. This service must be enabled and [configured](#) if you need to convert telemetry data to a uniform temporal grid and log incidents when data loss is detected and data arrives too soon or too late.
- *HTTP Connector*. This connector must be enabled and [configured](#) if you need to receive telemetry data over HTTP or HTTPS.
- *OPC UA Connector*. This connector must be enabled and [configured](#) if you need to receive telemetry data over OPC UA.
- *AMQP Connector*. This connector must be enabled and [configured](#) if you need to receive telemetry data and incident registration messages over AMQP.
- *KICS Connector*. This connector must be enabled and [configured](#) if you need to receive telemetry data from Kaspersky Industrial CyberSecurity for Networks.
- *MQTT Connector*. This connector must be enabled and [configured](#) if you need to receive telemetry data and incident registration messages over MQTT.
- *CEF Connector*. This connector must be enabled and [configured](#) if you need to receive telemetry data and incident registration messages via CEF.
- *WebSocket Connector*. This connector must be enabled and [configured](#) if you need to receive telemetry data and incident registration messages over WebSocket.

- *Mail Notifier*. This service must be enabled and [configured](#) if you need to send registered incident alerts by email.
- *Logger*. This service must be enabled if you need to maintain and store Kaspersky MLAD logs. This service is enabled by default.

To start, stop, or restart a service:

1. In the [main menu](#), select the **Services** section.
2. On the opened page, select one of the following subsections: **Data processing**, **Connectors**, or **Other**.
3. Do one of the following for the relevant service:
 - If you want to start a service, click .
 - If you want to stop a service, click .
 - If you want to restart a service, click .

The new status of the service is displayed in the **Status** column.

Troubleshooting

This section describes possible problems in the operation of Kaspersky MLAD and methods for resolving them.

When connecting to Kaspersky MLAD, the browser displays a certificate warning

Problem

When attempting to connect to Kaspersky MLAD, the browser displays a warning that the security certificate or the established connection is not trusted. The contents of the warning depend on the specific browser being used.

Solution

After Kaspersky MLAD is installed, a self-signed certificate is used by default to connect to the web interface. When using a self-signed certificate, the browser displays a warning that the security certificate or the connection being established is not trusted. To obtain a trusted certificate, you need to contact the information security department or the information technology department of the Customer. To install the received trusted certificate, you need to contact a qualified technical specialist of the Customer, a Kaspersky employee, or a certified integrator. An employee can [update certificates](#) for connecting to Kaspersky MLAD using the web interface.

You can temporarily use a self-signed certificate to connect to Kaspersky MLAD (for example, during trial operation). When using a self-signed certificate, in the browser warning window select the option that lets you continue connecting. After connecting to Kaspersky MLAD, the browser window displays a warning about the certificate. The text of the message depends on the specific browser being used.

If the browser displays a warning after a trusted certificate is installed, then the certificate may have been spoofed by a malicious actor. Contact the [Technical Support](#).

The hard drive is running out of free space

Problem

The hard drive of the computer where Kaspersky MLAD is installed is running out of free disk space.

Solution

The computer must meet the [hardware and software requirements](#) to ensure proper functioning of the application.

Do not allow the hard drive to completely run out of free disk space. Periodically check for free space by using the `df -h` command.

To ensure that the application functions properly:

On the hard drive of the computer, free up sufficient space to satisfy the [minimum free disk space requirements](#). You can free up space on the hard drive in the following ways:

- Move backup copies of Kaspersky MLAD, the Kaspersky MLAD distribution package and extraneous files to another drive or server.
- Create an image of the drive on which Kaspersky MLAD is installed, replace the hard drive with a larger one, and deploy the image on the new drive.

The operating system restarted unexpectedly

Problem

Unexpected restart of a computer with Kaspersky MLAD installed.

Solution

Wait for the computer restart to finish. After the computer has restarted, the following statuses of Kaspersky MLAD are possible:

- Kaspersky MLAD has fully resumed normal operation.
- Kaspersky MLAD has not resumed normal operation.

If the problem persists, [restart Kaspersky MLAD](#). If the problem is not resolved after the restart, please [contact Kaspersky Technical Support](#). Be prepared to submit process logs of Kaspersky MLAD and other data when requested by Technical Support representatives. Process logs are located in the directories listed in the [Folders for storing application data](#) article. Root privileges in the operating system are required for providing access to logs.

Cannot connect to the Kaspersky MLAD web interface

Problem

Error! is displayed after entering a valid password to sign in to the Kaspersky MLAD web interface. Invalid server error.

Solution

Often, the error *Error! Invalid server error* occurs when the Kaspersky MLAD server cannot process a request. For example, this could happen if the hard drive of the server where the application is installed has run out of free space.

To restore proper operation:

On the hard drive of the server, free up sufficient space to satisfy the [minimum free disk space requirements](#). You can free up space on the hard drive in the following ways:

- Move backup copies of Kaspersky MLAD, the Kaspersky MLAD distribution package and extraneous files to another drive or server.
- If Kaspersky MLAD is installed on a virtual machine, increase the size of the virtual hard drive of the virtual machine.
- Create an image of the drive on which Kaspersky MLAD is installed, replace the hard drive with a larger one, and deploy the image on the new drive.

Data graphs or graphic areas are not displayed in the History and Monitoring sections

Problem

The **History** and **Monitoring** sections are not displaying data graphs or graphic areas.

This may be caused by the following:

- Presets were not imported into Kaspersky MLAD.
- The selected preset contains no tags.
- No graphic areas are defined in the selected preset.
- Time interval with no data is selected in the **History** section.
- The connector used to receive data from the monitored asset is not running.
- The monitored asset is powered off.

Solution

[Import](#) or [create presets](#) that contain tags and graphic areas.

To display data in the **Monitoring** section, make sure that the monitored asset is enabled. [Enable the connector](#) used to receive data from the monitored asset. [Select a time interval](#) and [preset](#).

To display data in the **History** section, [select a preset](#), [date and time interval](#) in which data on preset tags is available.

Events are not transmitted between Kaspersky MLAD and external systems

Problem

Events are not received by Kaspersky MLAD and/or alerts about the monitor activation are not sent to external systems.

Solution

To restore the exchange of events with external systems:

1. [Start](#) the Event Processor service and the CEF Connector.
2. When [configuring the Event Processor service](#), do the following:
 - a. In the **Event processor configuration file** field, upload the configuration file describing the event parameters.
 - b. In the **Interval for receiving batch events (sec.)** field, specify the time interval in seconds required to generate an episode, taking into account the speed of receiving events from the monitored asset.
3. To receive events in the [.env file](#), specify the IP address and port number used to connect to the external event source.
4. To send events, when [configuring the CEF Connector](#), specify the IP address and the port number for connecting to the external system.

Cannot load data to view in the Event Processor section

Problem

After restarting Kaspersky MLAD, it is impossible to upload data for [viewing the events history](#) and/or [pattern history](#) in the **Event Processor** section (the **Process request** button is not available). This problem also arises after [changing the settings of the Event Processor service](#).

Solution

Wait a few minutes. After Kaspersky MLAD is restarted, the state of the Event Processor service is restored. It may take several minutes to restore the state of the service if there is a significantly large number of processed events or registered patterns. Until the state of the Event Processor service is restored in the **Event Processor** section, requests are not fulfilled, data is not updated, and data received from the CEF Connector is not processed. This data is temporarily stored in the system message queue and is processed after the state of the Event Processor service is restored.

Data is incorrectly processed in the Event Processor section

Problem

A large number of short patterns is being created.

Solution

In the [Event Processor settings](#), increase the episode length to reduce the number of short patterns being registered.

Problem

A large number of monitor activation alerts is being received.

Solution

To reduce the number of monitor activation notifications, check previously created monitors and delete the ones you do not need. It is also recommended to update the following monitor activation parameters: **Sliding window** and **Threshold**.

Events are not displayed in the Event Processor section

Problem

When you make a request to [view the event history](#), the **Event Processor** → **Event history** section does not display the events that were displayed before.

Solution

Make sure that Kaspersky MLAD [saves the state of the Event Processor service](#) to the database table.

If the state of the Event Processor service is saved to a file in bit format, Kaspersky MLAD saves the state of the service with the frequency specified in the [Component backup frequency](#) field. When the Event Processor service is restarted, the results of processing of the event stream received by the Event Processor since the last time the service state was saved are lost.

Previously created monitors and the specified attention settings are not displayed in the Event Processor section

Problem

After [restarting](#) or [modifying the Event Processor service settings](#), the **Event Processor** → **Monitoring** section does not display [previously created monitors](#) and the [specified attention settings](#).

Solution

The Event Processor saves the created monitors and the specified attention settings after saving the state of the Event Processor service to a database table or a file in bit format. If Kaspersky MLAD saves the state of the service to a database table, it is recommended not to restart the Event Processor service or change its settings until the first episode of events from the monitored asset is processed, in order to save the created monitors and the specified attention settings. If the application saves the state of the Event Processor service to a file in bit format, it is recommended not to restart the Event Processor service or change its settings until the first backup of the service, in order to save the created monitors and the defined attention settings. The frequency of the Event Processor service backups depends on the value of the [Component backup frequency](#) setting.

To receive events, configure the settings and [start](#) the [Event Processor service](#) and the [CEF Connector](#). To process the registered incidents as events, [configure and start the Anomaly Detector service](#) and the connector required to [receive telemetry data](#) from the monitored asset. Go to the **Dashboard** section and make sure that events are received by Kaspersky MLAD in the online mode.

If the malfunction persists, please [contact Kaspersky Technical Support](#).

A markup result is not displayed

Problem

In the **Models** section, when viewing the markup graph, the data intervals to be selected by the markup appear uncolored.

Solution

The selected time interval may be too short and may not contain enough data for the markup to display the data intervals on the graph. In the **Scale** field, [specify a longer time interval](#) to display data on the markup graph.

A Trainer service stopped message is displayed

Problem

When you go to the **Training** tab of an ML model element, you see the **Trainer service is stopped** message. The Trainer service was started by a system administrator or a user with the **Manage statuses of application services** permission from the [Working with application services](#) permission group.

Solution

Wait approximately two seconds. If the Trainer service is running, the message will disappear automatically.

Training of an ML model element completed with an error

Problem

Training of an ML model element completed with an error. A training error may appear both immediately after training start and after several training epochs are completed.

Solution

If the training of an ML model element completed with an error immediately after training start, make sure that there is data available for all ML model tags within the training interval specified in the **Data selection interval** setting, while considering markup ([learning indicator](#)). To do this, when changing the training settings, click **On graph** and visually check that at least one data interval selected by the markup is displayed within the defined training interval, and that observations for all tags involved in training the ML model element are displayed within these data intervals.

If the training of an ML model element ended with an error after the training lasted for some time, this means that the selected architecture of the element, parameters of the element, or parameters of training the element are not suitable for accomplishing your task with the available data because the target quality metrics of the ML model cannot be achieved. Try to use an element of a different type or architecture, or change the element parameters or its training parameters.

Email notifications about incidents are not being received

Problem

Email notifications about incidents are not being received.

Solution

Make sure that [interaction between Kaspersky MLAD and the SMTP server is configured](#) correctly and that the [Mail Notifier service is running](#). [Create an incident notification](#) and specify the email addresses and types of incidents for which you need to send notifications. Use the **State** toggle switch to enable forwarding of notifications.

You need to change the Help localization language

Problem


The Help localization language must be changed in the Kaspersky MLAD web interface.

To change the Help localization language, visit the Technical Support website at <https://support.kaspersky.com/> select the required language in the upper right corner of the page.

Solution

To change the Help localization language in the application web interface:

1. Open the browser installed on your computer.
2. In the address bar of the browser, enter the web address of Kaspersky MLAD received from a qualified technical specialist of the Customer, a Kaspersky specialist, or a certified integrator.
3. To open Help, perform one of the following actions:
 - In the upper-right corner of the account credentials entry page that opens, click the **Help** link.

- On the login page, enter your user name and password, and click  in the lower left corner of the page that opens.

A new browser tab opens displaying the application Help.

4. Specify the necessary localization language in the web address:

- ru – if you want to open Help in Russian (for example, <https://<Kaspersky MLAD web address>/help/ru/171583.htm>).
- en – if you want to open Help in English (for example, <https://<Kaspersky MLAD web address>/help/en/171583.htm>).

After connecting to the application, you can change the language of the interface and Help in the user menu.

Contacting Technical Support

This section describes the ways to receive technical support, and its terms and conditions.

If you cannot find a solution to your problem in the application documentation or in one of the other sources of information about the application, you are advised to contact Technical Support. Technical Support experts will answer your questions about installing and using the application.

Technical support services are provided if you have an active *Technical Support Agreement*. The scope of provided technical support services is determined by the current *Technical Support Agreement*.

Before contacting Technical Support, please read the [technical support rules](#).

You can contact Technical Support experts by emailing them at mlad-support@kaspersky.com.

Technical Support experts may request that you provide information from the [Kaspersky MLAD logging system](#).

Limitations

Kaspersky MLAD has a number of limitations that are not critical for application operation:

- Alerts about the activation of the Event Processor service monitors are sent to external systems only using the CEF connector. Sending alerts by email is not available.
- Alerts about the activation of the Event Processor service monitors are not saved in the Kaspersky MLAD database.
- The Event Processor service processes only categorical data. All event parameter values are set in or converted to the string data type. Although the string values for each event parameter can be extremely diverse (up to tens of thousands of values), they are finite.
- Data processing performance for the current version of the Event Processor is about five thousand events per second and may decrease due to a large number of attention heads and monitors. A significant amount of RAM is required to work with a large stream of events. The estimate of the required volume depends on the stream of events, the variety of events in the stream, and the attention and monitor configuration settings.
- The Event Processor service is sensitive to how its settings are configured. Incorrectly defined event parameters, episode size and creation time, and attention configuration can significantly reduce service efficiency and performance.
- It is recommended to save the Event Processor service state to the database table. If the service state is saved to a file in bit format, Kaspersky MLAD saves the state of the Event Processor service according to the specified backup creation frequency for the service. It may take some time to save and restore the state of the Event Processor service (up to several minutes if there is a large volume of processed data). Restarting the service results in the loss of data since the last time it was saved to a file in bit format.
- Kaspersky MLAD is compatible with Kaspersky Industrial CyberSecurity for Networks version 4.0 and later.
- Kaspersky MLAD is designed to work with a tag stream whose rate does not exceed 10000 tags per second (short-term bursts of no more than 20% are permissible). If the tag stream rate exceeds the specified value, there may be delays in tag processing, prediction, and anomaly detection.
- Computers with Kaspersky MLAD and Kaspersky Industrial CyberSecurity for Networks installed must belong to the same network.
- You can save data during an application update only when updating Kaspersky MLAD 5.0.0-001 or later. To migrate from Kaspersky MLAD 4.0.0 to Kaspersky MLAD 5.0.0 or later, you need to perform a new installation of Kaspersky MLAD and manually import data from the previously installed Kaspersky MLAD 4.0.0. For detailed information on migration from Kaspersky MLAD 4.0.0 to Kaspersky MLAD version 5.0.0 or later, you are advised to contact Kaspersky Technical Support.
- Application rollback to the previously installed version is supported only for Kaspersky MLAD 5.0.0-001 or later.
- There is no capability to use model elements based on the XGBoost detector.
- In the asset tree, the **Assets** section does not display the icon that is selected when you create or edit tags or assets.
- In the **Incidents** section, in the period selection window, when moving to the right along the time axis, the blocks for selecting the beginning and end of the period display the year of the beginning of the defined period.
- After loading the configuration of assets and tags, the markups that were previously created for the loaded assets are not displayed in the **Models** section.

- In the **Monitoring** and **History** sections, the vertical axis that appears when you hover the mouse over the graph of the ML model element artifact at the bottom of the page does not coincide with the vertical axis that appears in the graphic areas at the top of the page.
- In the **Presets** section, a cleared check box is displayed for a tag included in a preset when searching by the name of this tag in the asset tree. If you select the check box for this tag, the **Save selection** button becomes available when there are no actual changes in the preset tags. The list of tags displayed in the asset tree according to the search query scrolls down when you select a tag within an asset that has a large number of tags.
- In the **Models** section, in some cases the **Import model** button is not successfully pressed in the vertical menu of the selected asset. For correct operation, you must move the mouse cursor over the right side of the button and click it.
- In the **Models** section, you cannot preview a markup if you have selected at least one of its tags for which no observations have been received by Kaspersky MLAD.
- In the **Models** section, the markup used in the imported ML model is displayed after re-importing the ML model, and a copy of the markup is created.
- In the **Assets** section, it is impossible to create new assets and/or tags after deleting the head element of the hierarchical structure (*Root*).
- In the **History** section, the deleted ML model elements are displayed in the drop-down list for selecting ML model elements.
- The Similar Anomaly service stops working when historical inference is restarted. When restarting historical inference, it is recommended to disable the Similar Anomaly service.
- In the **Models** section, when you zoom in on a markup, the horizontal scroll bar is not displayed while the markup is being [viewed](#). Some of the markup viewing controls become unavailable.
- Kaspersky MLAD stores the entire history of received tag values and predicted tag values. Therefore, you must estimate the potential storage volume based on the data update rate (tags per second) and the time interval for storing the telemetry data monitoring history.
- The **Models** section does not always display the results of training a predictive element after it has been successfully trained. You must refresh the page to display the results.
- The value of the **Monitored asset time zone** [setting that is defined by the system administrator in the main settings of Kaspersky MLAD](#) is applied only to dates and times when selecting time intervals for markups. This setting does not apply to other sections of the web interface in which the date and time can be selected for displaying data.

Appendix

This section provides information that supplements the main text of the document.

Settings of a .env configuration file

The settings of the configuration file can be changed only by a qualified technical specialist of the Customer, a Kaspersky employee, or a certified integrator.

The .env configuration file is filled in to configure the CEF Connector and has the settings described in the table below.

Settings of a .env configuration file

Setting	Description
CEF_CONNECTOR_INCOMING_IP	IP address used to connect an external event source to the CEF Connector.
CEF_INCOMING_PORT	Port number used to connect an external event source to the CEF Connector.
AUDIT_LOGGER_LOGIN	Database service user name.
AUDIT_LOGGER_PASSWORD	Database service user password.
DEFAULT_ADMINISTRATOR_USER	The name of the first user with the system administrator role.
DEFAULT_ADMINISTRATOR_PASSWORD	The password of the first user with the system administrator role.


To apply changes to the configuration file, [restart Kaspersky MLAD](#).

Settings and example of the Excel file containing tag and asset configuration

The configuration file is created by a qualified technical specialist of the Customer, a Kaspersky employee or a certified integrator. The system administrator [loads the configuration of assets and tags of a hierarchical structure](#) in the **Assets** section in the [administrator menu](#).

The configuration file contains the following tabs:

- **readme**: A tab containing general information about the configuration file.
- **directory_types**: A tab that describes the hierarchical structure asset types using the following settings:
 - **directory_type_id**: The asset type ID. The ID is assigned automatically when exporting the asset tree.
 - **directory_type**: A unique name for the asset type.
 - **parameter<parameter number>_label**: Names of special parameters, where <parameter number> corresponds to a value in the range from 1 to 5. If an asset of a given type does not have any special parameter, leave the corresponding field in the configuration file blank.

- **description:** The description of the asset type. This field is optional.
- **directories:** A tab that describes assets of the hierarchical structure using the following settings:
 - **directory_id:** The asset ID. The ID is assigned automatically when exporting the asset tree.
 - **directory_type:** The type of asset. The type is selected from the asset types specified on the **directory_types** tab.
 - **directory_type row:** The number of the row on the **directory_types** tab that describes the selected asset type. The field is filled in automatically.
 - **directory_name:** The unique name of an asset within its parent asset.
 - **directory_info:** The description of the asset. This field is optional.
 - **parent:** The parent asset. If the imported asset is at the top level of the asset hierarchy, leave the **parent** field blank.
 - **parent row:** The number of the row on which the selected parent asset is described. The field is filled in automatically.
 - **parent_id:** The ID of the parent asset. The ID is assigned automatically when exporting the asset tree.
 - **parameter<parameter number>:** Names of special parameters, where <parameter number> corresponds to a value in the range from 1 to 5. Names of special parameters are filled in automatically if special parameters are defined for the selected asset type.
 - **value <parameter number>:** Values of special parameters, where <parameter number> corresponds to a value in the range from 1 to 5. If an asset does not have a special parameter, leave the field for entering the corresponding value blank.
- **tags:** A tab that describes tags of the hierarchical structure using the following parameters:
 - **tag_id** refers to the tag ID. An ID value is assigned automatically in the range of 1 to 1,000,000 when exporting primary members of the hierarchical structure.
 - **tag_name** is the unique name of the tag.
 - **alternate_name:** A unique alternative name for the tag. This field is optional.
 - **tag_description** refers to a description of the tag.
 - **parent:** The parent asset to which the tag belongs. If the head element of the hierarchical structure is the tag imported by the parent element, leave the **parent** field blank.
 - **parent_row:** The number of the row on the **directories** tab that describes the selected parent asset. The field is filled in automatically.
 - **parent_id:** The ID of the parent asset. The ID is assigned automatically when exporting the asset tree.
 - **tag_type:** [Type of tag](#) . This field is optional.

- **PV**: To designate measurements or observed values of physical parameters.
- **CV**: To designate calculated values of physical parameters.
- **IV**: To designate tags that are independent of other tags.
- **SV**: To designate a setpoint.
- **MV**: To designate the controlled values of physical parameters.
- **B**: To designate tags in bit format.
- **X**: To designate cases that are not tagged.

If you are finding it difficult to determine the tag type, you can use a question mark (?) as the tag type instead.

- **tag_units**: The unit of measure for the tag.
- **red_min**: The lower threshold for the maximum permissible tag value. This field is optional.
- **red_max**: The upper threshold for the maximum permissible tag value. This field is optional.
- **yellow_min**: Lower signaling threshold, upon reaching which the operator should pay attention to the tag behavior. This field is optional.
- **yellow_max**: Upper signaling threshold, upon reaching which the operator should pay attention to the tag behavior. This field is optional.
- **validity_min**: The lower threshold for physically possible tag values. This field is optional.
- **validity_max**: The upper threshold for physically possible tag values. This field is optional.
- **display_min**: The lower boundary for displaying tag values on graphs. This field is optional.
- **display_max**: The upper boundary for displaying tag values on graphs. This field is optional.
- **scale**: The expression used to calculate the tag value from the value passed to Kaspersky MLAD. Instead of an expression, you can specify a specific number by which the value of the transmitted tag is to be multiplied. If the tag value does not need to be recalculated, leave this field blank.
- **comment**: A comment relating to the tag.
- **X** is the coordinate of the monitored asset's sensor location. This field is optional.
- **Y** is the Y coordinate of the monitored asset's sensor location. This field is optional.
- **Z** is the Z coordinate of the monitored asset's sensor location. This field is optional.
- **bias** is the value to be added to the tag value passed to Kaspersky MLAD. Enter 0 if no value conversion is required.

Values received in Kaspersky MLAD are converted if the **Scale obtained tag values** toggle switch is on in the settings of the employed connector. The **bias** value is added to the result from multiplying the resulting tag value by the **multiplier** value.

- **multiplier** is the multiplier for the tag value passed to Kaspersky MLAD. Enter **1** if no value conversion is required.

Values received in Kaspersky MLAD are converted if the **Scale obtained tag values** toggle switch is on in the settings of the employed connector. The resulting tag value is multiplied by the **multiplier** value before adding the **bias** value.

Below is an example of a XLSX file containing descriptions of assets and tags and their configuration.

Directory_types tab

directory_type_id	directory_type	parameter1_label	parameter2_label	parameter3_label	parameter4_label
	Factory	Process	Region		
	Unit	Vendor	Model	Year of manufacture	Responsible
	Setpoints				

Directories tab

directory_id	directory_type	directory_type row	directory_name	directory_info	parent	parent row	parent
	Factory	2	Chemical plant	Tennessee Eastman Process			
	Unit	3	Reactor	Chemical reactor	Chemical plant	2	
	Setpoints	4	Setpoints	Reactor setpoints	Chemical plant; Reactor	3	

tags tab

tag_id	tag_name	alternate_tag_name	tag_description	parent	parent_row	parent
	Reactor_pressure_setpoint		Reactor pressure setpoint	Chemical plant; Reactor; Setpoints	4	
	A_feed_stream1		Reagent consumption A	Chemical plant; Reactor	3	
	No reactor temperature response		Rule	Chemical plant	2	

Settings and an example of JSON file that describes presets

JSON file that describes presets is created by a qualified technical specialist of the Customer, a Kaspersky employee or a certified integrator. The file that describes presets is uploaded by the user in the **Presets** section.

JSON file defines information about presets under `presets`, which itself contains the following settings:

- `name`: preset name.
- `Code` is the preset ID in the Kaspersky MLAD database. An ID is assigned automatically when the file is uploaded.
- `Sort` is a sequential display number of the preset under **Presets**.
- `tags`: a list of IDs for the tags included in the preset
- `icon`: preset icon name
- `css_class`: the name of the CSS class for preset icon display
- `is_display_on_time_slice` determines whether the preset should be displayed under **Time slice**. If this is set to `true`, the preset is displayed under **Time slice**.
- `evaluations`: describes the graph in the **Time slice** section by using the following settings:
 - `axis_x_name`: x-axis label.
 - `evaluations`: describes the expression used to calculate tag values for graph display by using the following:
 - `name`: the name of the expression used to calculate tag values for graph display.
 - `yAxis`: y-axis label.
 - `expression` is an expression used to calculate the tag values
 - `expression_color`: preset graph color. Chart color is set with the help of RGB codes by using the following settings:
 - `a`: alpha opacity setting You can specify a value in the range of 0 to 1.
 - `b`: blue channel coordinate You can specify a value in the range of 0 to 255.
 - `g`: green channel coordinate You can specify a value in the range of 0 to 255.
 - `r`: red channel coordinate You can specify a value in the range of 0 to 255.
- `Graphic_areas` is a group of settings that describe the graphic areas within the preset.
 - `Name` is the name of the graphic area.
 - `Description` is the description of the graphic area.
 - `Sort` is a sequential display number of the graphic area in the preset under **Presets**, **History**, and **Monitoring**.

- `Display_max` is an upper limit for displaying tags associated with the graphic area. If `is_scale_mode_auto` is true, set `display_max` to null.
- `Display_min` is a lower boundary for displaying tags associated with the graphics area. If `is_scale_mode_auto` is true, set `display_min` to null.
- `Additional_thresholds` is a group of settings that describe additional threshold lines.
 - `Id` is the ID of the additional threshold line. An ID is assigned automatically when the file is uploaded.
 - `value`: threshold value.
 - `Color` is a color of the graph that will be displayed for the additional threshold line. Chart color is set with the help of RGB codes by using the following settings:
 - `a`: alpha opacity setting You can specify a value in the range of 0 to 1.
 - `r`: red channel coordinate You can specify a value in the range of 0 to 255.
 - `g`: green channel coordinate You can specify a value in the range of 0 to 255.
 - `b`: blue channel coordinate You can specify a value in the range of 0 to 255.
- `Scale_mode` is the axis scaling mode. You can specify one of the following values for the `scale_mode` setting:
 - `single_axis`: uses one Y-axis to display tag data on the graph.
 - `cast`: scales data along the Y-axis for each tag individually, irrespective of data from other tags in the graphic area.
- `Is_scale_mode_auto` is a setting that defines the method of graph scaling in single axis mode. When this is set to true, the tag graph is automatically scaled based on the minimum and maximum data values of all tags in the graphic area.
- `tag_id_list`: a list of tag IDs that are included in the graphic area.
- `Graphic_area_id` is the ID of the graphic area. An ID is assigned automatically when the file is uploaded.
- `Preset_id` is the ID of the preset the graphic area belongs to. An ID is assigned automatically when the file is uploaded.

Below is an example of a JSON file containing descriptions of presets.

```
{
  "presets": [
    {
      "name": "Product",
      "code": null,
      "sort": 0,
      "tags": [
        51,
        52,
        53,
        49,
```

```

50
],
"icon": "logout-signout",
"css_class": null,
"is_display_on_time_slice": false,
"evaluations": {
"axis_x_name": "",
"evaluations": []
},
"graphic_areas": [
{
"name": "F_product",
"description": "Content of F agent in final product",
"sort": 0,
"display_max": null,
"display_min": null,
"additional_thresholds": [],
"scale_mode": "single_axis",
"is_scale_mode_auto": true,
"tag_id_list": [
51
],
"graphic_area_id": null,
"preset_id": null
},
...
],
},
...
{
"name": "Cooler",
"code": null,
"sort": 0,
"tags": [
64
],
"icon": "graph",
"css_class": null,
"is_display_on_time_slice": false,
"evaluations": {
"axis_x_name": "",
"evaluations": []
},
"graphic_areas": [
{
"name": "Condenser_cool_water_flow",
"description": "Cooled condenser water flow rate control",
"sort": 0,
"display_max": null,
"display_min": null,
"additional_thresholds": [],
"scale_mode": "single_axis",
"is_scale_mode_auto": true,
"tag_id_list": [
64
],
"graphic_area_id": null,
"preset_id": null
}
]
}

```

```
]
}
```

Settings and an example of JSON file containing a configuration for the Event Processor service

The configuration file is created by a technical specialist of the Customer, a Kaspersky employee, or a certified integrator. The system administrator uploads the Event Processor configuration file when [configuring the Event Processor service settings](#).

When re-uploading a configuration file in which other event parameters are defined, the event parameters defined in the previous configuration file will become unavailable for configuration in the application web interface.

The CEF Connector receives information about each detected event from external systems in CEF format:

```
CEF:<CEF format version>|<name of the external system vendor>|<name of the external system application>|<version of the external system application>|<unique identifier of the event type>|<event description>|<event severity level>|<parameter 1>=<value of parameter 1> ... <parameter N>=<value of parameter N>
```

where:

- CEF:<CEF format version>|<name of the external system vendor>|<name of the external system application>|<version of the external system application>|<unique identifier of the event type>|<event description>|<event severity level>| is the event header.
- <parameter 1>=<value of parameter 1> ... <parameter N>=<value of parameter N> is the event body containing the sequence of <event parameter>=<event parameter value> pairs.

The configuration file describes the parameters in the events received by the CEF Connector. The names of event parameters in Kaspersky MLAD may coincide with the names of parameters received in CEF format by the CEF Connector. If necessary, you can specify other names for the parameters to be processed in Kaspersky MLAD according to certain rules. The rules for mapping event parameters are defined in the `mapping_fields` parameter of the configuration file.

The `nodes` and `links` parameters of the configuration file are intended to describe the method for constructing the [event search](#) results graph. The graph displays the relationships between event parameters whose nodes are defined in the `nodes` parameter and whose arcs are defined in the `links` parameter.

The configuration file contains the following settings:

- `timestamp_field`: the name of the setting for indicating the date and time in events that CEF Connector receives from an external system.
- `timestamp_scale`: the unit of time for events.
- `sep`: separator between the parameters of values in events received by the CEF Connector.
- `sep_kv`: separator between the key and value in events received by the CEF Connector.

- `sep_cef_caption`: separator in the header of events received by the CEF Connector.
- `mapping_fields`: rules for mapping event parameters received by the CEF Connector to the names of event parameters to be processed in Kaspersky MLAD. If necessary, you can specify the conditions for writing event parameters in Kaspersky MLAD depending on the values of other parameters received by the CEF Connector. This parameter is optional.
- `fields`: list of event parameters processed by the Event Processor service. The names of these parameters may coincide with the names of parameters received in CEF format, or may coincide with the names of parameters defined in the rules using the `mapping_fields` parameter.
- `nodes`: the group of settings that describe event parameters relationship graph nodes by using the following settings:
 - `name`: the name of the event parameter corresponding to the graph node.
 - `depth`: the order (left to right) of displaying the graph node in [event history](#).
 - `tooltip`: enables `templates`. `templates`: defines the tooltip displayed when you hover over the node.
 - `fieldShortCut`: defines an alternate name for the event parameter. The event parameters relationship graph displays the alternate name in brackets next to the value of the parameter corresponding to the graph node. This parameter is optional.
- `links`: a group of settings that describe graph arcs (event parameters relationships) by using the following settings:
 - `source`: the name of the first event parameter that creates a link on the graph.
 - `target`: the name of the second event parameter that creates a link on the graph.
 - `tooltip`: enables `templates`. The `templates` setting defines the tooltip displayed when you hover over the node. You can use the following variables with double curly braces:
 - Any parameter in the `fields` event parameter list.
 - `onIntervalActivationsCount`: the number of event detections in the event stream during the period defined when viewing the events history.
 - `onIntervalLastActivationTimestamp`: the date and time when the event was last detected in the event stream for the period defined when viewing the events history.
 - `lastActivationTimestamp`: the date and time when the event was last detected in the event stream.
 - `totalActivationsCount`: the number of event detections in the event stream.
 - `isGraphGroup`: defines how to display a connection on the event parameters relationship graph. If this parameter is set to `true`, events with different values of the parameters that are not used as the graph nodes are displayed as one event group. If the parameter is set to `false`, events with different values of the parameters are displayed as different events. The default value of this parameter is `false`.

Below is an example of a JSON file containing a configuration for the Event Processor service. The file contains a description of the event parameters for the Event Processor. According to the values specified in the `mapping_fields` parameter, events with the following event parameters will be displayed in Kaspersky MLAD:

- `EventType`: corresponds to the `cat` parameter in an event received by the CEF Connector.

- User_Name: corresponds to the cs1 parameter if the value user is received for the cs1Label parameter.
- Destination_Host: corresponds to the cs1 parameter if the value destination is received for the cs1Label parameter.
- Access_Result: corresponds to the cs1 parameter if the value access is received for the cs1Label parameter.

```
{
  "timestamp_field": "TimeStamp",
  "timestamp_scale": "ms",
  "sep": " ",
  "sep_kv": "=",
  "sep_cef_caption": "|",
  "mapping_fields": {
    "cat": "User_Host",
    "cs1": {"map_label": "cs1Label", "values": {"user": "User_Name",
    "destination": "Destination_Host",
    "access": "Access_Result"}}
  },
  "fields": [
    "User_Host",
    "User_Name",
    "Destination_Host",
    "Access_Result"
  ],
  "nodes": [
    {
      "name": "User_Name",
      "depth": 0,
      "tooltip": {
        "templates": [
          "User: {{User_Name}}"
        ]
      },
      "fieldShortCut": "User"
    },
    {
      "name": "User_Host",
      "depth": 1,
      "tooltip": {
        "templates": [
          "User host: {{User_Host}}"
        ]
      },
      "fieldShortCut": "Src"
    },
    {
      "name": "Destination_Host",
      "depth": 2,
      "tooltip": {
        "templates": [
          "Destination: {{Destination_Host}}"
        ]
      },
      "fieldShortCut": "Dst"
    }
  ],
  "links": [
    {
      "source": "User_Name",
```



```

"target": "User_Host",
"tooltip": {
  "templates": [
    "{{User_Name}} » {{User_Host}}",
    "Count: {{onIntervalActivationsCount}}"
  ]
},
"isGraphGroup": true
}, {
  "source": "User_Host",
  "target": "Destination_Host",
  "tooltip": {
    "templates": [
      "{{User_Host}} » {{Destination_Host}}",
      "DeviceEventClassID: {{Access_Result}}",
      "Count: {{onIntervalActivationsCount}}"
    ]
  }
}
]
}

```

Viewing the Kaspersky MLAD log

Kaspersky MLAD uses the Grafana logging system to monitor the state of application services and to track information security events.

Tracking information security events of Kaspersky MLAD in the logging subsystem

The table below shows the types of information security events that are tracked in Kaspersky MLAD.

Types of information security events

Information security event ID in the logging system	Information security event type
login	Connecting and attempting to connect users to Kaspersky MLAD
access_control	Verifying user rights when performing actions in the Kaspersky MLAD web interface
logout	Terminating a Kaspersky MLAD user connection
service_control	Starting, stopping, and restarting Kaspersky MLAD services
user_control	Editing user accounts
system_settings_control	Changing Kaspersky MLAD settings
model_control	Creating, modifying, and deleting models
tag_control	Importing, creating, modifying, and deleting tags
log_control	Deleting information security event logs from the Kaspersky MLAD

Each entry about an information security event contains the following parameters:

- **event_id** is the ID of the information security event.
- **timestamp** is the date and time of the information security event.
- **event_type** is the ID of the information security event type.
- **sub_type** specifies the type of information security event.
- **severity** is the importance of the information security event. Kaspersky MLAD provides the following severity levels for information security events:
 - **1 (low).**
These information security events include entries involving users being granted access to perform a specific action in the web interface, and regarding the successful completion of any user actions.
 - **5 (medium).**
These information security events include entries involving user actions in the web interface for managing ML models, tags, user accounts and passwords, and entries regarding exceeded thresholds for storage time and volume of information security event logs.
 - **8 (high).**
These information security events include entries involving users entering an incorrect user name and/or password when connecting to the web interface of the application, and entries regarding unsuccessful attempts to change a password.
 - **10 (highest).**
These information security events include entries involving attempts to connect to the application web interface using a system account or a blocked account, and entries regarding attempts to perform specific actions in the application without the appropriate access rights.
- **username** is the name of the user whose actions resulted in the information security event entry.
- **ip_address** is the IP address of the computer from which the user performed the action logged into the information security event log.
- **outcome** is the result of an information security event. The OK result corresponds to successful completion of the operation by the user. The FAIL result corresponds to failure of the user to perform the operation.
- **msg** is a brief summary of the information security event.
- **info** is a detailed description of the information security event.

Tracking the state of Kaspersky MLAD services in the logging subsystem

Kaspersky MLAD services whose states are monitored in the logging subsystem are identified based on the names of their corresponding containers or images in Docker. In most cases, the abbreviated name of the service is used as the name of the image. The container name is formed according to the following template:

`< application directory >-< image name >-#`,

where # is the number of the Docker container.

By default, Kaspersky MLAD uses the `mlad-release-5.0.0-<installation build number>` directory.

The Kaspersky MLAD log stores entries about the state of application services only for the last 48 hours.

The table below presents the correspondence between Kaspersky MLAD services and the names of Docker containers and images.

Correspondence between Kaspersky MLAD services and the names of Docker containers and images

Kaspersky MLAD service	Image name	Container name
Anomaly Detector	anomaly_detector	mlad-release-5.0.0-<installation build number>-anomaly_detector-1
Time Series Database	influxdb	mlad-release-5.0.0-<installation build number>-influxdb-1
Message Broker	kafka	mlad-release-5.0.0-<installation build number>-kafka-1
Keeper	keeper	mlad-release-5.0.0-<installation build number>-keeper-1
Logger	logger	mlad-release-5.0.0-<installation build number>-logger-1
Database	postgres	mlad-release-5.0.0-<installation build number>-postgres-1
Similar Anomaly	similar_anomaly	mlad-release-5.0.0-<installation build number>-similar_anomaly-1
Event Processor	event-processor	mlad-release-5.0.0-<installation build number>-event-processor-1
Stream Processor	stream-processor	mlad-release-5.0.0-<installation build number>-stream-processor-1
Trainer	trainer	mlad-release-5.0.0-<installation build number>-trainer-1
Web Server	nginx-ui	mlad-release-5.0.0-<installation build number>-nginx-ui-1
API Server	web-server	mlad-release-5.0.0-<installation build number>-web-server-1
Mail Notifier	postman	mlad-release-5.0.0-<installation build number>-postman-1
OPC UA Connector	opcua-connector	mlad-release-5.0.0-<installation build number>-opcua-connector-1
MQTT Connector	mqtt-connector	mlad-release-5.0.0-<installation build number>-mqtt-connector-1
AMQP Connector	amqp-connector	mlad-release-5.0.0-<installation build number>-amqp-connector-1
HTTP Connector	http-connector	mlad-release-5.0.0-<installation build number>-http-connector-1
KICS Connector	kics3-connector	mlad-release-5.0.0-<installation build number>-kics3-connector-1
CEF Connector	cef-connector	mlad-release-5.0.0-<installation build number>-cef-connector-1
WebSocket Connector	ws-connector	mlad-release-5.0.0-<installation build number>-ws-connector-1
Docker API Server	docker-api-server	mlad-release-5.0.0-<installation build number>-docker-api-

		server-1
Migrations	migrations	mlad-release-5.0.0-<installation build number>-migrations-1
Push Server	Push server	mlad-release-5.0.0-<installation build number>-push-server-1
	webstatic	mlad-release-5.0.0-<installation build number>-webstatic-1

The Info logging level is used for the Time Series Database, Message Broker, Logger, Database and Web Server services, and for webstatic image. The logging levels for all other Kaspersky MLAD services are defined by the system administrator when [configuring the application settings](#).

Scenario: viewing information security event logs


Before starting to work with the logging subsystem, it is recommended to read the [Grafana User Guide](#).

The maximum volume and storage time for information security event entries are defined when [configuring the security settings](#).

Information security event logs are written to the Kaspersky MLAD database automatically. If necessary, the system administrator can [specify the settings of an external system](#) to which the information security event logs should be sent.

The scenario for viewing information security event logs consists of the following steps:

1 Navigating to the logging subsystem


Go to the logging system by clicking the  button. This opens the Grafana interface in which you need to enter the name and password of the Kaspersky MLAD user.

Available only to the system administrators and users with the [Manage application logs](#) permission.


2 Navigating to the section containing information security event logs

Go to the **Security audit** section.

3 Analyzing information security event logs

Analyze the information security event log entries for the selected period. You can filter them based on parameters of the information security event logs. To do so, click the  button in the column containing the relevant log parameter, select the check boxes next to the necessary filtering criteria, and click **OK**. To reset the filtering criteria, clear the relevant check boxes and click **OK**.

4 Exporting information security event logs

To export the information security event logs for the selected period to a text file, under **Security audit**, choose **Inspect** → **Data** from the vertical menu  in the upper right corner of the information security event log table, and in the panel that opens, click **Download CSV**.

Scenario: assessing the main metrics of Kaspersky MLAD


Before starting to work with the logging subsystem, it is recommended to read the [Grafana User Guide](#).

When connecting to the logging subsystem for the first time, you must change the default password.

This subsection provides a sequence of actions that must be performed to assess the health and general state of Kaspersky MLAD.

The scenario for assessing the health and general state of Kaspersky MLAD consists of the following steps:

1 Navigating to the logging subsystem

Go to the logging system by clicking the  button. This opens the Grafana interface in which you need to enter the name and password of the Kaspersky MLAD user.

Available only to the system administrators and users with the [Manage application logs](#) permission.

2 Analyzing the main metrics of Kaspersky MLAD

In the **Summary docker metrics** section, analyze the graphs of the main Kaspersky MLAD metrics for the selected period.

The following metrics are displayed for each container of Kaspersky MLAD services:

- *CPU usage* – history of central processor workload caused by the container. This is measured as a percentage.
- *RAM usage* – history of the container's RAM usage. This is measured in bytes.
- *Disk usage Read/Write* – history of the container's load on the disk subsystem (read/write operations). This is measured in bytes.
- *Network usage* – history of the container's use of network resources. This is measured in bytes per second.

Scenario: viewing container logs and metrics


Before starting to work with the logging subsystem, it is recommended to read the [Grafana User Guide](#).

The Kaspersky MLAD log stores entries only for the last 48 hours.

This subsection provides steps for assessing the performance and viewing the logs of a specific container from the Kaspersky MLAD distribution kit.

The scenario for assessing the performance and viewing the logs of a specific container consists of the following steps:

1 Navigating to the logging subsystem

Go to the logging system by clicking the  button. This opens the Grafana interface in which you need to enter the name and password of the Kaspersky MLAD user.

2 Navigating to the section with container logs and metrics

Go to the **Service detailed monitoring** section and select the relevant container from the **Container** drop-down list.

3 Analyzing container metrics

In the **Service detailed monitoring** section, analyze the graphs of Kaspersky MLAD metrics for the selected container during the relevant period.


The **Service detailed monitoring** section provides the following metrics:

- *Memory* – history of the container's RAM usage. This is measured in bytes.
- *CPU* – history of central processor workload caused by the container. This is measured as a percentage.
- *File system* – history of the container's load on the disk subsystem (read/write operations). This is measured in bytes.
- *Network* – history of the container's use of network resources. This is measured in bytes per second.

4 Analyzing container metrics

Analyze the container log records for the selected period, which are displayed under the metrics dashboard. You can search the container log records. To do so, enter a search query in the **Log search** field and press the **ENTER** key. To reset the search results, clear the **Log search** field and press the **ENTER** key.

5 Exporting container logs

To export container logs for the selected period to a text file, under **Service detailed monitoring**, choose **Inspect** → **Data** from the vertical menu  in the upper right corner of the relevant metric section, and in the panel that opens, click **Download CSV**.

Special characters of regular expressions

You can use regular expressions to search for events, patterns and values of event parameters in the **Event Processor** section. Kaspersky MLAD supports use of the following special characters in regular expressions:

- **^** – Corresponds to the start of the parameter value. For example, **^A** means that the event parameter search will look for values beginning with the letter A.
- **\$** – Corresponds to the end of the parameter value. For example, **A\$** means that the event parameter search will look for values ending with the letter A.
- **.** – Corresponds to any single character.
- **|** – Splits permissible options for characters or a set of characters in a parameter value. For example, **c(o|a)t** matches both the **cot** and **cat** values.
- **** – Indicates that the next character is an ordinary character (not a special character) in the parameter value. You can use the **** character to search for special characters in a parameter value. For example, **\.** describes a dot in the parameter value, while **** describes a backslash.

- `[]` – Corresponds to any character from the set of permissible characters. For example, `[abc]` matches the occurrence of any one of the three specified characters.

To search for a range of values, you can use the `-` character. To find the characters that are not within the specified range, you can use the `^` character in the square brackets. For example, `[^0-9]` means any character except numerals can be present.

You can use the following special characters to indicate the necessary number of repetitions of an expression in the values of event parameters:

- `?` – Character indicating that the preceding expression may occur zero or one time in a parameter value.
- `*` – Character indicating that the preceding expression may occur zero or more times in a parameter value.
- `+` – Character indicating that the preceding expression may occur one or more times in a parameter value.
- `{ }` – Character class that lets you indicate the necessary number of repetitions of the preceding expression. You can specify the repetition count in one of the following ways:
 - `{n}` – The expression preceding the curly brackets occurs in the parameter value exactly `n` times.
 - `{m,n}` – The expression preceding the curly brackets occurs in the parameter value from `m` to `n` times inclusive.
 - `{m, }` – The expression preceding the curly brackets occurs in the parameter value at least `m` times.
 - `{ ,n}` – The expression preceding the curly brackets occurs in the parameter value no more than `n` times.

You can also use parentheses `()` to group elements of an expression. For example, `(c[oa]t){2}` matches `cotcot`, `catcat`, `cotcat`, and `catcot`.

Cipher suites for secure TLS connection

It is recommended to use the following cipher suite for a secure TLS connection via the TLS-1.2 protocol:

- `TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384;`
- `TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256;`
- `TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256;`
- `TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384;`
- `TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256;`
- `TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256;`
- `TLS_DHE_RSA_WITH_AES_256_GCM_SHA384.`

It is recommended to use the following cipher suite for a secure TLS connection via the TLS-1.3 protocol:

- `TLS_AES_128_GCM_SHA256;`
- `TLS_AES_256_GCM_SHA384;`

- TLS_CHACHA20_POLY1305_SHA256;
- TLS_AES_128_CCM_SHA256.

Glossary

Account role

Set of access rights that determine the actions available to a user when connected to the application web interface. Kaspersky MLAD includes a system administrator role and custom roles.

AMQP topic

A hierarchical path to the data source used for sending messages via the AMQP protocol.

Anomaly

Any deviation in the behavior of a monitored asset that is abnormal, unexpected, and not otherwise prescribed by the industrial process.

Artifact

A sequence of numerical values (time series) generated as a result of ML model inference. An ML model can generate artifacts associated with tag values received from the monitored asset or ML model element artifacts.

Asset

A section of a hierarchical structure representing, for example, a plant, a shop, or a separate unit of a monitored asset.

Attention

A special event processor configuration intended for tracking events and patterns for specific subsets of event history, and detecting commonalities in the behavior of the monitored asset.

Connector

Service that facilitates the exchange of data with external systems.

Data sampling

A method for adjusting the training set with reference to the time scale steps in the original dataset.

Event

A set of values taken from a predetermined list of parameters and indicating what happened on a monitored asset at a given moment

Gradient boosting

Machine learning technique for classification and regression problems that builds a prediction model in the form of an ensemble of prediction models, which are typically decision trees (XGBoost).

Graphic area

A collection of tags whose data is displayed together by overlapping on a single graph in **History** and **Monitoring** sections. A graphic area can display data for one or more tags in a preset.

ICS

Abbreviation for Industrial Control System. A package of hardware and software designed to automate control of process equipment at industrial enterprises.

Incident

An identified deviation from the expected (normal) behavior of a monitored asset.

Inference

The ML model works with telemetry data to detect anomalous behavior.

Inference indicator

A set of criteria used to determine the data time intervals on which the ML model performs the inference.

Learning indicator

A set of criteria used to determine the data time intervals on which the ML model performs the training.

Markup

Tool for selecting time intervals. Markups are used to generate learning indicators and inference of the ML model. A markup may utilize two types of criteria: conditions on the behavior of specific tags (time intervals are selected where these conditions are met) and a time filter (time intervals are selected independently of tag behavior).

ML model

Algorithm based on machine learning methods tasked with analyzing the telemetry of the monitored asset and detecting anomalies.

Monitor

Source of notifications about patterns, events, or values of event parameters detected by the Event Processor according to the defined monitoring criteria. The monitoring criteria define the attention head, additional filters for event parameter values, a sliding time window, and the number of consecutive monitor activations within that window.

Monitored asset hierarchical structure

A tree-like representation of the monitored asset where the leaf nodes represent tags associated with incoming telemetry data.

MQTT topic

A hierarchical path to the data source used for sending messages via the MQTT protocol.

Notification

A message with information about an incident (or incidents), which is sent by the application via notification delivery systems (for example, via email) to the specified addresses.

Pattern

Sequence of events or other patterns identified within the stream of events from the monitored asset.

Preset

Set of tags generated by a user in arbitrary order or created automatically when an incident is registered. A set of tags in a custom preset can correspond to a certain aspect of the technological process or a section of the monitored asset.

Tag

Variable that contains the value of a specific process parameter such as temperature.

Top tag

The process parameter that had the greatest impact on incident registration.

Uniform temporal grid (UTG)

An infinite sequence of points in time separated by equal intervals, to which the stream of incoming telemetry data is converted.

Information about third-party code

Information about third-party code is contained in the file `legal_notices.txt` located in the application installation directory (in the 'legal' subdirectory).

Trademark notices

Registered trademarks and service marks are the property of their respective owners.

Ubuntu and LTS are registered trademarks of Canonical Ltd.

The Grafana Word Mark and Grafana Logo are either registered trademarks/service marks or trademarks/service marks of Coding Instinct AB, in the United States and other countries and are used with Coding Instinct's permission. We are not affiliated with, endorsed or sponsored by Coding Instinct, or the Grafana community.

Docker and the Docker logo are trademarks or registered trademarks of Docker, Inc. in the United States and/or other countries. Docker, Inc. and other parties may also have trademark rights in other terms used herein.

Google, Chrome, Google Chrome are trademarks of Google LLC.

TensorFlow and any related marks are trademarks of Google LLC.

Intel, Core, and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Microsoft and Excel are trademarks of the Microsoft group of companies.

OpenSSL is a trademark of the OpenSSL Software Foundation.

Python is a trademark or registered trademark of the Python Software Foundation.

PGP is a trademark or registered trademark of Symantec Corporation or its affiliates in the U.S. and other countries.