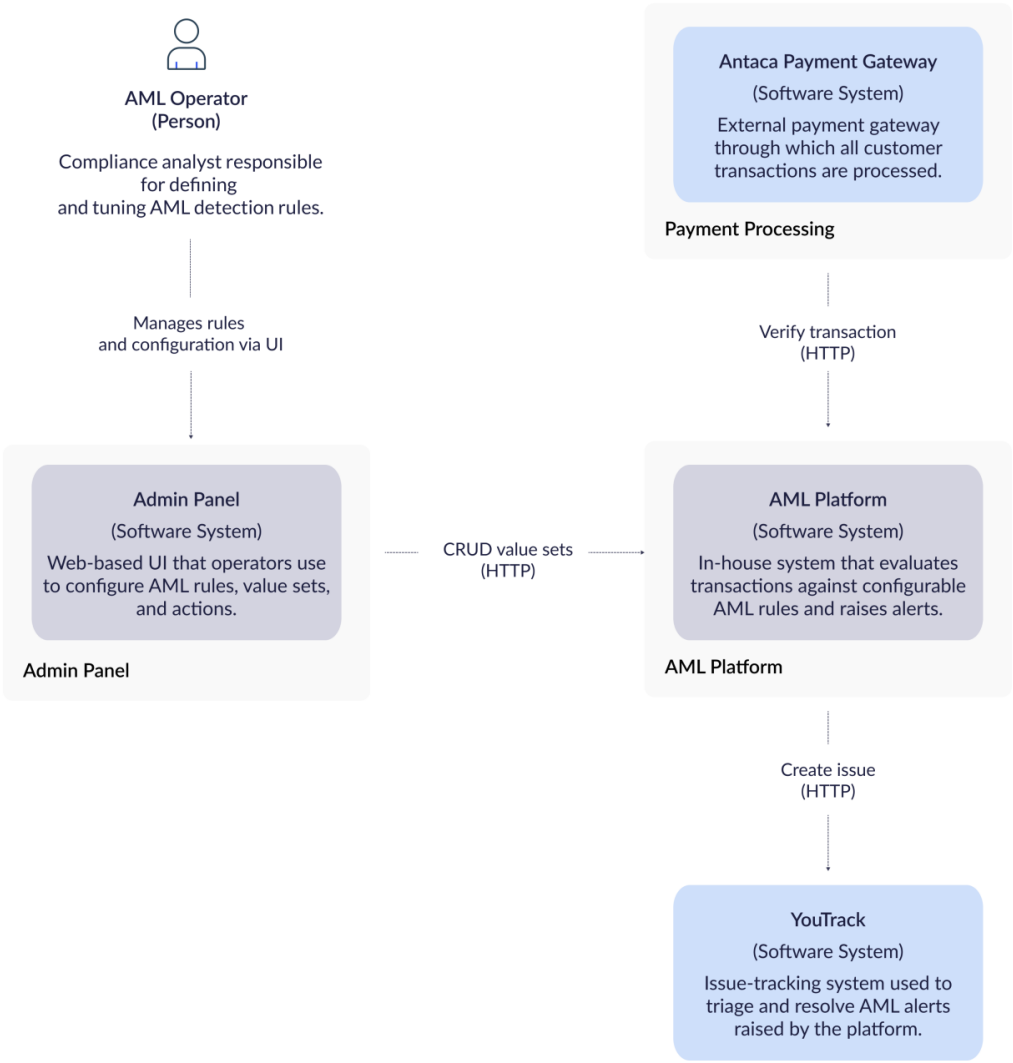
Overview

This document provides a high-level overview of the functionalities offered by the AML Transaction Monitoring service.
AML Transaction Monitoring allows you to define rulesets that process transactions in real time.
This document explains in detail how to construct AML rulesets and describes the available conditions that can be used within those rulesets.
Other configurable system elements required for a complete AML Transaction Monitoring setup are also described.

Architecture

C4 Context diagram



Ruleset configuration elements

Elements used to configure the logic for transaction processing are:

Element name	Element description
Ruleset	A set of conditions evaluated during transaction processing, and instructions that the system considers when returning the transaction outcome.
Action	Returned by the engine when processing a transaction; actions are interpreted and handled by the service requesting AML verification.

Value set	Helper variables that let you define sets of data, such as a list of high-risk countries.

The following sections describe each of these elements in detail.

Ruleset

A ruleset defines a set of conditions to check during transaction monitoring and the outcomes to apply when a transaction matches those conditions.
When a transaction is processed, all enabled rulesets are applied to see if the transaction matches their criteria.

Each ruleset consists of:

- a ruleset name,
- logic conditions checked by the processing engine,
- a trigger that defines the outcome when the ruleset is matched.

Condition types

Conditions define what the ruleset checks when the AML engine processes a transaction. Each ruleset can contain one or more conditions, combined with logical operators (`AND` / `OR`). The engine evaluates these conditions and returns a single decision for the transaction. The following types of conditions are supported:

Condition name	Condition description
AND / OR	Group multiple conditions: `AND` means all conditions must be true, `OR` means at least one must be true.
request_property_check	Compares a specific field from the transaction request (for example, amount, currency or transaction country) to a value or list of values.
kyc_property_check	Compares a property from the end user's KYC record, such as risk level or nationality, to a value or list of values.
transaction_volume_check	Evaluates the cumulative transaction volume over a defined period and scope (e.g. by end user, card, balance or corporation) and can be grouped by merchant or country.
transaction_quantity_check	Evaluates the number of transactions over a defined period and scope, also grouped by merchant or country.
blacklist_check	Checks whether certain end user or transaction values (e.g. PESEL, IBAN, name) match entries in a predefined blacklist.

greylist_check	Similar to `blacklist_check`, but matches data against a greylist of entities that are considered suspicious but not definitively fraudulent.
compare_with_last_transaction	Compares fields from the current transaction with those from the last transaction within a specified time window (e.g. 5 minutes) and context (card, balance or balance owner).

Each condition uses comparators (e.g. `=`, `!=`, `>`, `>=`, `<`, `<=`, `IN`, `NIN`, `CONTAINS`) to compare values. You can also refer to predefined *value sets* when supplying a list of values (for example, a set of high-risk countries).

> **Info:** We continuously develop new condition types to address all business needs related to AML transaction monitoring.

> **Tip:** In order to set up AML rulesets, check out **Technical Documentation – AML Ruleset Definition Language**

Trigger types

A trigger defines the outcome when a ruleset is matched during transaction processing.
You can combine actions and alerts with a decision in a single trigger block.
There are three trigger types:

Trigger type	Trigger description
Decision (required)	The main result of single ruleset processing. You set it to one of: `APPROVED`, `DECLINED`, or `ON_HOLD`. Only one decision is returned for the whole transaction, even if multiple rulesets match.
Actions (optional)	Actions are instructions returned by the engine and executed by the requesting system. Example action: block an end user. Actions must be defined in advance in your system or via API.
Alert (optional)	Alerts are notifications sent to selected channels (for example: `YouTrack ticket`) when a ruleset is matched. You can use alerts to inform AML officers about suspicious activity.

Action

An action is an instruction returned by the AML engine when a ruleset is triggered.
Actions define additional operations that should be performed by the system requesting AML

verification.

Examples of actions:

- Block an end user
- Block an account

Actions are optional and must be defined in advance via GUI or via API.

> **Warning:** The AML system only allows you to define and return actions when processing a transaction. The logic for executing each action is the responsibility of the system that receives the action from the AML engine.

> **Warning:** Before using actions in AML rulesets, make sure they are defined in the system.

Value set

A value set is a helper list you can use in AML rulesets to simplify configuration and maintenance. Value sets let you store reusable sets of values, such as lists of high-risk countries, transaction types, or merchants.
You can reference a value set in any ruleset condition, instead of listing all the values every time.

Examples of value sets:

- List of high-risk countries
- List of blocked merchants
- List of suspicious transaction types

Value sets are optional. You define them in the administration panel or via API, and update them as your risk policies change.

> **Warning:** Before using value set in AML rulesets, make sure they are defined in the system.

Watchlist management

The AML system supports two types of watchlists for identifying suspicious end users.

1. **Blacklist** – designed to store end users confirmed or proven to be involved in fraudulent activity.
2. **Greylist** – designed to store end users suspected of fraudulent behavior but not yet confirmed.

Managing watchlists involves adding end users by their personal and address data. End users can also be removed from a list at any time.
Entries on these lists are automatically verified when the `blacklist_check` or `greylist_check` condition is used in AML rulesets.

Purpose and benefits

1. **Cross-instance fraud detection** - with properly defined AML rulesets, the system can identify and block a known fraudulent end user across all BIN sponsor instances. This is possible because verification relies on personal and address data instead of instance-specific end user identifiers.
2. **Monitoring suspicious end users** - using greylists, AML operators can add end users suspected of fraudulent activity and monitor their presence across different instances. Configured AML rulesets can trigger an alert when a transaction involves an end user found on the greylist.

Alerts

When defining an AML ruleset, the operator configures how alerts related to that ruleset should be sent. The system sends an alert according to the defined alert configuration when a transaction matches the AML ruleset.

Alert configuration

Each AML ruleset can define how alerts are sent when triggered.
The alert configuration includes following key parameters:

Alert parameter	Alert parameter description
channel	Defines where the alert should be sent. Currently, the supported channel is `YOUTRACK_TICKET`, which creates a new issue in the predefined YouTrack project.
cooldown_period	Defines the minimum time (in seconds) between two identical alerts generated by the same ruleset for the same end user or corporation. This prevents multiple alerts of the same type from being created within a short time window.

These parameters are defined inside the `alert` block of the AML ruleset configuration.

> **Info:** We plan to extend the alerting mechanism in the future by adding new alert channels, such as email notifications sent to the person/people responsible for manual AML case review when required.

AI risk scoring

Each verified transaction is also analyzed by a machine learning–based system that calculates the probability of fraud.
The result is a numerical score ranging from **0 to 100.**

- **0** means the transaction is considered completely safe
- **100** represents the highest possible fraud probability assigned by the system

> **Warning:** At this stage, the AI score is not used in any way when making AML decisions about a transaction.
> All decisions are made solely based on the outcomes of individual AML rulesets that process the transaction.

Transaction evaluation process

To enable the system to check transactions against AML rulesets, the rulesets must first be defined.

This can be done through the graphical interface available in the **Administration Panel**.

Once the rulesets are configured, the transaction processing system must start calling the AML service using the `aml-verify` endpoint.
When the AML system receives a request on this endpoint, it begins evaluating the transaction against all rulesets defined in the system.

As a result of this evaluation, the system returns:

- **verificationId** – a unique identifier of the verification entity,
- **result** – a single decision for the transaction,
- **actions** – an array of actions returned by the matching rulesets.

The AML system always returns all actions triggered by the rulesets, but only one final decision.

> **Warning: decision return logic**
> If all rulesets return `APPROVED`, the final result is `APPROVED`.
> If at least one ruleset returns `DECLINED`, the final result is `DECLINED`.
> If no ruleset returns `DECLINED` but at least one returns `ON_HOLD`, the final result is `ON_HOLD`.

Revision #69
Created 22 September 2025 06:49:54
Updated 27 March 2026 13:40:11 by Kamil Szalast