

ИНТЕГРИРОВАННАЯ СИСТЕМА ОБНАРУЖЕНИЯ АНОМАЛИЙ СЕРВЕРА И IoT-УСТРОЙСТВ

В. Ю. Маханова¹, Т. А. Пулко²

¹Национальный детский технопарк, г. Минск, Республика Беларусь

²Белорусский государственный университет информатики и радиоэлектроники, г. Минск

Рассмотрена система обнаружения аномального поведения IoT-устройств и сервера (MQTT-брокер), разработанная с применением методов машинного обучения. Для детекции аномалий телеметрии, получаемой от сенсоров, используется модель K-Nearest Neighbors, для перехвата вредоносного сетевого трафика – Random Forest. Реализована единая архитектура сбора данных, обучения моделей и их интеграции в режиме реального времени.

Ключевые слова: IoT, MQTT, обнаружение аномалий, DDoS, KNN, машинное обучение, Random Forest.

INTEGRATED ANOMALIES DETECTED SYSTEM FOR SERVERS AND IoT-DEVICES

V. Y. Makhanava¹, T. A. Pulko²

¹National Children's Technopark, Belarus

²Belarusian state university of informatics and radioelectronics, Minsk

This work presents an end-to-end system for detecting anomalous behavior in IoT devices and DDoS/DoS attacks on an MQTT broker using machine learning. K-Nearest Neighbors is applied to telemetry from sensors, while a Random Forest ensemble identifies malicious network patterns. A unified architecture for data collection, model training, and real-time inference is implemented.

Keywords: IoT, MQTT, anomaly detection, DDoS, KNN, machine learning, Random Forest.

С учетом развития Интернета вещей (IoT) количество подключенных устройств экспоненциально растет, охватывая сферы умного дома, носимых гаджетов и промышленной автоматизации, что предоставляет возможности для оптимизации процессов и улучшения качества жизни, но одновременно порождает риски безопасности: уязвимости в прошивках, недостаточная аутентификация и целенаправленные атаки могут приводить к утечке данных и нарушению работы критически важных систем. Актуально создание решения, способного автоматически выявлять отклонения в поведении сенсоров и классифицировать DoS-атаку на MQTT-брокер с помощью моделей машинного обучения.

Была развернута система симуляции IoT-сети, для этого создан MQTT-брокер, который стал сервером связи между IoT-устройствами. Далее к нему подключили датчик температуры, контролируемый через микроконтроллер, который по заранее написанному скетчу на Arduino IDE поддерживает подключение к Wi-Fi, инициализирует датчик и публикует показания температуры в заданный топик каждые несколько секунд. Код предусматривает автоматическое восстановление соединений и

логирование результатов публикации. Для построения обучающего набора телеметрии был написан скрипт на Python с использованием библиотеки `paho-mqtt` [1]. Он подписывается на топик с данными о температуре, преобразует полученные сообщения в строки вида `[timestamp, temperature]` и сохраняет их в CSV-файл. Особое внимание уделялось равномерности временного шага и точности временных меток. Далее были добавлены метки «аномалия» для описания резких скачков и дрейфовых значений в поведении датчика.

Обучение модели KNN включало последовательные этапы: предварительная обработка данных, вычисление вторичных признаков (разности температур dT , временных интервалов dt , скользящего среднего и стандартного отклонения), масштабирование через `StandardScaler` и подбор параметров алгоритма `KNeighborsClassifier` с учетом дистанционного взвешивания соседей. Вторая ветвь системы предназначена для выявления DoS- и DDoS-атак на уровне брокера. Для этого использован публичный набор данных, полученный на физическом IoT-тестовом стенде с эмуляцией нормального MQTT-трафика и пяти типов атак. Каждому сетевому событию соответствует набор признаков: длина кадра, интервал между пакетами, уровень QoS, флаг Retain, длина топика и длина полезной нагрузки. На основе этих данных был обучен Random Forest-классификатор.

Интеграция обеих моделей осуществляется единым мониторинговым скриптом, который подписывается на все топики и одновременно поддерживает два буфера данных: один для пакетов телеметрии, другой – для сетевых фреймов. Внутри основного цикла периодически происходит очистка устаревших записей, формируются векторы признаков и выполняются запросы к предварительно загруженным моделям.

Обнаруженный инцидент классифицируется по четырем типам (рис. 1):

- тип 2 – подозрение на DoS/DDoS по превышению скорости пакетов или классификации Random Forest;
- тип 3 – аномалия телеметрии по результату KNN;
- тип 4 – значительное отклонение скользящих средних за заданный интервал;
- тип 1 – нормальный режим (уведомление не отправляется).

При детекции любая из ситуаций формирует уведомление в формате “<тип>, <дополнительная информация>” и публикует его в отдельный топик для визуализации в пользовательском приложении.

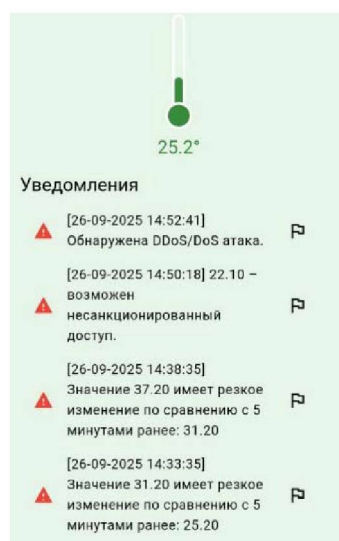


Рис. 1. Пример отслеживания уведомлений в приложении

В рамках исследования предложена система анализа поведения IoT-устройств и сетевого трафика на основе моделей K-NN и RF. Для этого был создан обучающий набор данных, подобраны нужные гиперпараметры моделей. Также предложено создание пользовательского приложения, в котором можно отслеживать уведомления об аномальном поведении устройств и сервера. Перспективы дальнейшей работы включают возможность добавления различных типов датчиков, а также автономное переобучение модели, способную подстраиваться под разные ситуации.

Л и т е р а т у р а

1. Dekun Tao MQTT in Python with Paho Client: Beginner's Guide 2025 – URL: <https://www.emqx.com/en/blog/how-to-use-mqtt-in-python> (дата обращения: 11.10.2025).