КОМПЬЮТЕРНЫЕ НАУКИ, ПРИБОСТРОЕНИЕ И АВТОМАТИЗАЦИЯ
КОМПЬЮТЕРЛІК ҒЫЛЫМДАР, АСПАП ЖАСАУ ЖӘНЕ АВТОМАТТАНДЫРУ
COMPUTER SCIENCE, INSTRUMENTATION AND AUTOMATION

**УДК 608.4**
**МРНТИ 81.93.29**
**https://doi.org/10.53364/24138614_2026_41_2_18**

**Е.Н.Турсынбек[1,2*], Н.Албанбай[1*], Ж.К.Кегенбеков[2], А.Аханкызы[1]**

[1]Казахский национальный исследовательский технический университет им.К.И.Сатбаева, Алматы, Казахстан
[2] Казахстанско-Немецкий Университет, Алматы, Казахстан

[*]*E-mail: y.tursynbek@satbayev.university*

# РАЗРАБОТКА СИСТЕМЫ IOTECTOR ДЛЯ МОНИТОРИНГА СЕТЕВОГО ТРАФИКА И ОБНАРУЖЕНИЯ КИБЕРАТАК В IOT-СЕТЯХ

***Аннотация.*** *Рост числа устройств Интернета вещей (IoT) сопровождается увеличением количества сетевых инцидентов и атак, что требует разработки средств защиты, способных работать в условиях ограниченных вычислительных ресурсов и высокой интенсивности обмена данными. В данной работе представлена платформа IoTector, предназначенная для мониторинга сетевого трафика и выявления кибератак в IoT-среде. Система реализована в виде промежуточного узла между IoT-устройствами и сетевым шлюзом, что позволяет анализировать трафик непосредственно на уровне локальной инфраструктуры. Для классификации сетевых событий использованы модели глубокого обучения DNN, CNN и CNN–BiLSTM. Экспериментальная часть выполнена на наборе данных CICIoT2023, а прототип платформы развернут на базе Raspberry Pi 5. Помимо модуля обнаружения атак разработан веб-интерфейс, обеспечивающий мониторинг сетевой активности, просмотр событий безопасности и управление процессом обучения моделей.*

*Проведённые эксперименты показали, что наиболее эффективной оказалась модель CNN, обеспечившая высокие показатели качества классификации при сохранении приемлемой вычислительной нагрузки. На основе этих результатов можно говорить о практической применимости IoTector — платформа способна непрерывно отслеживать трафик и своевременно фиксировать угрозы в IoT-сетях.*

***Ключевые слова:*** *Интернет вещей, IoT-безопасность, обнаружения вторжений, сетевой трафик, глубокое обучение, мониторинг IoT-сетей.*

**Введение.**

В настоящее время технологии Интернета вещей занимают всё более заметное место в цифровой инфраструктуре различных отраслей. Подключённые устройства используются в промышленности, транспортных системах, здравоохранении, энергетике и проектах умных городов. Расширение IoT-экосистемы сопровождается постоянным ростом объёмов передаваемых данных и увеличением числа взаимодействующих узлов.

Одновременно с этим возрастает количество угроз, направленных на IoT-инфраструктуру. Компрометация отдельных устройств может приводить не только к нарушению их функционирования, но и создавать риски для всей сети. Особую опасность представляют распределённые атаки отказа в обслуживании, сетевые аномалии и другие виды вредоносной активности, способные повлиять на доступность и устойчивость

сервисов. По этой причине задачи мониторинга сетевого трафика и своевременного обнаружения атак остаются одним из ключевых направлений исследований в области информационной безопасности IoT-систем [1].

Существующие системы обнаружения вторжений достаточно успешно применяются для защиты корпоративных и традиционных сетей, однако их использование в IoT-среде связано с рядом ограничений. Во многих случаях анализ трафика выполняется централизованно, что требует передачи значительных объёмов данных на удалённые вычислительные узлы. Такой подход увеличивает задержки обработки и создаёт дополнительную нагрузку на сеть. Дополнительную сложность представляет специфика самих IoT-устройств. Большинство из них обладают ограниченными вычислительными возможностями, объёмом памяти и энергетическими ресурсами, поэтому применение ресурсоёмких алгоритмов непосредственно на конечных узлах оказывается затруднительным. В результате задачи оперативного выявления атак и аномалий становятся особенно актуальными для динамически изменяющихся IoT-сетей, где требуется непрерывный анализ трафика и быстрое реагирование на потенциальные угрозы. Указанные особенности определяют необходимость разработки решений, способных выполнять интеллектуальную обработку сетевых данных непосредственно в инфраструктуре IoT, сохраняя баланс между качеством обнаружения атак и вычислительными затратами.

Значительное число современных исследований связано с использованием методов глубокого обучения для выявления атак в IoT-сетях. В работах [2–6] рассматриваются различные архитектуры нейронных сетей, предназначенные для анализа сетевого трафика и классификации вредоносной активности. В исследовании [2] предложена гибридная модель CNN–GRU, тогда как авторы работ [3–6] анализируют возможности других подходов глубокого обучения при решении задач обнаружения вторжений. Несмотря на различия в используемых архитектурах, результаты этих исследований подтверждают высокий потенциал методов глубокого обучения для повышения качества обнаружения атак. Вместе с тем основное внимание в большинстве работ уделено оценке моделей на наборах данных, тогда как вопросы их практического внедрения рассматриваются значительно реже.

Отдельный интерес представляют исследования, использующие более сложные способы представления сетевого взаимодействия, включая графовые модели и методы предварительного преобразования признаков [7, 8]. Такие подходы позволяют повысить качество обнаружения атак, но часто требуют больших вычислительных затрат. Для IoT-сред это становится существенным ограничением, поскольку вычислительные возможности конечных устройств и промежуточных узлов обычно ограничены.

Федеративное обучение [9–12] решает другую задачу — распределённое обучение моделей без передачи исходных данных на центральный сервер, что критично там, где конфиденциальность данных нельзя нарушать. Цена за это — усложнение архитектуры: обмен параметрами между узлами и необходимость их согласованной работы требуют дополнительных ресурсов.

В работах [13–18] предложены также более легковесные и практико-ориентированные подходы, лучше приспособленные к ограничениям IoT-устройств. Однако многие из них ориентированы на отдельные типы атак, конкретные наборы данных или узкие сценарии использования. Как показывают обзорные исследования [19], задача разработки решений, сочетающих приемлемую вычислительную сложность, высокое качество детекции и возможность практического внедрения, по-прежнему остаётся актуальной.

Проведённый анализ показывает, что существующие подходы обеспечивают высокий уровень качества обнаружения атак, однако вопросы их практического применения в реальных IoT-инфраструктурах остаются недостаточно изученными. Особый интерес

представляет возможность совмещения эффективных методов анализа трафика с требованиями к вычислительным ресурсам и непрерывному мониторингу сети. Это определяет актуальность разработки решений, ориентированных не только на точность классификации, но и на возможность эксплуатации в реальных условиях.

В отличие от существующих решений [1, 2, 11, 12], в основном ориентированных на программную оценку моделей на эталонных датасетах, в настоящей работе предложена и реализована комплексная система обнаружения кибератак в IoT-сетях на базе Raspberry Pi 5. Научная значимость работы определяется сочетанием нескольких компонентов в рамках единой платформы. Выполнено сравнительное исследование моделей DNN, CNN и CNN-BiLSTM на наборе данных CICIoT2023, реализован веб-интерфейс для мониторинга состояния сети, управления устройствами и регистрации событий безопасности, а также предусмотрена возможность дальнейшего развития платформы в направлении федеративного обучения для распределённых IoT-инфраструктур. Экспериментальная оценка показала, что среднее время инференса составляет 107 мс, что позволяет использовать систему для анализа сетевого трафика в режиме, близком к реальному времени.

**Материалы и методы исследования.**

Для решения задачи мониторинга сетевого трафика и выявления кибератак была разработана платформа IoTector, функционирующая между IoT-устройствами и сетевым шлюзом. В качестве аппаратной основы использовался одноплатный компьютер Raspberry Pi 5, обеспечивающий возможность развёртывания системы непосредственно в локальной IoT-инфраструктуре. Подключение устройств осуществляется посредством беспроводных технологий, включая Wi-Fi и Bluetooth. Основной функцией платформы является анализ сетевого трафика с использованием моделей глубокого обучения для обнаружения атак и аномального поведения. Обработка данных выполняется в режиме, близком к реальному времени — благодаря этому угрозы выявляются оперативно, а события безопасности фиксируются без задержек. Для взаимодействия с системой разработан веб-интерфейс, предоставляющий средства мониторинга состояния сети, визуализации сетевой активности и регистрации обнаруженных инцидентов. Дополнительно реализованы функции управления устройствами, обновления моделей и запуска процедур обучения. Такое построение платформы позволяет объединить средства анализа трафика, мониторинга и администрирования в рамках единого программного комплекса.

В рамках данной работы необходимо различать реализованную часть прототипа IoTector и концептуальные архитектурные расширения платформы. Реализованная часть включает развертывание системы на базе Raspberry Pi 5, модуль анализа сетевого трафика с использованием моделей глубокого обучения, веб-интерфейс мониторинга, журналирование событий безопасности, управление IoT-устройствами и экспериментальную оценку моделей DNN, CNN и CNN-BiLSTM на наборе данных CICIoT2023.

Поддержка федеративного обучения в настоящей работе рассматривается как концептуальное архитектурное расширение IoTector для многозвенных IoT-инфраструктур. Данный режим предполагает наличие нескольких распределенных узлов IoTector, выполняющих локальное обучение моделей на собственных данных с последующей агрегацией параметров на центральном сервере. При этом исходные сетевые данные не передаются за пределы локальных узлов. Экспериментальная оценка федеративной конфигурации не входит в реализованную часть настоящего исследования и рассматривается как направление дальнейшей работы.

Для экспериментальной оценки использовался набор данных CICIoT2023, содержащий записи сетевого трафика IoT-устройств и различные категории атак. После очистки и предварительной обработки была сформирована выборка объёмом 50 000

записей. Исследование рассматривало задачу многоклассовой классификации по 12 классам, включающим нормальный трафик и различные типы атак. Каждая запись описывалась 39 признаками, предусмотренными структурой набора данных.

Подготовка данных включала удаление некорректных и неполных записей, преобразование признаков в формат, пригодный для обучения моделей, а также их нормализацию. Выполненная предобработка позволила уменьшить влияние различий в масштабах признаков и обеспечить стабильность процесса обучения. После завершения подготовки данные были разделены на обучающую и тестовую выборки в соотношении 80:20.

Для всех исследуемых моделей использовались одинаковые входные данные, единая схема разделения выборки и одинаковые параметры обучения, что обеспечивало корректность сравнительного сравнения результатов.

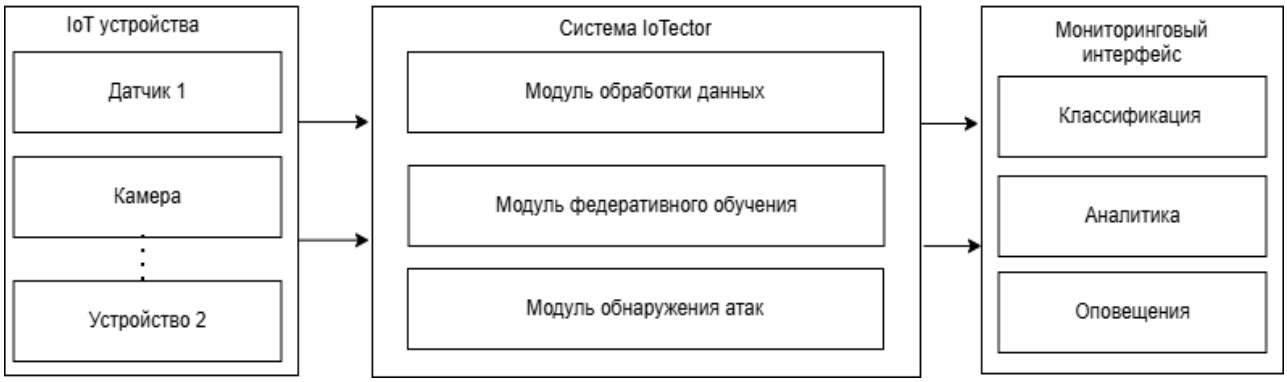


Рисунок 1 – Структурная схема системы IoTector для обнаружения кибератак в IoT-сетях.

Для исследования были выбраны три модели глубокого обучения — DNN, CNN и CNN–BiLSTM. Их обучение и последующая оценка выполнялись на одинаковых данных, прошедших одну и ту же процедуру предварительной обработки. Это позволило провести сопоставление моделей в равных условиях и объективно оценить особенности каждой архитектуры. Обучение всех моделей выполнялось с использованием оптимизатора Adam со скоростью обучения 0.001, размером батча 1024 и количеством эпох 10. Модель DNN представляла собой полносвязную нейронную сеть и использовалась как базовая архитектура для сравнения. Модель CNN применялась как сверточная архитектура для классификации признаков сетевого трафика. Гибридная модель CNN–BiLSTM объединяла сверточные слои и двунаправленный рекуррентный слой, что позволяло использовать её для многоклассовой классификации атак. Основные архитектурные параметры и гиперпараметры используемых моделей приведены в таблице 1.

Таблица 1 – Архитектуры моделей и гиперпараметры

| **Модель** | **Гиперпараметры** | **Значение** |
|---|---|---|
| **DNN** | Скрытые слои | 128, 64, 32 |
| | Функция активации | ReLU |
| | Выходная активация | Softmax |
| | Оптимизатор | Adam |
| | Функция потерь | Sparse categorical crossentropy |
| | Скорость обучения | 0.001 |
| | Размер входа | 39 |
| **CNN** | Сверточные слои (Conv1D) | 64, 32 |

| | | |
|---|---|---|
| | Размер ядра | 3 |
| | Dropout | 0.2, 0.3 |
| | Полносвязный слой | 128 |
| | Batch normalization | True |
| | Выходная активация | Softmax |
| | Оптимизатор | Adam |
| | Функция потерь | Sparse categorical crossentropy |
| | Скорость обучения | 0.001 |
| | Размер входа | 39 |
| **CNN–BiLSTM** | Изменение формы входа | (39, 1) |
| | Сверточные фильтры (Conv1D) | 32, 64 |
| | Размер ядра | 3 |
| | BiLSTM units | 64 |
| | Dropout (сверточные слои) | 0.2 |
| | Dropout (BiLSTM) | 0.2 |
| | Выходная активация | Softmax |
| | Оптимизатор | Adam |
| | Функция потерь | Sparse categorical crossentropy |
| | Скорость обучения | 0.001 |
| | Метрика | Accuracy |
| | Размер входа | 39 |

В качестве перспективного архитектурного расширения IoTector может быть использован в многозвенной конфигурации с поддержкой федеративного обучения, при котором несколько распределённых узлов IoTector выполняют локальное обучение моделей глубокого обучения на основе собственных данных. На каждом узле IoTector сетевой трафик проходит локальную подготовку. Она включает очистку данных, нормализацию признаков и формирование признакового пространства. Параметры предобработки могут различаться в зависимости от особенностей трафика и доступных вычислительных ресурсов конкретного узла.

После завершения локального обучения на центральный сервер передаются только параметры модели. Исходные данные при этом не покидают пределы локального узла. Для объединения результатов используется алгоритм FedAvg. Он агрегирует параметры локальных моделей и формирует глобальную модель, которая затем распространяется между узлами системы.

При такой схеме исходные данные не покидают локальный узел — это сохраняет конфиденциальность и одновременно сокращает объём сетевого обмена между участниками федеративной системы.

Глобальная модель на раунде $t + 1$ вычисляется следующим образом:

$$w^{(t+1)} = \sum_{k=1}^{K} \frac{n_k}{\sum_{j=1}^{K} n_j} w_k^{(t)} \quad (1)$$

где $w_k^{(t)}$ — параметры локальной модели $k$-го клиента после $t$ -го раунда обучения; $n_k$ — число локальных образцов на $k$-м устройстве; $K$ — количество узлов, участвующих в федеративном обучении.

Сформированная глобальная модель передаётся в модуль обнаружения вторжений. Далее она используется для анализа сетевого трафика в режиме реального времени. Для каждого входящего окна трафика $x_t$вычисляется вектор вероятностей классов

$$p_t = f_\theta(x_t) \quad (2)$$

где $f_\theta$ — обученная модель с параметрами $\theta$. Предсказанный класс определяется как

$$y_t = \arg max\,(p_t). \quad (3)$$

Если максимальное значение вероятности превышает заданный порог уверенности $\tau$, формируется событие безопасности, которое регистрируется в журнале и отображается в интерфейсе мониторинга. Пороговый механизм снижает число ложных срабатываний и делает результаты детекции более интерпретируемыми.

Для оценки качества классификации используются следующие метрики на основе матрицы ошибок (TP — истинно положительные, TN — истинно отрицательные, FP — ложно положительные, FN — ложно отрицательные):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (7)$$

В условиях многоклассовой классификации метрики вычисляются для каждого класса отдельно и усредняются по методу macro-averaging.

Разработка платформы выполнялась на языке Python 3.12. Серверная логика реализована средствами Django 5. Обмен данными в режиме реального времени обеспечивается с помощью Django Channels и протокола WebSocket (ASGI), тогда как функции сервера приложений выполняет Daphne. Для построения и обучения моделей глубокого обучения использовалась библиотека TensorFlow. В качестве системы хранения данных используется SQLite. Визуализация результатов мониторинга осуществляется с применением Bootstrap 5 и библиотеки Chart.js. Выбранные программные средства позволяют развернуть платформу в виде веб-ориентированной системы и обеспечить непрерывную обработку данных. Кроме того, используемый стек технологий поддерживает интеграцию моделей глубокого обучения в единое программное решение.

**Результаты и их обсуждение.**

Эффективность платформы IoTector оценивалась на основе сравнительного анализа моделей глубокого обучения, используемых для классификации сетевого трафика и обнаружения атак в IoT-сетях. Результаты по метрикам Accuracy, Precision, Recall и F1-score приведены в таблице 2.

Таблица 2 – Сравнение производительности моделей глубокого обучения

| Модель | Accuracy (%) | Precision (%) | Recall (%) | F1-score (%) |
|---|---|---|---|---|
| DNN | 94.95 | 96.09 | 94.95 | 94.45 |
| CNN | 95.27 | 96.07 | 95.27 | 94.92 |
| CNN+BiLSTM | 94.97 | 95.85 | 94.97 | 94.58 |

Преимущество CNN может быть связано со способностью сверточных слоев выделять устойчивые локальные зависимости между признаками сетевого трафика. При этом гибридная модель CNN-BiLSTM не продемонстрировала преимуществ по сравнению с CNN, что, вероятно, связано с особенностями набора данных CICIoT2023. Используемые данные представлены в виде агрегированных характеристик сетевых соединений, а не непрерывных временных последовательностей, где рекуррентный слой BiLSTM мог бы проявить себя более эффективно.

Измерение времени инференса проводилось на платформе Raspberry Pi 5 с использованием TensorFlow Lite. Для оценки производительности было выполнено 1000 последовательных запросов к модели CNN, которая показала лучшие результаты классификации среди исследованных архитектур. Среднее время инференса составило 107 мс на один образец при стандартном отклонении ±3,2 мс. Это соответствует обработке примерно девяти образцов сетевого трафика в секунду и позволяет использовать платформу для мониторинга IoT-сети в режиме, близком к реальному времени.

В рамках разработки платформы IoTector реализован веб-интерфейс мониторинга, обеспечивающий визуализацию состояния IoT-сети и событий безопасности в режиме реального времени. Интерфейс объединяет четыре функциональных модуля, каждый из которых решает самостоятельную задачу в рамках общего процесса мониторинга и управления безопасностью IoT-инфраструктуры.

Панель мониторинга, представленная на рисунке 2, отображает распределение типов сетевого трафика по категориям — нормальный трафик и различные классы атак, а также динамику событий безопасности во времени. Благодаря такому представлению оператор быстрее замечает аномальные всплески активности, оценивает соотношение легитимного и вредоносного трафика и отслеживает общее состояние безопасности сети в реальном времени.

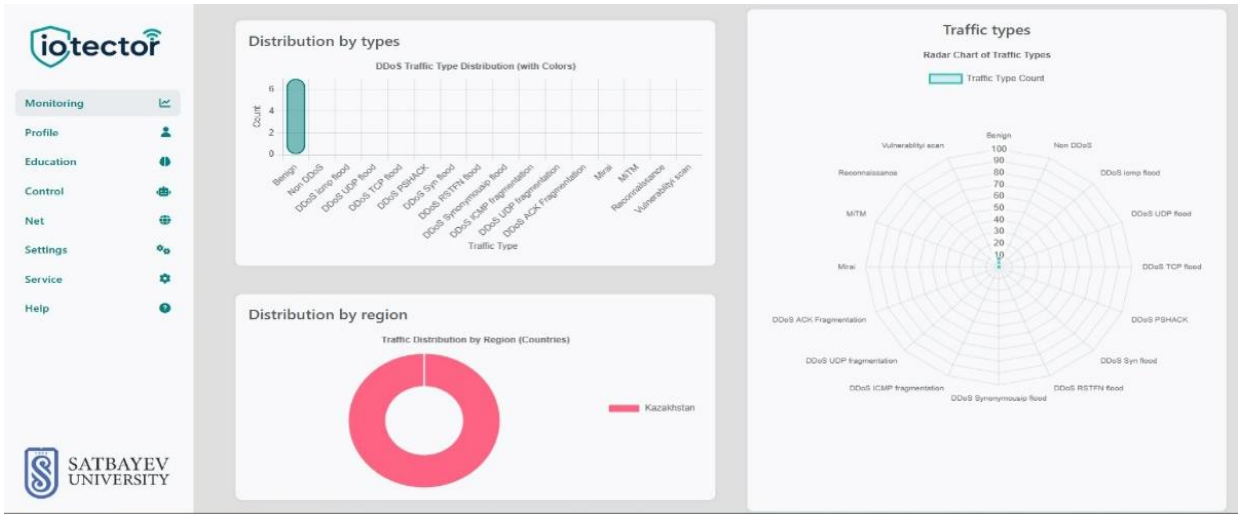


Рисунок 2 – Панель мониторинга IoTector с визуализацией распределения сетевого трафика

Модуль регистрации событий безопасности (рисунок 3) предназначен для ведения журнала событий на уровне отдельных устройств. Для каждого обнаруженного события сохраняются идентификатор устройства, временная метка, предсказанный класс трафика и уровень уверенности модели. Пороговое значение уверенности помогает отделять наиболее вероятные атаки от спорных классификаций. Это позволяет уменьшить количество ложных срабатываний и облегчает анализ результатов обнаружения.

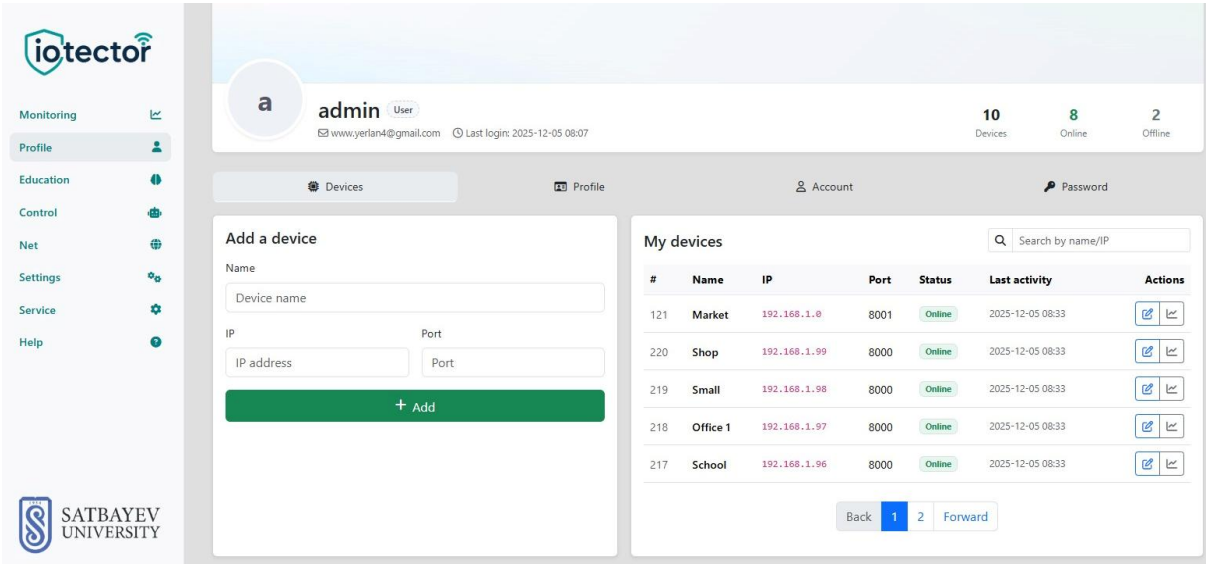


Рисунок 3 – Интерфейс мониторинга устройств и журнал событий безопасности

Модуль управления устройствами (рисунок 4) используется для централизованной регистрации IoT-узлов, отслеживания их текущего состояния и контроля сетевой активности. Система позволяет определить, находится ли устройство в активном состоянии, временно недоступно или отключено. Такая функция особенно важна для динамических IoT-сред, где состав подключённых устройств может изменяться со временем.

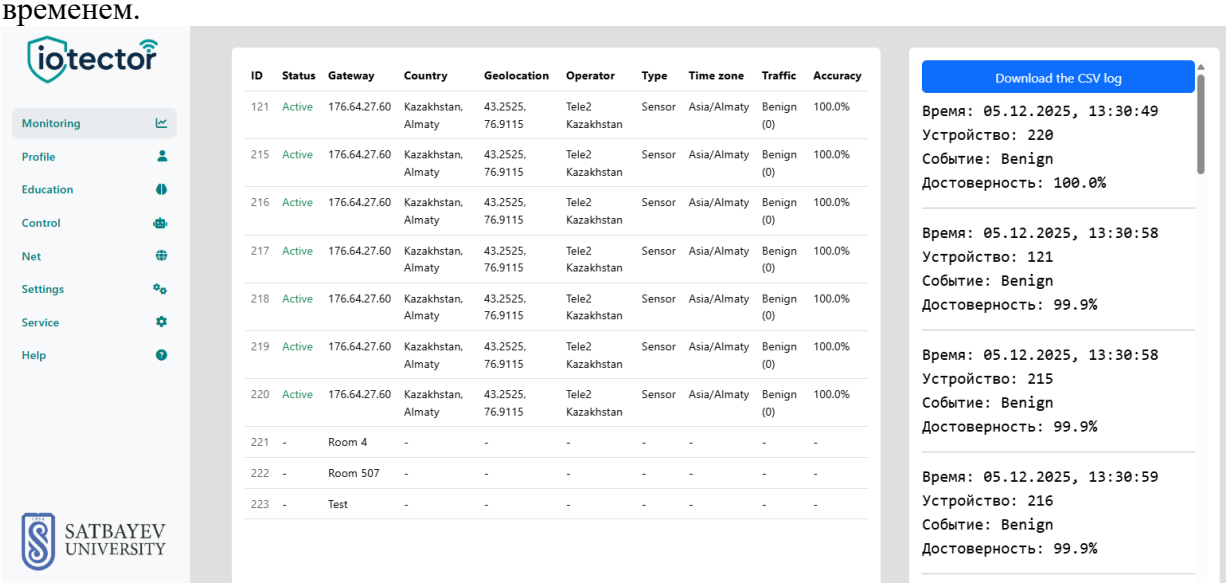


Рисунок 4 – Модуль управления IoT-устройствами

Модуль обучения и валидации моделей (рисунок 5) позволяет выбирать архитектуру модели и запускать процесс обучения непосредственно через интерфейс платформы. После завершения обучения пользователь может проанализировать кривые сходимости и матрицу ошибок. Это упрощает сравнение моделей друг с другом и их настройку перед развёртыванием в рабочей среде.

Благодаря этому IoTector объединяет процессы обучения моделей, их оценки и последующего использования в системе мониторинга в рамках единой программной платформы.

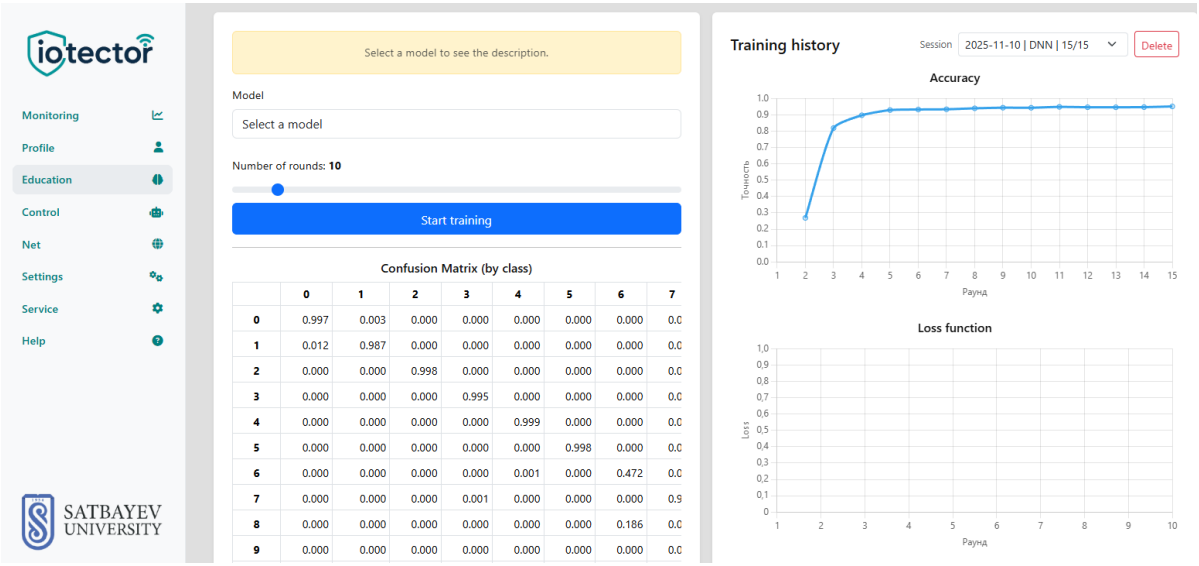


Рисунок 5 – Интерфейс обучения и валидации моделей

Результаты экспериментального исследования подтверждают эффективность платформы IoTector для обнаружения кибератак в IoT-сетях в условиях реального развёртывания. Модель CNN продемонстрировала наилучшее соотношение точности и вычислительной эффективности среди трёх исследованных архитектур, достигнув Accuracy 95.27% и F1-меры 94.92% при времени инференса 107 мс на Raspberry Pi 5. В работах [1, 6, 12] оценка эффективности выполнялась преимущественно в программной среде. В данном исследовании дополнительно проведено тестирование на аппаратной платформе Raspberry Pi 5. Это позволило оценить работу системы в условиях, приближенных к её практическому использованию.

Экспериментальная часть была выполнена на выборке, сформированной на основе датасета CICIoT2023, а тестирование проводилось в одиночной конфигурации IoTector. По этой причине полученные результаты следует рассматривать с учётом особенностей используемого набора данных и условий аппаратного тестирования. Федеративное обучение в данной работе рассматривается как архитектурное расширение платформы и требует отдельной экспериментальной проверки. В дальнейшем планируется оценить IoTector в многозвенной IoT-среде, проверить федеративный режим и исследовать устойчивость системы к новым типам атак.

**Заключение.**

В ходе исследования была разработана и реализована платформа IoTector, предназначенная для мониторинга сетевого трафика и обнаружения кибератак в IoT-сетях. Система развернута на базе Raspberry Pi 5 и включает средства мониторинга, регистрации событий безопасности, управления устройствами и работы с моделями глубокого обучения.

Проведённая экспериментальная оценка показала, что среди рассмотренных архитектур наиболее высокие результаты продемонстрировала модель CNN. На наборе данных CICIoT2023 она обеспечила точность классификации 95,27% при значении F1-меры 94,92%. Среднее время инференса составило около 107 мс — этот показатель укладывается в требования к оперативному анализу трафика, хотя дальнейшее снижение задержки остаётся желательным для более нагруженных сценариев эксплуатации.

Полученные результаты свидетельствуют о возможности практического применения IoTector в качестве компонента системы мониторинга и обнаружения атак в IoT-инфраструктуре. Вместе с тем выполненные эксперименты проводились в рамках одиночной конфигурации платформы, поэтому дальнейшая работа будет связана с исследованием многозвенных сценариев развертывания и оценкой возможностей

федеративного обучения. Дополнительный интерес представляет анализ устойчивости системы к новым типам атак и изменяющимся условиям функционирования IoT-сред.

**Список литературы**

1. Албанбай, Н., Турсынбек, Е., Граффи, К., Ускенбаева, Р., Калпеева, Ж., Абилкайыр, З., Аяпов, Е. Обнаружение вторжений в IoT-сетях на основе федеративного обучения: оценка производительности и исследование масштабирования данных. J. Sens. Actuator Netw. 2025, 14, 78. https://doi.org/10.3390/jsan14040078

2. Каддос, А., Ясин, М.У., Аль-Шамайлех, А.С., Имран, М., Ахунзада, А., Альхарти, С.З. Новая система обнаружения вторжений для оптимизации безопасности IoT. Sci. Rep. 2024, 14, 21789. https://doi.org/10.1038/s41598-024-72049-z

3. Инува, М.М., Дас, Р. Новая улучшенная нейронная сеть для обнаружения аномалий в среде IoT. Comput. Electr. Eng. 2026, 129, 110833. https://doi.org/10.1016/j.compeleceng.2025.110833

4. Саба, Т., Рехман, А., Садад, Т., Коливанд, Х., Бахадж, С.А. Система обнаружения вторжений на основе аномалий для IoT-сетей с использованием модели глубокого обучения. Comput. Electr. Eng. 2022, 99, 107810. https://doi.org/10.1016/j.compeleceng.2022.107810

5. Эльсайед, Р.А., Хамада, Р.А., Абдалла, М.И., Эльсайд, С.А. Защита систем IoT и SDN с использованием автоматического обнаружения вторжений на основе глубокого обучения. Ain Shams Eng. J. 2023, 14, 102211. https://doi.org/10.1016/j.asej.2023.102211

6. Бахш, С.А., Хан, М.А., Ахмед, Ф., Аль-Шехри, М.С., Али, Х., Ахмад, Дж. Повышение безопасности IoT-сетей с помощью системы обнаружения вторжений на основе глубокого обучения. Internet Things 2023, 24, 100936. https://doi.org/10.1016/j.iot.2023.100936

7. Вильегас-Ч, В., Говеа, Х., Мальдонадо Наварро, А., Палациос Хатива, П. Обнаружение вторжений в IoT-сетях с использованием динамического графового моделирования и графовых нейронных сетей. IEEE Access 2025, 13, 65356–65375. https://doi.org/10.1109/ACCESS.2025.3559325

8. Габер, Т., Авотанде, Дж.Б., Торки, М., Аджагбе, С.А., Хаммудех, М., Ли, В. Metaverse-IDS: система обнаружения вторжений на основе глубокого обучения для сетей Metaverse-IoT. Internet Things 2023, 24, 100977. https://doi.org/10.1016/j.iot.2023.100977

9. Чжао, Р., Ван, Й., Сюэ, З., Оцуки, Т., Адебиси, Б., Гуй, Г. Метод обнаружения вторжений на основе полуконтролируемого федеративного обучения для Интернета вещей. IEEE Internet Things J. 2023, 10, 8645–8657. https://doi.org/10.1109/JIOT.2022.3175918

10. Сахоо, Дж.П., Кар, Б., Абдельмоньем, А.М.,Чатзопулос, Д. Choir-IDS: платформа федеративного обучения для объяснимой системы обнаружения вторжений с калибровкой точности для граничных IoT-сетей. Inf. Fusion 2026, 125, 103473. https://doi.org/10.1016/j.inffus.2025.103473

11. Джавид, Д., Саид, М.С., Адил, М., Кумар, П., Джолфаи, А. Система обнаружения вторжений с нулевым доверием на основе федеративного обучения для Интернета вещей. Ad Hoc Netw. 2024, 162, 103540. https://doi.org/10.1016/j.adhoc.2024.103540

12. Карунамурти, А.,Виджаян, К., Кширсагар, П.Р., Тан, К.Т. Оптимальная система обнаружения вторжений для среды IoT на основе федеративного обучения. Sci. Rep. 2025, 15, 8696. https://doi.org/10.1038/s41598-025-93501-8

13. Рид, А., Дули, Л., Мостефауи, С.К. SA-IDS: система обнаружения вторжений по единственному атрибуту для медленных DoS-атак в IoT-сетях. Internet Things 2025, 30, 101512. https://doi.org/10.1016/j.iot.2025.101512

14. Рой, С., Ли, Дж., Чой, Б.-Дж., Бай, Й. Облегчённый механизм контролируемого обнаружения вторжений для IoT-сетей. Future Gener. Comput. Syst. 2022, 127, 276–285. https://doi.org/10.1016/j.future.2021.09.027

15. Агбедану, П.Р., Ян, С. (Джей), Мусабе, Р., Гатаре, И., Рвигема, Дж. ALMANET: гибридная система обнаружения вторжений с онлайн-обучением для обеспечения безопасности IoT в реальном времени. Egypt. Inform. J. 2025, 31, 100764. https://doi.org/10.1016/j.eij.2025.100764

16. Зухуриан, А., Дадхах, С., Молино, Х., Нето, Э.К.П., Горбани, А.А. IoT-PRIDS: использование представлений пакетов для обнаружения вторжений в IoT-сетях. Comput. Secur. 2024, 146, 104034. https://doi.org/10.1016/j.cose.2024.104034

17. Хе, В., Цай, С., Юй, Й., Лай, Й., Юань, С. IDMM-IDS: эффективная и устойчивая система обнаружения вторжений для IoT на основе инвертированной модели смеси Дирихле. Neural Netw. 2026, 193, 108002. https://doi.org/10.1016/j.neunet.2025.108002

18. Захос, Г., Мантас, Г., Порфиракис, К., Мануэль Камойш Собрал де Баштош, Дж., Родригес, Х. Обнаружение вторжений на основе аномалий для сетей IoMT: проектирование, реализация, генерация датасета и оценка алгоритмов МО. IEEE Access 2025, 13, 41994–42028. https://doi.org/10.1109/ACCESS.2025.3547572

19. Хазман, К., Гуэззаз, А., Бенкиран, С., Азроур, М., Рави, В., Алабдулатиф, А. Всесторонний обзор современных методов обнаружения аномалий для обеспечения безопасности интеллектуальных систем IoT. Comput. Mater. Contin. 2025, 85, 301–329. https://doi.org/10.32604/cmc.2025.064777

## References

1. Albanbay, N., Tursynbek, Y., Graffi, K., Uskenbayeva, R., Kalpeyeva, Z., Abilkaiyr, Z., & Ayapov, Y. (2025). Federated learning-based intrusion detection in IoT networks: Performance evaluation and data scaling study. *Journal of Sensor and Actuator Networks*, *14*(4), 78. https://doi.org/10.3390/jsan14040078

2. Qaddos, A., Yaseen, M. U., Al-Shamayleh, A. S., Imran, M., Akhunzada, A., & Alharthi, S. Z. (2024). A novel intrusion detection framework for optimizing IoT security. *Scientific Reports*, *14*(1), 21789. https://doi.org/10.1038/s41598-024-72049-z

3. Inuwa, M. M., & Das, R. (2026). A novel enhanced neural network for anomaly detection in the IoT environment. *Computers and Electrical Engineering*, *129*, 110833. https://doi.org/10.1016/j.compeleceng.2025.110833

4. Saba, T., Rehman, A., Sadad, T., Kolivand, H., & Bahaj, S. A. (2022). Anomaly-based intrusion detection system for IoT networks through deep learning model. *Computers and Electrical Engineering*, *99*, 107810. https://doi.org/10.1016/j.compeleceng.2022.107810

5. Elsayed, R. A., Hamada, R. A., Abdalla, M. I., & Elsaid, S. A. (2023). Securing IoT and SDN systems using deep-learning based automatic intrusion detection. *Ain Shams Engineering Journal*, *14*(10), 102211. https://doi.org/10.1016/j.asej.2023.102211

6. Bakhsh, S. A., Khan, M. A., Ahmed, F., Alshehri, M. S., Ali, H., & Ahmad, J. (2023). Enhancing IoT network security through deep learning-powered Intrusion Detection System. *Internet of Things*, *24*, 100936. https://doi.org/10.1016/j.iot.2023.100936

7. Villegas-Ch, W., Govea, J., Navarro, A. M., & Játiva, P. P. (2025). Intrusion detection in IoT networks using dynamic graph modeling and graph-based neural networks. *IEEE Access*. https://doi.org/10.1109/ACCESS.2025.3559325

8. Gaber, T., Awotunde, J. B., Torky, M., Ajagbe, S. A., Hammoudeh, M., & Li, W. (2023). Metaverse-IDS: Deep learning-based intrusion detection system for Metaverse-IoT networks. *Internet of Things*, *24*, 100977. https://doi.org/10.1016/j.iot.2023.100977

9. Zhao, R., Wang, Y., Xue, Z., Ohtsuki, T., Adebisi, B., & Gui, G. (2022). Semisupervised federated-learning-based intrusion detection method for internet of things. *IEEE Internet of Things Journal*, *10*(10), 8645-8657. https://doi.org/10.1109/JIOT.2022.3175918

10. Sahoo, J. P., Kar, B., Abdelmoniem, A. M., & Chatzopoulos, D. (2026). Choir-IDS: A federated learning framework for fidelity-calibrated explainable intrusion detection system for

edge-IoT networks. *Information Fusion*, *125*, 103473. https://doi.org/10.1016/j.inffus.2025.103473

11. Javeed, D., Saeed, M. S., Adil, M., Kumar, P., & Jolfaei, A. (2024). A federated learning-based zero trust intrusion detection system for Internet of Things. *Ad Hoc Networks*, *162*, 103540. 103540. https://doi.org/10.1016/j.adhoc.2024.103540

12. Karunamurthy, A., Vijayan, K., Kshirsagar, P. R., & Tan, K. T. (2025). An optimal federated learning-based intrusion detection for IoT environment. *Scientific Reports*, *15*(1), 8696. https://doi.org/10.1038/s41598-025-93501-8

13. Reed, A., Dooley, L., & Mostefaoui, S. K. (2025). SA-IDS: A single attribute intrusion detection system for Slow DoS attacks in IoT networks. *Internet of Things*, *30*, 101512. https://doi.org/10.1016/j.iot.2025.101512

14. Roy, S., Li, J., Choi, B. J., & Bai, Y. (2022). A lightweight supervised intrusion detection mechanism for IoT networks. *Future Generation Computer Systems*, *127*, 276-285. https://doi.org/10.1016/j.future.2021.09.027

15. Agbedanu, P. R., Yang, S. J., Musabe, R., Gatare, I., & Rwigema, J. (2025). ALMANET: A hybrid online learning IDS for real-time IoT security. *Egyptian Informatics Journal*, *31*, 100764. 100764. https://doi.org/10.1016/j.eij.2025.100764

16. Zohourian, A., Dadkhah, S., Molyneaux, H., Neto, E. C. P., & Ghorbani, A. A. (2024). IoT-PRIDS: Leveraging packet representations for intrusion detection in IoT networks. *Computers & Security*, *146*, 104034. https://doi.org/10.1016/j.cose.2024.104034

17. He, W., Cai, X., Yu, Y., Lai, Y., & Yuan, X. (2025). IDMM-IDS: An efficient and robust intrusion detection system for the IoT based on the inverted Dirichlet mixture model. *Neural Networks*, 108002. https://doi.org/10.1016/j.neunet.2025.108002

18. Zachos, G., Mantas, G., Porfyrakis, K., de Bastos, J. M. C. S., & Rodriguez, J. (2025). Anomaly-based intrusion detection for IoMT networks: Design, implementation, dataset generation, and ML algorithms evaluation. *IEEE Access*, *13*, 41994-42028. https://doi.org/10.1109/ACCESS.2025.3547572

19. Hazman, C., Guezzaz, A., Benkirane, S., Azrour, M., Ravi, V., & Alabdulatif, A. (2025). A comprehensive survey of contemporary anomaly detection methods for Securing smart IoT systems. *Computers Mater. Continua/Computers Mater. Continua (Print)*, *85*(1), 301-329. https://doi.org/10.32604/cmc.2025.064777

## IOT-ЖЕЛІЛЕРІНДЕ ЖЕЛІЛІК ТРАФИК МОНИТОРИНГІ МЕН КИБЕРШАБУЫЛДАРДЫ АНЫҚТАУҒА АРНАЛҒАН IOTECTOR ЖҮЙЕСІН ӘЗІРЛЕУ

***Аңдатпа***. *Заттар интернетінің (IoT) қарқынды дамуы қосылған құрылғылар санының артуына және кибершабуылдар санының көбеюіне алып келуде, бұл тиімді әрі ауқымдалатын қорғаныс тетіктерін әзірлеуді талап етеді. Осы жұмыста IoT-желілеріндегі желілік трафикті мониторингтеу мен кибершабуылдарды анықтауға арналған IoTector платформасы ұсынылады. IoTector IoT құрылғылары мен желілік инфрақұрылым арасындағы интеллектуалды шлюз ретінде қарастырылып, шабуылдар мен аномалияларды нақты уақыт режимінде анықтауды қамтамасыз етеді. Желілік трафикті талдау үшін жүйеде DNN, CNN және CNN–BiLSTM сияқты терең оқыту модельдері қолданылады, бұл әртүрлі шабуыл түрлерін тиімді анықтауға мүмкіндік береді. Платформаның прототипі Raspberry Pi 5 негізінде жүзеге асырылып, құрылғыларды Wi-Fi және Bluetooth сияқты сымсыз технологиялар арқылы қосуды қолдайды. Сонымен қатар, желі күйін мониторингтеуге, анықталған қауіптерді визуализациялауға, құрылғыларды басқаруға және модельдерді оқытуға мүмкіндік беретін бағдарламалық интерфейс әзірленді. Дәстүрлі, негізінен трафикті орталықтандырылған талдауға*

*бағытталған шабуылдарды анықтау жүйелерінен айырмашылығы, ұсынылған тәсіл интеллектуалды сүзгілеу, мониторинг және басқару функцияларын бірыңғай платформа шеңберінде біріктіреді. Алынған нәтижелер IoTector платформасының шабуылдар мен аномалияларды анықтауда жоғары тиімділік көрсететінін, сондай-ақ оның нақты IoT-орта жағдайларында практикалық тұрғыдан қолдануға жарамды екенін растайды.*

***Түйін сөздер:*** *Заттар интернеті, IoT қауіпсіздігі, шабуылдарды анықтау, желілік трафик, терең оқыту, IoT желілерін бақылау.*

## DEVELOPMENT OF THE IOTECTOR SYSTEM FOR NETWORK TRAFFIC MONITORING AND CYBERATTACK DETECTION IN IOT NETWORKS

***Abstract.*** *The rapid development of the Internet of Things (IoT) is accompanied by a growing number of connected devices and an increasing volume of cyberattacks, which necessitates the development of effective and scalable protection mechanisms. This paper proposes IoTector, a platform for network traffic monitoring and cyberattack detection in IoT networks. IoTector is designed as an intelligent gateway deployed between IoT devices and the network infrastructure, providing real-time detection of attacks and anomalies. To analyze network traffic, the system employs deep learning models, including DNN, CNN, and CNN–BiLSTM, enabling effective detection of various attack types. A prototype of the platform was implemented on Raspberry Pi 5 and supports device connectivity through wireless technologies such as Wi-Fi and Bluetooth. In addition, a software interface was developed to provide network status monitoring, threat visualization, device management, and support for model training. Unlike traditional intrusion detection systems mainly focused on centralized traffic analysis, the proposed approach integrates intelligent filtering, monitoring, and management within a unified platform. The obtained results confirm the high effectiveness of IoTector in detecting attacks and anomalies, as well as its practical applicability in real-world IoT environments.*

***Keywords:*** *Internet of Things, IoT security, intrusion detection, network traffic, deep learning, IoT network monitoring.*

**Авторлар туралы мәлімет**


| | |
|---|---|
| Турсынбек Ерлан Нуржанулы | Жетекші ғылыми қызметкер, Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті, Автоматтандыру және ақпараттық технологиялар институты Киберқауіпсіздік кафедрасы, Алматы, Қазақстан, E-mail: y.tursynbek@satbayev.university |
| Албанбай Нұртай | PhD, Қауымдастырылған профессор Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті, Автоматтандыру және ақпараттық технологиялар институты Киберқауіпсіздік кафедрасы, Алматы, Қазақстан. E-mail: n.albanbay@satbayev.university |
| Кегенбеков Жандос Кадырханович | Т.ғ.к., профессор, Қазақстан-Неміс Университеті, Алматы, Қазақстан. E-mail: kegenbekov@dku.kz |
| Аханқызы Ақбота | Ғылыми қызметкер, Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті, Автоматтандыру және ақпараттық технологиялар институты Киберқауіпсіздік кафедрасы, Алматы, Қазақстан. E-mail:a.akhankyzy@satbayev.university |


**Сведение об авторах**


| | |
|---|---|
| Турсынбек Ерлан Нуржанулы | Ведущий научный содрудник, Казахский национальный исследовательский технический университет имени К.И. Сатпаева, Алматы, Казахстан. E-mail: y.tursynbek@satbayev.university |
| Албанбай Нұртай | PhD, ассоцированный профессор, Казахский национальный исследовательский технический университет имени К.И. Сатпаева, Алматы к., Казахстан. E-mail: n.albanbay@satbayev.university |
| Кегенбеков Жандос Кадырханович | К.т.н, профессор, Казахстанско-Немецкий Университет, Алматы, Казахстан. E-mail: kegenbekov@dku.kz |

| | |
|---|---|
| Аханкызы Акбота | Научный сотрудник, Казахский национальный исследовательский технический университет имени К.И. Сатпаева, Алматы, Казахстан.<br>E-mail: a.akhankyzy@satbayev.university |


## Information about the authors


| | |
|---|---|
| Tursynbek Yerlan | Leading Researcher, Satbayev University, Institute of Automation and Information Technologies, Department of Cybersecurity, Almaty, Kazakhstan.<br>E-mail: y.tursynbek@satbayev.university |
| Albanbay Nurtay | PhD, Associate Professor, Institute of Automation and Information Technologies, Department of Cybersecurity, Lecturer, Doctoral Student Almaty, Kazakhstan<br>E-mail: n.albanbay@satbayev.university |
| Kegenbekov Zhandos | PhD (Engineering), Professor, Kazakh-German University, Almaty, Kazakhstan. E-mail: kegenbekov@dku.kz |
| Akhankyzy Akbota | Researcher, Satbayev University, Institute of Automation and Information Technologies, Almaty, Kazakhstan, E-mail: a.akhankyzy@satbayev.university |