DOI: 10.15514/ISPRAS-2025-37(6)-55

Детектирование атак на отказ в программно-конфигурируемых сетях с использованием методов машинного обучения

М.А. Лапина, ORCID: 0000-0001-8117-9142 <mlapina@ncfu.ru>
Д.Д. Гриценко, ORCID: 0009-0004-7414-5453 <rlitvindan@mail.ru>
Е.А. Кеньков, ORCID: 0009-0005-3078-1412 <kenkov2045@yandex.ru>
А.Р. Багаутдинова, ORCID: 0009-0003-4952-4497 <bagaauttdinova@mail.ru>
А.А. Соломянко, ORCID: 0009-0002-3378-6743 <artemixol@xmail.ru>

Северо-Кавказский федеральный университет,
Россия, 355017, г. Ставрополь, ул. Пушкина, д.1.

Аннотация. Программно-конфигурируемые сети представляют собой современный подход к управлению сетевыми ресурсами, обеспечивающий гибкость и масштабируемость. Однако их централизованная архитектура делает их уязвимыми для различных типов атак, включая распределенные атаки на отказ в обслуживании. Такие атаки могут привести к значительным финансовым и операционным потерям, что подчеркивает необходимость разработки эффективных методов их обнаружения и предотвращения. В статье рассматриваются методы машинного обучения для противодействия DDoS-атакам в программно-конфигурируемых сетях. Исследование включает анализ атак, с использованием протоколов TCP, ICMP и UDP. Для обнаружения аномалий были применены модели машинного обучения, такие как k-ближайших соседей, деревья решений и вероятностные нейронные сети. Особое внимание уделено борьбе с переобучением с использованием методов кросс-валидации и метода главных компонент. В работе предлагаются сценарии обнаружения и классификации DDoS-атак в программно-конфигурируемых сетях использованием машинного обучения, а также оценка их точности. Результаты исследования помогут оценить эффективность выбранных моделей, на основе метрик F-меры, точности и полноты.

Ключевые слова: машинное обучение; программная платформа KNIME; сетевые атаки DDoS; программно-конфигурируемые сети; протокол.

Для цитирования: Лапина М.А., Гриценко Д.Д., Кеньков Е.А., Багаутдинова А.Р., Соломянко А.А. Детектирование атак на отказ в программно-конфигурируемых сетях с использованием методов машинного обучения. Труды ИСП РАН, том 37, вып. 6, часть 4, 2025 г., стр. 139–158. DOI: 10.15514/ISPRAS–2025–37(6)–55.

Благодарности: Исследование выполнено при поддержке Российского научного фонда, проект № 25-71-30007.

Detection of Denial-of-service Attacks in Software-configurable Networks Using Machine Learning Methods

M.A. Lapina, ORCID: 0000-0001-8117-9142 <mlapina@ncfu.ru>
D.D. Gritsenko, ORCID: 0009-0004-7414-5453 <rlitvindan@mail.ru>
E.A. Kenkov, ORCID: 0009-0005-3078-1412 <kenkov2045@yandex.ru>
A.R. Bagautdinova, ORCID: 0009-0003-4952-4497 <bagaauttdinova@mail.ru>
A.A. Solomyanko, ORCID: 0009-0002-3378-6743 <artemixol@xmail.ru>

North Caucasus Federal University,
1, Pushkina str., Stavropol, 355017, Russia.

Abstract: Software-defined networks (SDNs) represent a modern approach to network resource management, providing flexibility and scalability. However, their centralized architecture makes them vulnerable to various types of attacks, including distributed denial-of-service attacks. Such attacks can result in significant financial and operational losses, highlighting the need to develop effective detection and prevention methods. This paper examines machine learning methods for mitigating DDoS attacks in SDNs. The study includes an analysis of attacks using the TCP, ICMP, and UDP protocols. Machine learning models such as k-nearest neighbors, decision trees, and probabilistic neural networks were applied to detect anomalies. Special attention is paid to combating overfitting using cross-validation and principal component analysis. The paper proposes scenarios for detecting and classifying DDoS attacks in SDNs using machine learning, as well as assessing their accuracy. The results of the study will help evaluate the effectiveness of the selected models based on the F-measure, Precision, and Recall metrics.

Keywords: machine learning; KNIME; DDoS; software-defined networking; protocol.

For citation: Lapina M.A., Gritsenko D.D., Kenkov E.A., Bagautdinova A.R., Solomyanko A.A. Detection of denial-of-service attacks in software-configurable networks using machine learning methods. Trudy ISP RAN/Proc. ISP RAS, vol. 37, issue 6, part 4, 2025, pp. 139-158 (in Russian). DOI: 10.15514/ISPRAS-2025-37(6)-55.

Acknowledgements: The research was supported by the Russian Science Foundation Grant No. 25-71-30007.

1. Введение

В настоящее время важной задачей в области развития информационно-телекоммуникационных систем является внедрение новых сетевых технологий, наиболее перспективные из которых основаны на применении программно-конфигурируемых сетей (ПКС). Характерная особенность таких сетей заключается в разделении плоскостей управления и передачи данных, что позволяет централизовать управление сетью в ПКС-контроллере и обеспечить ряд преимуществ, таких как гибкость, масштабируемость и увеличение надежности функционирования сети. Однако их централизованная архитектура делает их уязвимыми для различных кибератак: подмена данных, атаки "человек посередине", а также DDoS, что расшифровывается как "Распределенный отказ в обслуживании" (Distributed Denial of Service). Подобные атаки могут привести к значительным финансовым и операционным потерям. DDoS в ПКС представляют особую угрозу из-за централизованного характера управления сетью. В исследованиях [1-2] рассматриваются различные компоненты ПКС, которые могут быть подвержены атакам.

DDoS атака реализуется массовыми запросами, превышающими допустимый объем, которые перегружают сервер и делают его недоступным для пользователей. Перегрузка системы запросами приводит к ее простоям, из-за которых организации могут нести убытки. Компьютер злоумышленника инициирует соединения, которые ничем не отличаются от действий легитимных клиентов. В совокупности все эти действия могут создать нагрузку, превышающую расчетную [3].

Существует похожий тип атаки DoS (Denial of Service) или "Отказ в обслуживании", это так же перегрузка системы с помощью сетевых запросов, но, в отличие от DDoS, при проведении DoS-атаки запросы осуществляются с одного компьютера. Они менее эффективны и более заметны, ведь атака осуществляется с одного IP-адреса источника [3].

Начиная с первых DDoS-угроз для ПКС, представлявших собой простые попытки переполнения памяти, происходила эволюция к более сложным адаптивным атакам. Традиционные методы защиты, такие как фильтрация пакетов и ограничение частоты запросов [1], становятся все менее эффективными против современных угроз. Целью работы является разработка и анализ сценариев противодействия атакам типа "отказ в обслуживании" в программно-конфигурируемых сетях с применением методов машинного обучения для повышения устойчивости и безопасности сетевой инфраструктуры.

2. Описание архитектуры и DDoS атак на ПКС

Для начала рассмотрим архитектуру ПКС. В сети выделяются три уровня (рис.1): инфраструктурный уровень (набор сетевых устройств, таких как коммутаторы и каналы передачи данных), уровень управления (сетевая ОС для обеспечения приложений сетевыми сервисами и программным интерфейсом) и уровень сетевых приложений (приложения для мониторинга, управления и обеспечения безопасности сети) [4].

Рис. 1. Архитектура программно-конфигурируемых сетей.
Fig. 1. Software-defined networking architecture.

Для связи инфраструктуры сети и уровня управления обычно используют протокол OpenFlow [5]. Принцип работы этого протокола заключается в том, что при поступлении первого пакета нового потока на коммутатор извлекает из заголовка адрес (IP или MAC). Если этот адрес есть в таблице потоков, то данные передаются в коммутационную матрицу, в противном случае по защищенному каналу управления отправляется запрос на ПКС-контроллер, и на основании полученной от него информации вносятся изменения в таблицу коммутации. После этого все последующие пакеты этого потока будут обрабатываться на основе новой записи. Каждый коммутатор содержит как минимум одну таблицу потоков, и для каждой записи в этой таблице существует набор определенных инструкций. Инструкции в каждой записи содержат действия по обработке пакетов и могут прекращать их обработку и пересылать на выходной порт, либо продолжать обработку следующих таблиц потоков, при этом перенаправляя пакеты только к таблицам с номером больше, чем текущий, поскольку пересылки назад запрещены [6].

Сети ПКС, как и все сети, подвержены различным угрозам [7], одной из таких угроз, являются атаки DDoS. Так как в основе работы ПКС находятся коммутационные матрицы, DDoS атаки на ПКС осуществляются зачастую по следующему сценарию: злоумышленник, зная, что сеть является программно-конфигурируемой, посылает новый запрос и, пока он обрабатывается контроллером, начинает посылать множество запросов на коммутаторы, тем самым перегружая сеть [8].

Одним из решений проблемы раннего обнаружения DDoS-атак является интеграция инструмента Snort [9] с ПКС-контроллерами. Snort представляет собой одну из популярных систем обнаружения и предотвращения вторжений. В работе [10] представлены эксперименты и результаты обнаружения DDoS-атак с использованием системы SNORT IDS (Intrusion Detection System) на контроллерах ODL [11] и ONOS [12]. Для анализа производительности авторами применялись различные сценарии с разным количеством хостов, коммутаторов и генерируемого трафика данных. В исследовании [2] представлены различные компьютерные атаки на сети ПКС, в том числе и атаки с переполнением (flood), там же описываются эксперименты и их результаты.

Одной из целей атак на сети ПКС являются транспортные протоколы TCP и UDP, а также сетевой протокол ICMP. Их стоит рассмотреть в первую очередь.

TCP (Transmission Control Protocol), или протокол управления передачей, является одним из основных протоколов для передачи данных. Его принцип заключается в предварительном установлении соединения между отправителем и получателем. Механизм работы протокола заключается в следующем: все отправленные данные подтверждаются контрагентом, подтверждение генерируется для всех данных с начала пакета, если подтверждение не поступает в течение некоторого времени, протокол снова отправляет данные и перезапускает таймер [13].

UDP (User Datagram Protocol), или протокол пользовательских датаграмм, отличается от TCP тем, что одна сторона отправляет пакеты, адресуя их другой стороне без установления соединения. Отправитель не располагает информацией о том, готов ли получатель к приему информации, и присутствует ли получатель на месте. Иногда пакеты, отправляемые по протоколу UDP, называются датаграммами [13].

Протокол ICMP (Internet Control Message Protocol) – это протокол сетевого уровня, интегрированный с протоколом IP (Internet Protocol). Он предназначен для обмена сообщениями об ошибках и операционной информацией между сетевыми устройствами [7]. В настоящее время активно используются две версии протоколов IP и ICMP, это IPv4 (ICMPv4) и IPv6 (ICMPv6). Разница между ними заключается в том, что адресное пространство составляет 2^{32} адресов для IPv4 и 2^{128} адресов для IPv6.

2.1 Атаки на протоколы

Наиболее распространенной атакой на протокол TCP является атака SYN-flood. Она основана на некоторых особенностях "трехстороннего рукопожатия" в результате установления соединения. Основной механизм атаки заключается в направлении на целевой узел TCP-пакетов с флагом SYN, исходящих с поддельных IP-адресов. Коммутатор, не находя правил для данных потоков, начинает генерировать для контроллера множество запросов решений. Данный процесс создает нагрузку на канал управления и ресурсы контроллера, что приводит к отказу в обслуживании [10].

При атаке UDP-flood большие объемы UDP-трафика с поддельными IP-адресами отправителя посылаются на случайные порты целевой системы. При этом коммутатор, также не находя правил для данных потоков, начинает генерировать для контроллера множество запросов решений. Поскольку системе необходимо проверять порт, указанный в каждом входящем пакете, на наличие подслушивающего приложения, ресурс целевого сервера может быть

быстро исчерпан, в результате чего он будет недоступен для обычного трафика и законных пользователей. Интернет-соединения могут легко стать перегруженными [10].

В случае атаки ICMP-flood [14] активируется механизм эхо-ответа. При атаке ICMP-flood злоумышленник отправляет большое количество IP-пакетов, содержащих ICMP-сообщения в систему жертвы. Аналогично атакам SYN-flood, коммутатор, не находя соответствующие правила отправляет запросы на контроллер. Это приводит к неэффективной трате системных ресурсов жертвы. Атака значительно снижает пропускную способность сети, в результате чего система не может обрабатывать реальные пакеты данных[10].

Проведенный анализ DDoS угроз для ПКС, позволил обратить внимание на работы, которые предлагают различные решения данной проблемы.

Основные проблемы сетей ПКС, затронутые в работе [2], заключаются в уязвимости центрального контроллера к атакам, таким как DDoS, отсутствие эффективных методов оценки устойчивости ПКС. Авторы статьи предлагают использовать комплексный подход для исследования вероятностно-временных характеристик с использованием метода топологического преобразования стохастических сетей, а также имитационную модель ПКС и экспериментальную проверку результатов. Результаты показали, что метод обеспечивает высокую точность моделирования (разница между аналитической и имитационной моделями не более 5%). На практике авторы для оценки устойчивости ПКС к кибератакам и разработки систем защиты предлагают использовать метод, основанный на топологическом преобразовании стохастических сетей.

В работе [7] описывается, что управление трафиком в ПКС сталкивается с такими проблемами как, непредсказуемость потоков трафика, ограничения традиционных статистических методов управления, неспособных быстро адаптироваться к изменяющейся сетевой среде и сложности эффективного распределения сетевых ресурсов. Авторы предлагают следующие методы машинного обучения для адаптивного управления трафиком: обучение с учителем, обучение без учителя, применение моделей глубокого обучения для анализа и прогнозирования трафика и интеграция машинного обучения в ПКС-контролеры для автоматического управления. В статье приводятся примеры успешного применения машинного обучения в управлении трафиком в программно-определяемых сетях (Software-Defined Networking, SDN), и авторы приходят к выводу, что машинное обучение повышает адаптивность, эффективность и гибкость управления, позволяя эффективно распределять ресурсы, предсказывать и предотвращать аномалии.

В работе [15] поднимаются вопросы проблемы обеспечения безопасности в сетях ПКС. Основное внимание уделяется уязвимости центрального контроллера ПКС, а также необходимости адаптивных методов защиты. Авторы исследования рассматривают методы машинного обучения для обеспечения безопасности управления трафиком в ПКС, такие как алгоритмы классификации методом опорных векторов (Support Vector Machine, SVM), нейронные сети (в частности, многослойный персептрон). В статье демонстрируется, что методы машинного обучения позволяют достигать высокой точности в классификации атак и адаптироваться к новым угрозам в ПКС.

Статья [16] посвящена классификации и идентификации сетевого трафика в ПКС. Авторы утверждают, что существующие методы сигнатурного анализа и классификации на основе портов имеют существенные ограничения и предлагают для решения возникающих проблем использовать методы машинного обучения. По результатам исследования наиболее перспективным решением являются сверточные нейронные сети, метаэвристический алгоритм оптимизации Ant-Lion (ALO) и метод машинного обучения самоорганизующихся карт (Self-Organizing Maps, SOM).

В статье [16] разбираются такие проблемы неэффективности существующих методов управления сетевыми ресурсами в ПКС как уязвимость из-за центрального контроллера,

неспособность классической маршрутизации справляться с ростом трафика и ограниченная применимость традиционных методов распределения нагрузки. Автор предлагает технологию сетевых взаимодействий на основе намерений (Intent-Based Networking, IBN), перечисляя следующие преимущества: возможность балансировки нагрузки и управления до возникновения инцидентов, непрерывный мониторинг и автоматическая перенастройка сети на основе заданных программ и машинного обучения. Автор заявляет, что существующие методы управления ресурсами в ПКС имеют ограничения и узкую специализацию. Для дальнейшего развития автором предлагается исследование возможностей интеграции с IND в ПКС архитектуры и разработка стандартизированных интерфейсов.

В статьях [17-20] проведены исследования с использованием технологий машинного обучения для решения актуальных задач кибербезопасности: подмена данных, fishing и атаки SQL-инъекциями. В работах также анализируются различные модели машинного обучения на платформе для анализа данных KNIME. Результаты демонстрируют, что обнаружение различных киберугроз при помощи методов машинного обучения требует тщательного отбора признаков и оптимизации параметров моделей, но при правильной реализации обеспечивает значительное улучшение защиты информационных систем.

3. Описание набора данных

Для проведения экспериментов был выбран набор данных "dataset_sdn" с сайта kaggle [21]. В наборе имеется таблица, содержащая среди прочего следующую информацию о сетевых потоках: количество пакетов, длительность потока, используемый протокол и пропускная способность (табл. 1).

Табл. 1. Описание колонок набора данных с указанием типа признаков.
Table 1. Description of the dataset columns indicating the type of features.

Поле	Тип признака	Содержание
Dt	Количественный	Временная метка
Switch	Категориальный	Уникальный идентификатор устройства
src	Категориальный	IP-адрес источника
dst	Категориальный	IP-адрес назначения
pktcount	Количественный	Число пакетов
bytecount	Количественный	Объем данных в байтах
dur	Количественный	Длительность в секундах
dur_nsec	Количественный	Дополнительная длительность в наносекундах
tot_dur	Количественный	Общая длительность в наносекундах
flows	Количественный	Количество потоков
packetins	Количественный	Число входящих сообщений
pktperflow	Количественный	Среднее число пакетов на поток
byteperflow	Количественный	Средний объем данных на поток
pktrate	Количественный	Скорость передачи пакетов
Pairflow	Категориальный	Идентификатор пары потоков
Protocol	Категориальный	Название протокола
port_no	Категориальный	Номер порта назначения(на устройстве)
tx_bytes	Количественный	Переданные байты
rx_bytes	Количественный	Принятые байты
tx_kbps	Количественный	Скорость передачи
rx_kbps	Количественный	Скорость приема
tot_kbps	Количественный	Общая скорость
label	Категориальный	Метка класса

Классификация трафика происходит, исходя из столбца label. В нем представлены два класса – "0" и "1". 63561 записей соответствуют классу "0" и обозначают нормальный трафик, классу "1" соответствует 40784 записей, и они обозначают аномальный трафик. Аномальный трафик в свою очередь включает три типа атаки: SYN-flood, UDP-flood и ICMP-flood.

Для правильного анализа мы исключили те столбцы таблиц записей о данных, которые не несут практической ценности для обучения модели (см. табл. 2).

Табл. 2. Исключённые параметры.
Table 2. Excluded parameters.

SRC	IP-адреса не несут смысловой нагрузки (для обнаружения DDoS-атак важны объемы данных, а не конкретные адреса).
DST	
Dur_nsec	Точность до наносекунд избыточна для анализа сетевого трафика
Switch	Категориальный признак. Модель должна работать для всей сети, а не для конкретных устройств, этот признак добавляет шум.
Dt	Сырой timestamp неинформативен для большинства моделей.
Flows	Flows используется в расчете других признаков (например, pktperflow = pktcount / flows), он становится избыточным.
Port_no	Номер порта не является важным, так как атака часто ведется на случайные порты.
tx - rx	Избыточность в сочетании с bytecount, tot_kbps и pktrate
Pairlows	Сложный для интерпретации признак

4. Моделирование

Для изучения DDoS в программно-определяемых сетях было решено использовать KNIME – платформу для анализа данных. Основные блоки, использованные для реализации машинного обучения, показаны в табл. 3 и на рис. 2.

Табл. 3. Описание блоков.
Table 3. Description of blocks.

File reader	Считывает данные из файла (CSV, Excel, JSON и др.) в табличный формат KNIME
Normalizer	Нормализует числовые колонки, приводя значения к заданному диапазону (0–1)
Column filter	Фильтрует колонки по имени или типу данных
One to many	Преобразует категориальную колонку в набор бинарных колонок
Number to string	Конвертирует числовые значения в строковые
PCA (principal component analysis)	Сокращает размерность данных, выделяя главные компоненты
X-Partitioner/X-Agregator	Блоки для реализации кросс-валидации
Learner	Обучающий блок, обучает модель (например, DecisionTree, RandomForest) на тренировочных данных
Predictor	Применяет обученную модель к новым данным для получения предсказаний
Scorer	Считает метрики качества модели

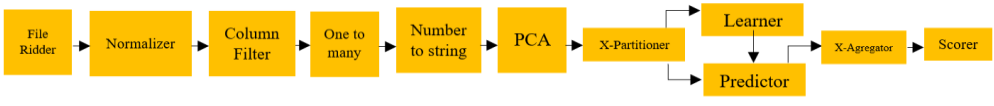

Рис. 2. Обобщающая схема обучающей модели в Knime.
Fig. 2. A general diagram of the training model in Knime.

Критерием для оценки моделей стала F-мера [22]. Причиной выбора данной метрики стало то, что набор данных, используемый в исследовании, является несбалансированным, с перекосом в сторону класса "0". Часто используются метрики полноты и точности [23], но в рамках проведенного исследования выбор пал именно на F-меру, поскольку она является гармоническим средним между полнотой и точностью [23] и является агрегированным критерием точности.

Для проведения экспериментов был сформирован набор данных, содержащий 11 ключевых признаков, включая признак label из исходного набора данных. Это позволило сфокусироваться на более важных для детектирования DDoS-атак характеристиках трафика. На данном наборе было исследовано 9 моделей машинного обучения, представляющих различные семейства алгоритмов. Исключением стало семейство k-ближайших соседей (k-Nearest Neighbor, kNN), так как для практического применения модель требует, как минимум, двухмерного пространства [24].

Представленные в табл. 4 результаты демонстрируют уровень эффективности различных алгоритмов классификации при использовании разного количества признаков, полученных с помощью блока PCA, причем нужное количество признаков определялось значением параметра dimension to reduce, изменявшемся в разных экспериментах от 1 до 10. Большинство алгоритмов достигают наивысшей точности с ростом величины этого параметра, который определяет до какого значения нужно уменьшить размерность исходного пространства признаков. У всех методов по мере уменьшения размерности наблюдается снижение точности, что говорит о потере информации в процессе редукции признаков. Наиболее эффективными оказались методы случайных деревьев (Random Forest, RF) [25], ансамблей деревьев (Tree Ensemble, TE) [26], градиентного бустинга (Gradient Boosted Trees, GBT) [27], k-ближайших соседей (k-Nearest Neighbor, kNN) [28] и вероятностной нейронной сети (Probabilistic Neural Network, PNN) [29]. Результаты работы с методом нечетких правил (Fuzzy Rule) [30] и деревом решений (Decision Tree) [31] было решено не принимать во внимание, так как этот метод при большинстве значений параметра dimension to reduce демонстрирует значение F-меры в 100%, что можно характеризовать как переобучение. Наименее эффективными оказались наивный байесовский классификатор (Naive Bayes, NB) [32] и метод логистической регрессии (Logistic Regression, LR) [33], их точность заметно уступает точности других методов (рис. 3).

Табл. 4. Общая таблица результатов вычисления F-меры по данным проведенных экспериментов.
Table 4. General table of F-measure calculation results based on the data from the experiments conducted.

PCA	RF [27]	TE [28]	PNN [31]	NB [34]	LR [35]	kNN [30]	GBT [29]	Fuzzy Rule[32]	Decision Tree[33]
10	100	100	87,2	59,3	56,7	99,9	97,7	100	100
9	100	100	87,2	58,4	58,2	99,9	97,9	100	100
8	100	100	87,2	57,1	52,5	99,9	97,9	100	100
7	100	100	86,9	60,4	52,4	100	98	100	100
6	100	100	88,1	64,2	58	99,8	98,2	100	99,9
5	100	100	84,1	66,7	59,3	99,8	98,4	100	100
4	100	100	87,6	66,6	54,6	99,9	98,3	100	100
3	100	100	92,7	62,9	44,9	99,8	94,1	100	99,8
2	88,7	90,1	90,8	60,4	48	99,7	93,1	100	99,8
1	69,9	70	95	60,4	52	-	82	99,9	99,5

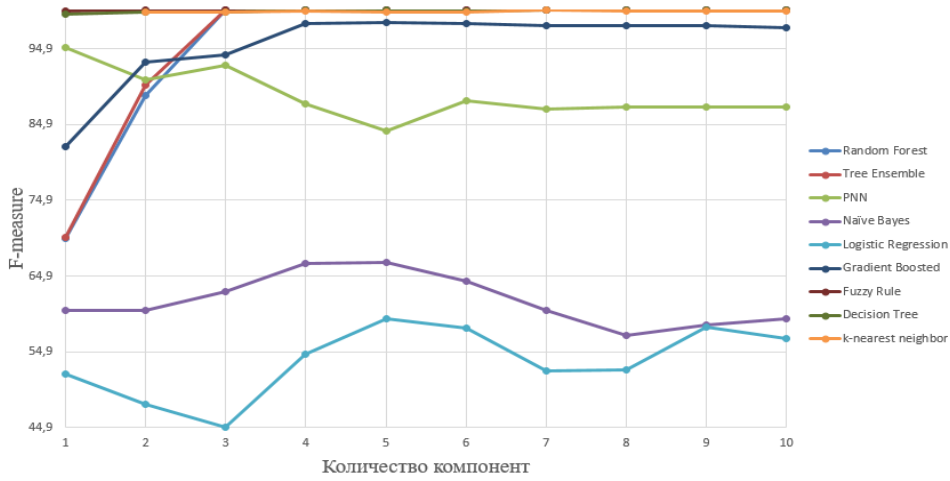

Рис. 3. График общего сравнения результатов экспериментов PCA.
Fig. 3. Overall comparison graph of PCA experiment result.

Для дальнейших экспериментов были выбраны методы kNN, GBT, TE, RF и PNN (рис. 4). Такой выбор был сделан из-за разнообразия алгоритмов: kNN представляет собой алгоритм ленивого обучения [34]; GBT – ансамблевый метод, итеративного обучения модели путем исправления своих предыдущих ошибок; TE – представляет собой ансамблевый метод построения множества деревьев решений и объединения их ответов в один; RF – ансамблевый метод построения множества деревьев и обучения их на случайных выборках данных; PNN – тип искусственных нейронных сетей, основанный на радиально-базисных функциях[35]. GBT, TE и RF были выбраны для оценки качества между ансамблевыми методами. Выбор данных моделей позволяет обеспечить всестороннее сравнение подходов к машинному обучению в задачах детектирования атак в ПКС. Кроме того, эти модели показали хороший результат детектирования, что особенно важно для систем безопасности.

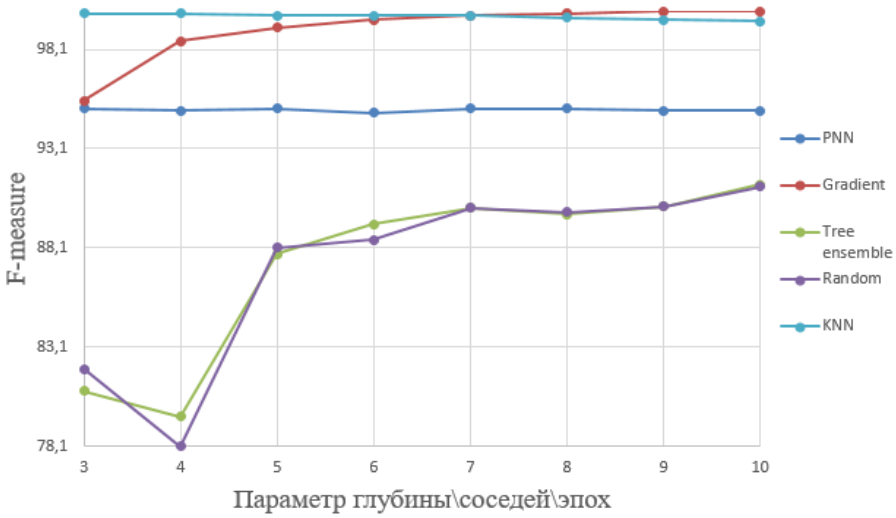

Рис. 4. График общего сравнения результатов экспериментов с различными параметрами.
Fig. 4. General comparison graph of experimental results with different parameters.

4.1 k-ближайших соседей (kNN)

Классификатор вида k-ближайших соседей исследует данные и соотносит их с наиболее подходящими классами маркированных примеров. Параметр k, определяет, сколько ближайших к текущему объекту соседей будет выбрано для использования в алгоритме.

Первый шаг исследований состоит в изучении модели и выяснении того значения параметра dimension to reduce в блоке PCA, которое дает наилучшую точность результатов. Проведены эксперименты, с модификатором в PCA dimension to reduce от 10 до 2 (табл. 5). В результате исследования были получены значения F-меры, на основе которых были составлены графики (рис. 5 и 6). Для kNN минимальное значение параметра блока PCA равно 2, так как данная модель работает с двухмерным пространством, а при указании одного признака в параметре dimension to reduce для блока PCA, все данные преобразуются в одномерное пространство.

Табл. 5. F-мера при исследовании с использованием модели kNN.
Table 5. F-measure in the study using the kNN model.

PCA	F-мера	Число соседей	F-мера
10	99,9	10	99,5
9	99,9	9	99,6
8	99,9	8	99,7
7	100	7	99,8
6	99,8	6	99,8
5	99,8	5	99,8
4	99,9	4	99,9
3	99,8	3	99,9
2	99,7	2	100

Рис. 5. График изменения значений F-меры от количества компонент при исследовании с использованием модели kNN.
Fig. 5. Graph of changes in F-measure values depending on the number of components in a study using the kNN model.

Анализ данных зависимости качества модели и числа компонент, показал, что для данной модели лучшее значение исследуемого параметра в блоке PCA равно 7 (F-мера = 100%) (рис. 5). Однако, столь высокий результат, может сигнализировать о переобучении, поэтому стоит обратить внимание на количество компонент с близкими значениями качества. Высокие показатели качества F-меры равной 99,9% так же показали эксперименты с 4, 8, 9 и 10

компонентами. Для более точных результатов стоит выбрать результат с 10 признаками. Данное количество компонент выбрано из-за того, что при большем количестве признаков сохраняется больше информации о признаках и соответственно точнее классификация.

Лучшим значением параметра для модели kNN будет являться 3, при котором F-мера равна 99,9 (рис. 6). На значениях параметра модели, равных 2, достигнуто значение F-меры, равное 100%, данный результат так же, как и в ситуации с 7 компонентами при экспериментах с PCA, может сигнализировать о переобучении. Поэтому стоит взять так же результат с близкими значениями качества, а именно 3.

Основными проблемами для данной модели может служить ее чувствительность к шуму, плохая масштабируемость и медленная скорость предсказаний на больших объемах данных.

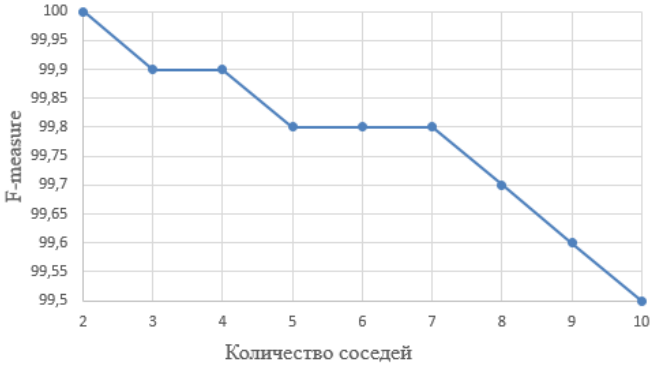

Рис. 6. График изменения значений F-меры от значений параметра модели kNN.
Fig. 6. Graph of the change in F-measure values from the values of the kNN model parameter.

4.2 Деревья решений

Деревья решений применяются в задачах классификации и регрессии. Классификация и регрессия на основе деревьев решений используются в задачах распознавания текста, информационного поиска, распознавания речи, анализе изображений, обнаружении спама, распознавания жестов и др. Для конструирований деревьев решений применяется машинное обучение – автоматическая настройка параметров алгоритма на основе обучающей выборки (множества объектов с известными правильными ответами). При этом от качества обучения зависит правильность решения задачи и практическая применимость результатов.

4.2.1 Градиентный бустинг (GBT)

Алгоритм градиентного бустинга – ансамблевый метод итеративного обучения модели путем исправления предыдущих ошибок. Дерево строится с использованием двоичных разделений для числовых и номинальных атрибутов. Встроенная обработка пропущенных значений помогает найти лучшее направление разбиения, проверяя каждое возможное направление и выбирая то, которое дает наилучший результат (то есть наибольшее уменьшение ошибки).

Прогнозируемое значение целевой переменной для листового узла дерева – это среднее целевое значение записей обучающей выборки в листе. Следовательно, прогнозы являются наилучшими (по отношению к обучающим данным), если дисперсия целевых значений в пределах листа минимальна. Это достигается путем разбиений пространства признаков, которые минимизируют сумму квадратов ошибок в соответствующих потомках текущего узла.

Само исследование проводилось также, как и исследование модели kNN. Авторами были проведены эксперименты со значениями модификатора dimension to reduce в PCA от 10 до 1.

В результате исследования были получены данные по значениям F-меры (табл. 6), на основе которых был составлен график (см. рис. 7). По полученным данным о значениях F-меры можно сделать вывод, что для данной модели лучшее значение параметра PCA равно 5 (табл. 6).

Следующие эксперименты с моделями были связаны с исследованиями глубины деревьев. Лучшая модель была получена при глубине дерева, равной 8, и значении F-меры, равном 99,9%. При значениях параметра, равных 9 и 10, достигнута F-мера в 100%. Как и при экспериментах с моделью kNN, это проявление переобучения.

В программно-конфигурируемых сетях модель GBT может показывать лучшую точность при правильной настройке, в сравнении с другими деревьями, но в то же время из-за сложного подбора параметров неправильная настройка может приводить к потере точности модели.

4.2.2 Ансамбль деревьев (TE)

Модель ансамбля деревьев (Tree ensemble) основана на объединении нескольких деревьев решений, которые в совокупности могут повысить точность. Построение ансамбля сводится к объединению выводов одиночных деревьев решений. Обучающая выборка в заранее заданном соотношении разделяется на две – собственно обучающую и валидационную. После обучения ансамбль обрабатывает контрольную выборку, каждое дерево, независимо от других, предсказывает ответ, после чего все ответы деревьев объединяются в один.

При исследовании модели TE было принято решение выбрать в качестве экспериментального значения величину параметра PCA, достигнутое перед явным падением точности, которое оказалось равным 3 (рис. 7).

На основе полученных при проведении экспериментов результатов был построен график, из которого было видно, что наилучший результат по F-мере достигается при глубине дерева, равном 10 (табл. 7).

Недостаток модели проявляется из-за требования больших вычислительных ресурсов, с одновременным ограничением вычислительной мощности контроллеров. В то же время модель достаточно эффективно способна подстраиваться на работу с новыми шаблонами атак из-за возможности "замены" отдельных деревьев без полного переобучения.

4.2.3 Случайный лес (RF)

Случайный лес (Random forest) – это ансамблевый метод машинного обучения, который для повышения устойчивости модели строит множество деревьев решений, обучающихся на случайной выборке данных, и объединяет их предсказания. Это позволяет снизить корреляцию между деревьями.

Аналогично работе с ансамблем деревьев, было принято решение взять значение параметра блока PCA перед падением точности, в результате было взято значение 3.

Модель имеет несколько критических моментов, в частности, долгое время обучения, что может плохо сказаться на детектировании атак. Из-за структуры модели процесс объединения результатов всех деревьев может потребовать слишком длительного времени, что в рамках мониторинга трафика в реальном времени может задерживать принятие решений.

Обобщенные данные для метода случайных деревьев показаны в табл. 7 и 8, а также на рис. 7 и 8.

Модель случайного леса достигает максимального значения F-меры 99,9% (табл. 7). В отличие от нее, структурно схожие модели (ансамбль деревьев и случайный лес) показали результат ниже 92%, это может показывать, что модель GBT более чувствительна к изменениям трафика в отличие от моделей TE и RF.

Табл. 6. F-мера при исследовании с использованием различных моделей деревьев решений.
Table 6. F-measure in a study using different decision tree models.

PCA	Gradient	Random	Ensemble
10	97,7	100	100
9	97,9	100	100
8	97,9	100	100
7	98	100	100
6	98,2	100	100
5	98,4	100	100
4	98,3	100	100
3	94,1	100	100
2	93,1	88,7	90,1
1	82	69,9	70

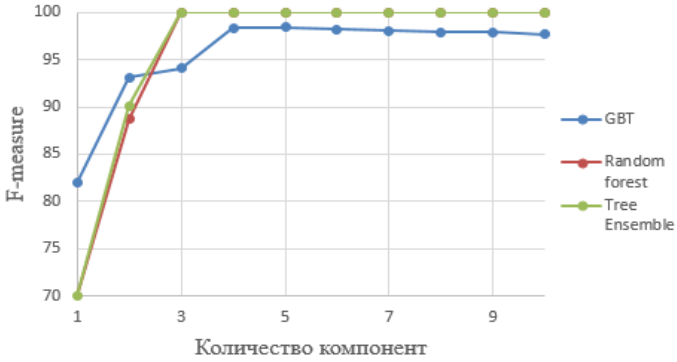

Рис. 7. График точности деревьев при экспериментах с деревьями решений.
Fig. 7. Accuracy plot of trees in decision tree experiments.

Табл. 7. F-мера при экспериментах с глубиной деревьев решений.
Table 7. F-measure in experiments with the depth of decision trees.

Tree depth	Gradient	Tree ensemble	Random
10	100	91,3	91,2
9	100	90,2	90,2
8	99,9	89,8	89,9
7	99,8	90,1	90,1
6	99,6	89,3	88,5
5	99,2	87,8	88,1
4	98,5	79,6	78,1
3	95,5	80,9	82

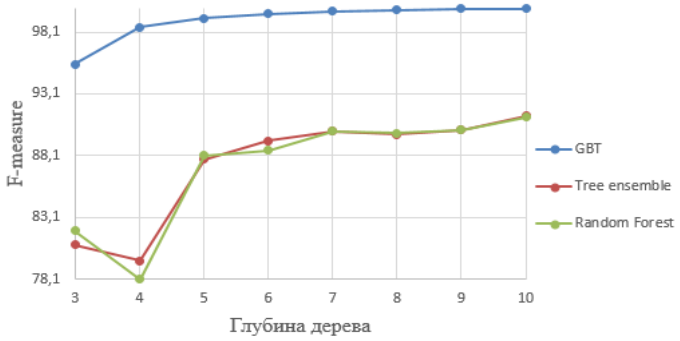

Рис. 8. График изменения значения F-меры в экспериментах с глубиной деревьев решений.
Fig. 8. Graph of the change in the F-measure value in experiments with the depth of decision trees.

4.3 Вероятностная нейронная сеть (PNN)

Алгоритм, основанный на вероятностных нейронных сетях, генерирует правила на основе обрабатываемых числовых данных. Каждое правило определяется как многомерная гауссова функция, которая регулируется двумя пороговыми значениями, тета-минус и тета-плюс, чтобы избежать конфликтов с правилами разных классов. Каждая гауссова функция определяется центром (вектором признаков объекта из обучающей выборки) и стандартным отклонением, которое корректируется в процессе обучения, чтобы охватывать только неконфликтующие объекты своего класса.

В табл. 8 показаны результаты, которые были получены при исследовании параметров блока PCA с моделью нейронной сети. Для точного исследования наиболее пригодна модель с параметром PCA, равным 1.

Табл. 8. F-мера в экспериментах с нейронными сетями.
Table 8. F-measure in experiments with neural networks.

PCA	F-мера
10	87,2
9	87,2
8	87,2
7	86,9
6	88,1
5	84,1
4	87,6
3	92,7
2	90,8
1	95

Рис. 9. График зависимости F-меры от значения параметра блока PCA.
Fig. 9. Graph of the dependence of the F-measure on the PCA block parameter value.

После 7 эпох обучения, где каждая эпоха представляет собой полный проход через весь набор данных, модель продемонстрировала точность 95,1% (табл. 9, рис. 10). Выбор данного количество эпох обусловлен возможностью недообучения на меньшем количестве эпох, а 8 эпох было исключено из-за стабильности результата и в целях экономии вычислительных ресурсов. Данная модель требовательна к вычислительным ресурсам из-за необходимости хранения и обработки всех образцов, и она может перегружать контроллеры; также она плохо масштабируется и адаптируется к изменениям данных.

Табл. 9. Зависимость F-меры от количества эпох в экспериментах с нейронными сетями.
Table 9. Dependence of the F-measure on the number of epochs in experiments with neural networks.

epoch	f-measure
10	95
9	95
8	95,1
7	95,1
6	94,9
5	95,1
4	95
3	95,1

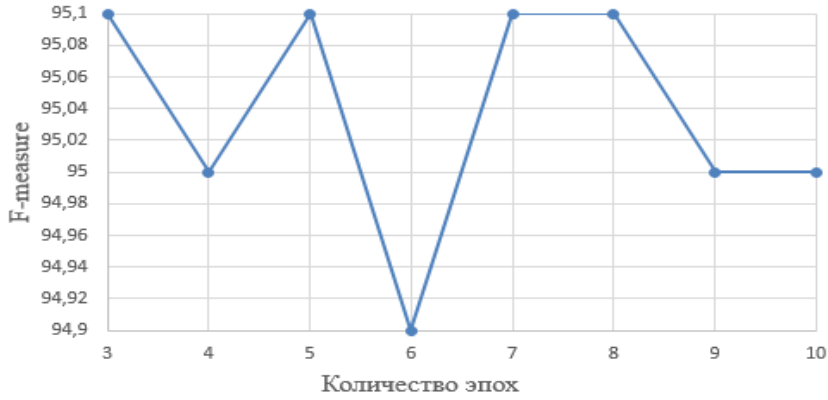

Рис. 10. График зависимости F-меры от числа эпох нейронной сети.
Fig. 10. Graph of the dependence of the F-measure on the number of epochs of the neural network.

5. Борьба с переобучением

Проблема переобучения в машинном обучении является критически важной при создании надежных систем обнаружения киберугроз, таких как программы-вымогатели. Переобучение возникает, когда модель слишком точно подстраивается под конкретные особенности обучающих данных, включая случайные шумы и редкие аномалии, что значительно снижает ее способность корректно работать с новыми, ранее не встречавшимися данными. В сфере кибербезопасности эта проблема особенно актуальна, поскольку злоумышленники постоянно совершенствуют методы обхода систем защиты, в том числе специально разрабатывают техники для дезориентации алгоритмов машинного обучения. Ключевая сложность заключается в следующем: сложные модели с множеством параметров могут демонстрировать исключительно высокую точность на обучающей выборке, но при этом оказываются практически бесполезными при столкновении с реальными атаками. Это происходит потому, что такие модели запоминают конкретные примеры вместо выявления истинных закономерностей, что делает их уязвимыми к новым, неизвестным ранее методам атак. В результате система может пропускать реальные угрозы или выдавать ложные срабатывания, что снижает ее практическую ценность для защиты информационных систем (см. работу [36]).

Одним из доступных способов борьбы с переобучением в Knime является кросс-валидация. Реализуется она при помощи двух блоков X-Partitioner и X-Aggregator. Для того, чтобы кросс-валидация работала правильно следует предпринять следующие шаги:

1. Определить число валидаций, в лучшем случае оно равно 5.
2. Выбрать целевой параметр (в нашем случае это label)

Нами используется 5-кратная кросс-валидация, данные разделяются на 5 подмножеств: модель обучается на 4-х частях и тестируется на 5-ой, это позволяет получить снижение дисперсии метрик (значение F-меры колеблется в пределах ±0.5% для kNN и деревьев). Для борьбы с переобучением может также использоваться алгоритм PCA, который на платформе Knime представлен как блок, в котором меняется параметр dimension to reduce. Снижение размерности данных, позволило устранить мультиколлинеарность и уменьшить влияние шумовых признаков. Так же был использован способ упрощения моделей, для деревьев решений и модели kNN. Для деревьев оптимальная глубина находится в диапазоне от 6 до 8 (F-мера от 99,2% до 99,8%), если делать деревья глубже 8, возникает переобучение (F-мера доходит до 100%). Для модели kNN лучший результат был достигнут при k = 3, меньшие значение 2 давало F-меру в 100%.

На основе исследований была составлена сравнительная значений F-меры для различных моделей, в которую внесены лучшие результаты (см. табл. 10).

Табл. 10. Сравнительная характеристика моделей.
Table 10. Comparative characteristics of models.

Модель	**F-мера**
kNN	99,9%
Gradient boosted	99,9%
Tree Ensemble	91,3%
Random forest	91,2%
PNN	95,1%

6. Заключение

Проведенное исследование подтвердило эффективность методов машинного обучения для обнаружения и классификации DDoS-атак в программно-конфигурируемых сетях. Анализ сетевого трафика и применение таких моделей, как k-ближайших соседей, деревья решений и вероятностные нейронные сети, позволили достичь высокой точности (F-мера до 99,9%) в идентификации аномалий. Для борьбы с переобучением использовалась кросс-валидация для снижения дисперсий метрик, блок PCA позволяет устранить мультиколлинеарность и излишние шумовые признаки. Совместно с возможностями блока PCA и кросс-валидацией использовалось упрощение моделей (уменьшение глубины деревьев, уменьшение числа эпох), что позволило добиться лучших результатов.

Наиболее эффективной и перспективной моделью использования в программно-конфигурируемых сетях является модель градиентного бустинга, так как она может показывать лучшую точность обработки трафика при правильной настройке, в свою очередь другие модели (например, k-ближайших соседей) неэффективно использовать в ПКС из-за их чувствительности к шуму и плохой масштабируемости.

Результаты работы подтверждают, что методы машинного обучения обладают высоким потенциалом в задачах обеспечения сетевой безопасности в программно-конфигурируемых

сетях, но модели следует адаптировать для работы в реальном времени, следует также расширять используемые наборы данных за счет включения в них новых типов атак.

Список литературы / References

[1]. Титов Ф. М. Исследование методов защиты от атаки DDOS. Научные междисциплинарные исследования, 2021, № 5, стр. 36-41.

[2]. Саенко И. Б. и др. Модели компьютерных атак на программно-конфигурируемые сети. Наукоемкие технологии в космических исследованиях Земли, т. 15, № 1, 2023, стр. 37-47.

[3]. Distributed Denial-of-Service, DDoS (отказ от обслуживания) (online). Доступно по ссылке: https://www.tadviser.ru/index.php/Статья:Distributed_Denial-of-Service,_DDoS_(отказ_от_обслуживания), дата обращения: 15.08.2025.

[4]. Смелянский Р. Л. Программно-конфигурируемые сети. Открытые системы. СУБД, 2012, № 9, стр. 15-26.

[5]. Shin S. W. et al. Fresco: Modular composable security services for software-defined networks. 20th annual network & distributed system security symposium. Ndss, 2013. Available at: https://www.ndss-symposium.org/wp-content/uploads/2017/09/07_2_0.pdf, accessed 07.12.2025.

[6]. Захаров В.А., Смелянский Р.Л., Чемерицкий Е.В. Формальная модель и задачи верификации программно-конфигурируемых сетей. Моделирование и анализ информационных систем. 2013;20(6):36-51. DOI: 10.18255/1818-1015-2013-6-36-51.

[7]. Нурудинов Г. М. Адаптивное управление трафиком в SDN-сетях с применением машинного обучения. Экономика и качество систем связи, 2024, № 1 (31), стр. 114-122.

[8]. Антипина А. В., Пашков В. Н. Метод предотвращения DDoS атак на контроллер в программно-конфигурируемых сетях. Программные системы и инструменты. Тематический сборник № 19 / Под ред. В. В. Балашов, А. Г. Бахмуров, Е. И. Большакова и др. Москва: ООО МАКС Пресс, 2019, стр. 6–17.

[9]. Kumar V., Sangwan O. P. Signature based intrusion detection system using SNORT. International Journal of Computer Applications & Information Technology, 2012, vol. 1, issue 3, pp. 35-41. DOI: 10.32604/jiot.2022.039271.

[10]. Степанюк О. М., Подшибякин А. С. Подход к раннему обнаружению ddos-атак с использованием интеграции инструмента SNORT с SDN-контроллерами. Известия Тульского государственного университета. Технические науки, 2024, № 7, стр. 259-266.

[11]. OpenDaylight, available at: https://www.opendaylight.org/, accessed 10.11.2025.

[12]. ONOS Project, available at: https://wiki.onosproject.org/, accessed 10.11.2025.

[13]. Лейкин А. Протоколы транспортного уровня UDP, TCP и SCTP: достоинства и недостатки. Первая миля, 2013, т. 38, № 5, стр. 62-69. Доступно по ссылке: https://elibrary.ru/download/elibrary_20452615_45493752.pdf.

[14]. Мигутина Е. А., Королькова Ю. Ю. Уязвимости протокола ICMP. Поколение будущего: взгляд молодых ученых-2020. Сборник трудов конференции, 2020, стр. 79-81. Доступно по ссылке: https://elibrary.ru/download/elibrary_44309380_95937478.pdf.

[15]. Богомолова Л. В. Классификация DDoS-атак и их реализация. Современные инновации, 2022, № 1 (41), стр. 51-53. Доступно по ссылке: https://cyberleninka.ru/article/n/klassifikatsiya-ddos-atak-i-ih-realizatsiya, дата обращения 08.12.2025.

[16]. Волков С. С., Курочкин И. И. Применение методов машинного обучения в SDN в задачах обнаружения вторжений. International Journal of Open Information Technologies, 2019, т. 7, № 11, стр. 49-58. Доступно по ссылке: https://elibrary.ru/download/elibrary_41321761_57467477.pdf, дата обращения 08.12.2025.

[17]. Лапина М. А. и др. Исследование методов машинного обучения для обнаружения сайтов-мошенников. Известия ЮФУ. Технические науки, 2025, № 4, стр. 250-262. DOI: 10.18522/2311-3103-2025-4-250-262.

[18]. Лапина М.А., Капшук Н.Р., Русанов М.А., Тимофеева Е.Ф. Обнаружение атак с использованием SQL-инъекций по сетевым журналам с помощью методов машинного обучения. Труды Института системного программирования РАН, т. 37, вып. 5, 2025, стр. 81-92. DOI: 10.15514/ISPRAS-2025-37(5)-6. / Lapina M.A., Kapshuk N.R., Rusanov M.A., Timofeeva E.F. Detection of SQL Injection Attacks through the Network Logs Using Machine Learning Methods. Trudy ISP RAN/Proc. ISP RAS, 2025, vol. 37, issue 5, pp. 81-92 (in Russian):81-92. DOI: 10.15514/ISPRAS-2025-37(5)-6.

[19]. Лапина М. А. и др. Исследование методов машинного обучения спуфинг-атак в децентрализованых сетях. Известия ЮФУ. Технические науки, 2025, № 3, стр. 16-31. DOI: 10.18522/2311-3103-2025-3-16-31.
[20]. Lapina M.A., Podruchny N.V., Rusanov M.A., Babenko M.G. Research of machine learning methods for detecting network attacks. Trudy ISP RAN/Proc. ISP RAS. vol. 37, issue 4, part 2, 2025, pp. 147-174. DOI: 10.15514/ISPRAS-2025-37(4)-24.
[21]. Kaggle. DDoS SDN Dataset. Available at: URL: https://www.kaggle.com/datasets/chiragchiku25/ddos-sdn-dataset, accessed 15.08.2025.
[22]. Sasaki Y. The truth of the F-measure. Teach tutor mater, 2007, vol. 1, No. 5, pp. 1-5. Available at: https://www.researchgate.net/publication/268185911_The_truth_of_the_F-measure, accessed 08.12.2025.
[23]. Михайличенко А. А. Аналитический обзор методов оценки качества алгоритмов классификации в задачах машинного обучения. Вестник Адыгейского государственного университета. Серия 4: Естественно-математические и технические науки, 2022, № 4 (311), стр. 52-59. Доступно по ссылке: https://www.elibrary.ru/download/elibrary_50355429_49480416.pdf. DOI: 10.53598/2410-3225-2022-4-311-52-59.
[24]. Batista G. et al. How k-nearest neighbor parameters affect its performance. Argentine Symposium on Artificial Intelligence. Princeton, NJ, USA: Citeseer, 2009, pp. 1-12.
[25]. Rigatti S. J. Random forest. Journal of Insurance Medicine, 2017, vol. 47, No. 1, pp. 31-39.
[26]. Ходашинский И. А., Дель В. А., Анфилофьев А. Е. Выявление вредоносного сетевого трафика на основе ансамблей деревьев решений. Доклады Томского государственного университета систем управления и радиоэлектроники, 2014, No. 2 (32), pp. 202-206.
[27]. Zhang Z., Jung C. GBDT-MO: Gradient-boosted decision trees for multiple outputs. IEEE transactions on neural networks and learning systems, 2020, vol. 32, No. 7, pp. 3156-3167.
[28]. Zhang Z. Introduction to machine learning: k-nearest neighbors. Annals of translational medicine, 2016, vol. 4, No. 11, pp. 218.
[29]. Феоктистов А. Г. Вероятностная нейронная сеть классификации вычислительных заданий", Прогрессивные научные исследования-основа современной инновационной доктрины: сборник статей Международной научно-практической конференции (г. Киров, РФ, 25 ноября 2022 г.), Уфа: Аэтерна. Прогрессивные научные исследования – основа современной инновационной доктрины, 2022, pp. 131-135.
[30]. Манжула В. Г., Федяшов Д. С. Нейронные сети Кохонена и нечеткие нейронные сети в интеллектуальном анализе данных. Фундаментальные исследования, 2011, т. 4, стр. 108-114.
[31]. Кафтанников И. Л., Парасич А. В. Особенности применения деревьев решений в задачах классификации. Вестник Южно-Уральского государственного университета. Серия: Компьютерные технологии, управление, радиоэлектроника, 2015, т. 15, № 3, стр. 26-32.
[32]. Leung K. M. et al. Naive bayesian classifier. Polytechnic University Department of Computer Science/Finance and Risk Engineering, 2007, vol. 2007, pp. 123-156.
[33]. LaValley M. P. Logistic regression. Circulation, 2008, vol. 117, No. 18, pp. 2395-2399.
[34]. Geeks.ForGeeks. K-Nearest Neighbours. Доступно по ссылке: https://www.geeksforgeeks.org/machine-learning/k-nearest-neighbours/, accessed 13.12.2025.
[35]. Глинский И. В. Искусственные нейронные сети на основе радиально-базисных функций. Белорусский государственный университет информатики и радиоэлектроники. 59-я научная конференция аспирантов, магистрантов и студентов БГУИР, Минск, 2023. Доступно по ссылке: https://libeldoc.bsuir.by/bitstream/123456789/52761/1/Glinskii_Iskusstvennie.pdf, обращение 13.12.2025.
[36]. В.А. Парасич, И.В. Парасич, Г.И. Волович и др. Переобучение в машинном обучении: проблемы и решения. Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника», 2024, т. 24, № 2, стр. 18–27. DOI: 10.14529/ctcr240202.

Информация об авторах / Information about authors

Мария Анатольевна ЛАПИНА – кандидат физико-математических наук, доцент кафедры вычислительно математики и кибернетики Северо-Кавказского федерального университета. Сфера научных интересов: цифровые технологии, управление информационной безопасностью, процессный подход, криптография.

Maria Anatolyevna LAPINA – Cand. Sci. (Phys.-Math.), Associate Professor at the Department of Computational Mathematics and Cybernetics at the North Caucasus Federal University. Research interests: digital technologies, information security management, process approach, and cryptography.

Даниил Дмитриевич ГРИЦЕНКО – студент Северо-Кавказского Федерального университета. Сфера научных интересов: криптография, машинное обучение, цифровые технологии, управление информационной безопасностью, процессный подход, образовательный процесс.

Daniil Dmitrievich GRITSENKO – Student of the North Caucasus Federal University. Research interests: cryptography, machine learning, digital technologies, information security management, process approach, and educational process.

Егор Александрович КЕНЬКОВ – аспирант Северо-Кавказского Федерального университета. Сфера научных интересов: комплексные системы защиты информации, информационно-коммуникационные технологии.

Egor Alexandrovich KENKOV – Postgraduate student of the North Caucasus Federal University. Research interests: complex information protection systems, Information and Communication Technologies.

Алина Раисовна БАГАУТДИНОВА – студент Северо-Кавказского Федерального университета. Сфера ее научных интересов: криптография, машинное обучение, цифровые технологии, управление информационной безопасностью, процессный подход, образовательный процесс.

Alina Raisovna BAGAUTDINOVA – Student of the North Caucasus Federal University. Her research interests: cryptography, machine learning, digital technologies, information security management, process approach, and educational process.

Артем Алексеевич СОЛОМЯНКО – студент Северо-Кавказского Федерального университета. Сфера научных интересов: криптография, машинное обучение, цифровые технологии, управление информационной безопасностью, процессный подход, образовательный процесс.

Artem Alekseevich SOLOMYANKO – Student of the North Caucasus Federal University. His research interests: cryptography, machine learning, digital technologies, information security management, process approach, and educational process.