Система обнаружения вторжений в сеть на основе SDN с использованием подходов машинного обучения

М. М. Аль-Тамими[1], М. Б. Хассан[2], С. А. Аббас[3]

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина)

[1]almokhalad44@gmail.com, [2]moubarekbarrehassan@gmail.com, [3]saddamabbas077@gmail.com

Аннотация. **Данное исследование исследует интеграцию методов машинного обучения (ML) и глубокого обучения (DL) в архитектуры программно-определенных сетей (SDN) для систем обнаружения вторжений в сеть (NIDS). С увеличением частоты и сложности киберугроз становится все более критической необходимость в надежных возможностях обнаружения вторжений. Методологии ML и DL предлагают перспективу в улучшении эффективности NIDS, обеспечивая возможность превентивного обнаружения и реагирования на угрозы. В исследовании рассматриваются различные алгоритмы ML/DL, используемые в NIDS, такие как машины опорных векторов, случайные леса и сверточные нейронные сети, выявляя их сильные и слабые стороны. Кроме того, слияние SDN с методами ML/DL представляет собой многоаспектное преимущество, включая усиление соблюдения безопасности и улучшение качества обслуживания (QoS). Через всесторонний анализ последних достижений и вызовов в SDN-основанных NIDS, это исследование нацелено на вклад в понимание и продвижение систем обнаружения вторжений в изменяющемся ландшафте киберугроз и сетевых архитектур.**

Ключевые слова: системы обнаружения вторжений в сеть (NIDS); машинное обучение (ML; глубокое обучение (DL); программно-определенные сети (SDN); кибербезопасность

I. Введение

Все более частые и изощренные кибератаки, направленные на правительства и коммерческие предприятия по всему миру, способствовали быстрому развитию систем обнаружения вторжений в сеть (NIDS) как в научных кругах, так и в промышленности. Поскольку стоимость киберпреступлений постоянно растет [1], последствия вредоносных действий, включая те, которые инициированы инсайдерами, атаки типа «отказ в обслуживании» и веб-атаки, становятся все более серьезными. Такие угрозы не только нарушают целостность данных организации, но и представляют значительный риск для критической национальной инфраструктуры. Чтобы укрепить защиту от несанкционированного доступа и смягчить потенциальные ущербы от кибервзломов, организации применяют множество защитных мер, включая брандмауэры, антивирусное программное обеспечение и NIDS [2]. Первостепенной задачей в борьбе с кибератаками является раннее обнаружение вредоносной деятельности в сети [1], что позволяет оперативно реагировать и принимать меры по ее снижению. Системы NIDS специально разработаны для обнаружения и предотвращения различных форм вредоносной деятельности, таких как вирусы, черви и распределенные атаки типа «отказ в обслуживании» (DDoS).

Эффективность NIDS зависит от таких факторов, как скорость обнаружения, точность и надежность. В последние годы наблюдается заметная тенденция к использованию методов машинного обучения (ML) для расширения возможностей NIDS с целью повышения точности обнаружения и снижения числа ложных срабатываний [3][4]. Среди методик ML особое внимание привлекают подходы глубокого обучения (ГО), поскольку они способны эффективно бороться со сложными и эволюционирующими киберугрозами. Более того, интеграция методов ML и DL с новыми сетевыми архитектурами, такими как программно-определяемые сети (SDN), открывает перспективы для расширения возможностей NIDS [5].

SDN представляет собой новую сетевую архитектуру, в которой разделены функции управления и пересылки данных, что позволяет программировать управление сетью. Отделяя плоскость управления от плоскости данных, SDN способствует более гибкому и динамичному управлению сетью, позволяя эффективно адаптироваться к изменяющимся условиям сети и требованиям безопасности. Синергия между SDN и NIDS на основе ML открывает возможности для усиления функций обнаружения вторжений, позволяя проактивно обнаруживать угрозы и реагировать на них во все более динамичных и сложных сетевых средах. В свете этих событий в данном обзоре предпринята попытка всесторонне изучить системы обнаружения вторжений в сеть на базе SDN, использующие подходы машинного обучения. Благодаря глубокому анализу последних достижений, проблем и будущих перспектив этот обзор призван внести вклад в понимание и развитие систем обнаружения вторжений в условиях меняющегося ландшафта киберугроз и сетевых архитектур.

II. Система обнаружения вторжений в сеть (NIDS)

В области сетевой безопасности системы обнаружения вторжений (Intrusion Detection Systems,

IDS) являются ключевыми компонентами, предназначенными для выявления и противодействия угрозам, встроенным в сетевой трафик. IDS разработаны для обнаружения аномальных и вредоносных действий, исходящих из внутренних и внешних источников [8]. Однако IDS сталкиваются с серьезными проблемами, в частности с управлением огромными объемами сетевого трафика и перекосами в распределении данных. Несмотря на эти препятствия, основная задача IDS остается неизменной: бдительно отслеживать информационные каналы в сети, охватывающие компьютеры и сетевые устройства, в поисках попыток несанкционированного доступа. Благодаря тщательному сбору и анализу данных, IDS стремятся выявить потенциальные угрозы и укрепить защиту сети [8].

Эволюция IDS привела к появлению специализированных итераций, включая сетевые системы обнаружения вторжений (NIDS) и системы обнаружения вторжений на базе хоста (HIDS). NIDS предназначены для мониторинга потоков сетевого трафика в реальном времени, в то время как HIDS сосредоточены на отдельных хост-устройствах, тщательно изучая их на предмет подозрительного поведения. Концептуальный ландшафт IDS представлен на рис. 1, иллюстрирующем разнообразие методов обнаружения и сценариев развертывания. Реализации IDS охватывают целый ряд методологий, начиная от статистических подходов и методов интеллектуального анализа данных и заканчивая алгоритмами на основе машинного обучения [9].

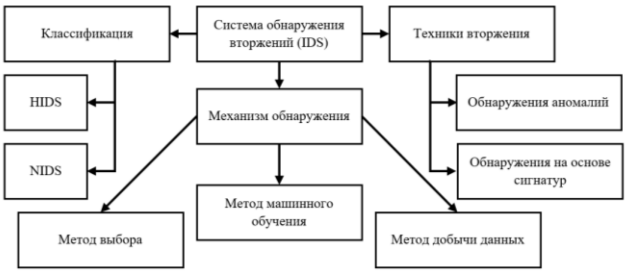

Рис. 1. Обзор вторжения

В сфере NIDS преобладают две основные методологии обнаружения: обнаружение на основе сигнатур и обнаружение на основе аномалий [25]. NIDS на основе сигнатур опираются на заранее определенные шаблоны или сигнатуры для выявления известных вредоносных угроз, в то время как NIDS на основе аномалий используют статистический анализ или алгоритмы машинного обучения для обнаружения отклонений от установленных норм, тем самым отмечая потенциальные угрозы, которые ускользают от обнаружения на основе сигнатур [9].

Методы машинного обучения являются олицетворением обнаружения вторжений на основе аномалий, предлагая адаптивные возможности для выявления тонких закономерностей, указывающих на аномальное поведение. Обычно используемые критерии оценки, такие как точность, частота ложноотрицательных (FNR) и ложноположительных (FPR) ошибок, служат в качестве эталонов для оценки эффективности алгоритмов в NIDS. Эти показатели играют ключевую роль в определении эффективности и надежности систем обнаружения вторжений [25].

Сравнительный анализ, хотя и не представлен в данном разделе, позволяет оценить различные методы обнаружения на основе различных критериев эффективности, тем самым направляя выбор и оптимизацию решений NIDS. В этом разделе представлен обзор современных алгоритмов машинного обучения, используемых в NIDS, проливающий свет на их эффективность, преимущества и ограничения в обнаружении и защите от сетевых угроз. Кроме того, в табл. 1 представлено сравнение трех методов обнаружения, основанных на различных критериях эффективности для NIDS, что поможет в процессе оценки и выбора.

III. Машинное обучение в системах обнаружения вторжений в сеть

Область машинного обучения (ML) посвящена разработке систем, которые могут автоматически обучаться на основе данных и выявлять скрытые закономерности, не будучи явно запрограммированными на это [8]. Алгоритмы ML классифицируются по стилю обучения, который они используют, и по функциональному сходству их работы [8]. Методы машинного обучения рассматриваются как эффективные способы повышения частоты обнаружения, снижения частоты ложных тревог и, в то же время, уменьшения вычислительных и коммуникационных затрат [10]. Подходы к машинному обучению, как показано на рис. 2, можно разделить на контролируемое, неконтролируемое и полу-контролируемое обучение [2].

A. Контролируемое обучение

В контролируемом обучении алгоритмы анализируют помеченные входные данные, чтобы изучить их представления и способность предсказывать неизвестные случаи. Примерами алгоритмов контролируемого машинного обучения являются метод опорных векторов (Support Vector Machine, SVM) для задач классификации и метод Random Forest для задач классификации и регрессии. Алгоритмы метода опорных векторов (SVM) широко применяются в исследованиях NIDS из-за их сильной классификационной способности и практичности в вычислениях. Они подходят для работы с высоко размерными данными, однако выбор подходящей функции ядра имеет решающее значение. Эти алгоритмы требовательны к ресурсам, включая вычислительные устройства и память [8]. Алгоритм Random Forest [11] представляет собой мощный метод ансамблевого контролируемого обучения, который позволяет эффективно работать с неоднородными данными, но подвержен перестройке.

ТАБЛИЦА I. СРАВНЕНИЕ МЕТОДОВ ОБНАРУЖЕНИЯ

Метод обнаружения	Скорость тревог	Скорость	Гибкость	Надежность	Масштабируемость	Прочность
На основе сигнатур	Медленная	Быстрая	Низкая	Высокая	Средняя	Высокая
На основе аномалий	Высокая	Медленная	Высокая	Средняя	Высокая	Средняя
На основе машинного обучения	Средняя	Средняя	Высокая	Высокая	Высокая	Высокая

B. Неконтролируемое обучение

В методологии обучения без контроля алгоритмы изучают структуру и представления на основе немаркированных входных данных. Целью алгоритма обучения без контроля является моделирование основной структуры или распределения данных с целью предсказания неизвестных данных. Примерами алгоритмов обучения без контроля являются методы сокращения признаков, такие как анализ главных компонент (PCA), и методы кластеризации, например, K-means и самоорганизующиеся карты (SOM).

Алгоритм анализа главных компонент (PCA) применяется для значительного ускорения неконтролируемого обучения признаков [24]. Многие исследователи используют PCA для выбора признаков перед применением классификации [12]. Для обнаружения аномалий часто используются алгоритмы кластеризации, такие как K-means и другие алгоритмы обучения на основе расстояний. Самоорганизующаяся карта (SOM) – это вид искусственной нейронной сети, который применяется для снижения полезной нагрузки в NIDS [13]. Однако недостатком использования алгоритмов кластеризации для обнаружения аномалий является их чувствительность к начальным условиям, таким как выбор центроидов, что может привести к высокой частоте ложных срабатываний [14].

C. Полу-контролируемое обучение

это тип контролируемого обучения, при котором для тренировки используются немаркированные данные. Обучающий набор данных состоит из небольшого количества помеченных данных и большого количества непомеченных данных. Оно применяется в случаях, когда большие объемы помеченных данных недоступны, например, в фотоархивах, где помечены лишь некоторые изображения (например, люди), а большинство не помечены [15]. Для повышения точности систем обнаружения вторжений в сети (NIDS) была использована полуконтролируемая машина опорных векторов [16] [17].

Два метода классификации с применением полуконтролируемого обучения – спектральное преобразование графов и подход гауссовых полей – использовались для обнаружения неизвестных атак, а метод кластеризации MPCK-means – для повышения производительности системы обнаружения [18].

Глубокое обучение – это современное обновление искусственных нейронных сетей, использующее доступные вычислительные ресурсы. Оно позволяет алгоритмам изучать представления данных на различных уровнях абстракции. Эти методы применяются в таких областях, как визуальное распознавание объектов, обнаружение объектов, обнаружение сетевых вторжений и многих других. Алгоритмы глубокого обучения могут быть обучены как контролируемым, так и неконтролируемым способом.

Алгоритм глубокого обучения под наблюдением

Конволюционная нейронная сеть (CNN) [19] обычно обучается контролируемым способом. В настоящее время CNN является эталонной моделью для компьютерного зрения. Архитектура CNN используется для анализа двумерных изображений, а наиболее важным применением CNN является распознавание лиц [19].

Алгоритм глубокого обучения без контроля

Авто-кодировщик используется для изучения представлений (кодирования) набора данных с целью снижения размерности. Сеть глубокого убеждения (DBN) [20] способна восстанавливать свои входы в процессе обучения на основе набора примеров без контроля. Затем слои действуют как детекторы признаков на входах. После завершения этапа обучения DBN контролируемым способом приступают к выполнению задачи классификации. DBN, такие как ограниченные машины Больцмана (RBM) или автокодировщики, применяются для сокращения размерности, регрессии, совместной фильтрации, изучения признаков, моделирования тем и других задач.

Алгоритм глубокого обучения в контролируемом или неконтролируемом режиме

Алгоритмы рекуррентных нейронных сетей (RNN) рассматриваются как методы контролируемого или неконтролируемого обучения. Рекуррентные нейронные сети могут использовать внутреннюю память для обработки входных данных в произвольном порядке. Распознавание речи – типичное приложение для RNN. RNN хорошо предсказывает символы в тексте и способен изучать зависимости и фактические данные, которые сохраняются в течение продолжительного времени [19].

Внедрение этих передовых методов ML в NIDS позволяет организациям активно защищаться от появляющихся киберугроз, обеспечивая обнаружение угроз и реагирование на них в режиме реального времени. По мере развития ML интеграция инновационных алгоритмов с NIDS обещает улучшить безопасность и устойчивость сетей в условиях постоянно изменяющегося цифрового ландшафта.

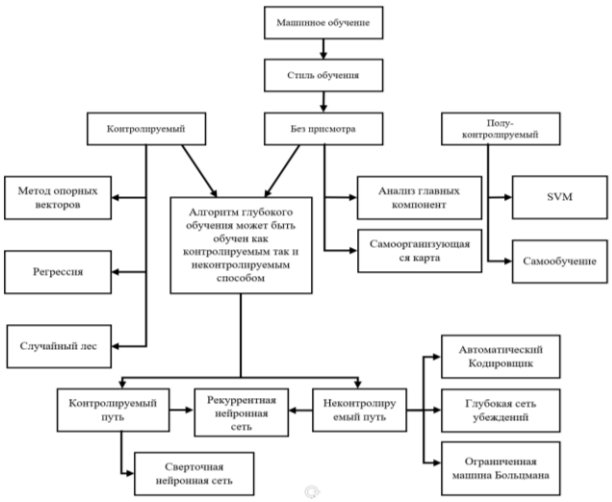

Рис. 2. Обзор подходов машинного обучения

IV. НАБЛЮДЕНИЕ ЗА NIDS НА ОСНОВЕ ML

Методы машинного обучения (ML) и глубокого обучения (DL) играют ключевую роль в разработке систем обнаружения вторжений в сети (NIDS). Различные алгоритмы ML/DL, такие как искусственные нейронные сети (ANN), машины опорных векторов (SVM), наивные байесовские классификаторы (NB), случайные леса (RF) и самоорганизующиеся карты (SOM), широко используются для повышения эффективности NIDS [12]. Например, в [21] была реализована система RBM-SVM с точностью 87 %. В [23] интегрировали дискриминативные RBM с генеративными моделями для надежной классификации, а в [11] провели оценку древовидных алгоритмов на наборе данных NSL-KDD. Кроме того, в [25] был оптимизирован выбор признаков с помощью PCA и SVM, а в [12] представлен гибкий фреймворк NIDS, включающий самообучение и регрессию softmax, что позволило достичь точности 92,48 %. Следует отметить, что в [12] использовались отдельные наборы данных для обучения и тестирования, что обеспечило тщательную оценку, в то время как в [23] наблюдалось снижение производительности при использовании различных обучающих данных. Кроме того, в [11] подчеркивается эффективность моделей случайных деревьев. Эти результаты подчеркивают преобразующее воздействие методологий ML/DL на NIDS, укрепляя защиту кибербезопасности. Продолжение исследований крайне важно для борьбы с возникающими угрозами и защиты критической инфраструктуры. В табл. 2 приведены результаты исследований [22], [23], [11], [25] и [12].

ТАБЛИЦА II. ML/DL TECHNIQUES IN NIDS DEVELOPMENT

Study	ML/DL Approach	Key Findings	Accuracy
[22]	RBM + SVM	Achieved 87% accuracy	~87%
[23]	Discriminative RBM + Generative Models	Strong classification accuracy	N/A
[11]	Decision Tree + Random Forest	High accuracy and low false alarm rate	N/A
[25]	PCA + SVM	Optimal feature subset identification	N/A
[12]	Self-taught Learning + Sparse Encoder + Softmax Regression	92.48% accuracy on training data	92.48%

V. NIDS НА БАЗЕ ПРОГРАММНО-ОПРЕДЕЛЯЕМЫХ СЕТЕЙ (SDN)

Программно-определяемые сети (SDN) произвели революцию в сетевых архитектурах, внедрив смену парадигмы, которая отделяет плоскость управления от плоскости данных, упрощая процессы пересылки пакетов [1]. В основе SDN лежит централизованный контроллер, наделенный возможностями управления с обратной связью в реальном времени [34] и открытыми интерфейсами, которые позволяют подключать модульные функции. Этот централизованный контроллер обеспечивает абстрактное представление сети, разграничивая задачи через API и обеспечивая большую программируемость сети [7]. Кроме того, он способен легко интегрировать устройства безопасности в сетевую топологию, обеспечивая повышенную точность обнаружения инцидентов безопасности и упрощая управление [26].

Архитектура SDN, разработанная Open Networking Foundation (ONF), включает три основных функциональных уровня: инфраструктурный, управляющий и прикладной. Инфраструктурный уровень, также известный как плоскость данных, состоит в основном из устройств пересылки (FE), включающих как физические коммутаторы, такие как Juniper и HP, так и виртуальные коммутаторы, такие как OpenvSwitch. Эти устройства соединены между собой проводными или беспроводными средствами связи. Над уровнем инфраструктуры расположен уровень управления, также называемый плоскостью управления. Этот уровень включает набор программных SDN-контроллеров, которые управляют функциями управления через открытые API. Связь между контроллерами и сетевым оборудованием осуществляется с помощью южных API, в то время как северные интерфейсы SDN (NBI) обеспечивают связь между приложениями SDN и уровнем управления, предоставляя всестороннюю информацию о сети. Интерфейсы с востока на запад служат для обеспечения связи между контроллерами, чтобы расширить возможности управления внутри домена. Наконец, прикладной уровень включает в себя бизнес-приложения для конечных пользователей, включая приложения для мониторинга сети и обеспечения безопасности. Используя расширенные возможности SDN, было разработано множество приложений SDN для повышения гибкости сети, сокращения времени выхода на рынок и снижения совокупной стоимости владения будущими сетевыми ИТ-инфраструктурами [27]. В частности, SDN нашла

применение в различных сетевых доменах благодаря своей способности быстро разрабатывать и внедрять новые услуги в условиях эскалации киберугроз. Несколько ключевых приложений SDN описаны ниже [30], Беспроводные коммуникации: Программируемость SDN расширяет сети мобильной связи, позволяя точно настраивать производительность мобильной связи. Применимость SDN распространяется на беспроводные сетевые среды, такие как сотовая связь, беспроводные ячеистые сети, сети доступа Wi-Fi и Интернет вещей (IoT). Используя SDN, парадигмы IoT могут достичь масштабируемости, упростить управление и организацию трафика в беспроводных ячеистых сетях, тем самым способствуя развитию сетевых соединений и совместному использованию пропускной способности. Центры обработки данных: В области центров обработки данных оркестровка трафика на базе SDN способствует оптимальному проектированию трафика, управлению сетью и реализации политик, особенно в крупномасштабных операциях. Такой оркестрованный подход помогает сократить задержки в сети и автоматизировать динамическое внедрение безопасности в центрах обработки данных.

Облако на базе SDN: Сочетание облачных технологий с парадигмой SDN позволяет создать тесно интегрированную среду облачных приложений. Благодаря сетевым программируемым интерфейсам и автоматизации SDN служит грозным инструментом в борьбе с вторжениями в облако и повышает масштабируемость сервисов в облачных средах.

VI. Наблюдение за NIDS на базе SDN с использованием ML/DL

В области сетевой безопасности интеграция программно-определяемых сетей (SDN) с системами обнаружения вторжений (IDS), использующими подходы машинного обучения (ML) и глубокого обучения (DL), привлекает значительное внимание благодаря своим многогранным преимуществам. В этом разделе критически рассматривается эффективность интеграции методологий ML и DL в рамках SDN для приложений NIDS, проливается свет на ключевые наблюдения и развивающиеся тенденции в этой области. Объединение SDN с методами ML/DL дает множество преимуществ, включая усиление контроля безопасности, расширение возможностей управления виртуальными сетями и повышение качества обслуживания (QoS) [8].

Использование SDN позволяет организациям повысить уровень безопасности сети и одновременно получить гибкость в программировании сетевых устройств и уменьшить зависимость от аппаратного обеспечения, тем самым повышая устойчивость сети к возникающим киберугрозам. Важнейшим аспектом развертывания NIDS на базе SDN является тщательный выбор и сравнение различных решений, доступных в экосистеме SDN. В табл. 2 представлен краткий обзор и сравнительный анализ различных решений NIDS, использующих платформу SDN, что способствует принятию взвешенных решений при реализации сетевой безопасности.

Развитие сетей SDN с программными коммутаторами и программируемыми функциями облегчается благодаря платформам моделирования и эмуляции, а такие протоколы, как OpenFlow, служат стандартными носителями для реализации концепций SDN в гетерогенных аппаратных и программных средах. Кроме того, такие инструменты моделирования, как NS-2, Mininet, NS-3 и OMNeT++, предлагают альтернативные пути развития сетей SDN, удовлетворяющие различным требованиям и сценариям [28, 29]. В основе архитектуры NIDS на базе SDN лежит контроллер SDN, который организует взаимодействие с программируемыми сетевыми элементами. Такие известные SDN-контроллеры, как NOX и POX, играют ключевую роль в консолидации коммуникаций и обеспечении единого представления о работе сети, тем самым способствуя эффективной работе функций обнаружения вторжений [31].

ТАБЛИЦА III. NIDS НА БАЗЕ SDN С ИСПОЛЬЗОВАНИЕМ ПОДХОДА ГЛУБОКОГО ОБУЧЕНИЯ

Публикация	Метод	Использование	Сравнение
Rodrigo Braga et al. [29]	Используемые самоорганизующиеся карты неконтролируемой искусственной нейронной сети	Легкий DDoS-Атаки Атака Наводнения	Эффективен при обнаружении DDoS атак но не установлены никакие правила потока
Quamar Niyaz et al. [6]	Используется стековый автоэнкодер, глубокое обучение для уменьшения характеристик	Система Обнаружения DDoS-Атак	Может обнаружить любую DDoS-атаку, но имеет узкое место контроллера в обширной сети
Damian Jankowski et al. [32]	Используется самоорганизующаяся карта и обучение векторное квантование.	Обнаружение Вторжений	Может обнаруживать атаки U2R, которые включают техника глубокой проверки пакетов
Tuan A Tang et al. [5]	Используется самоорганизующаяся карта и обучение векторное квантование.	Обнаружение Аномалий	Плохо масштабируется коммерческий продукт или альтернативное решение для идентификаторов на основе сигнатур

Одним из заметных трендов в области NIDS на базе SDN является увеличение использования методов глубокого обучения в сочетании с традиционными методами машинного обучения. Глубокое обучение, известное своей способностью автономно выявлять корреляции в данных, обещает продвижение техник обнаружения вторжений нового поколения [8]. Особенно заметным является превосходное качество работы подходов на основе глубокого обучения по сравнению с традиционными методами машинного обучения, особенно в задачах логического моделирования в сетях SDN [8]. Учитывая динамичный и развивающийся характер киберугроз, алгоритмы неконтролируемого обучения, такие как стековые автокодировщики, Рекуррентные Нейронные Сети (RNN) и гибридные алгоритмы, выступают в качестве оптимальных выборов для реализации NIDS в рамках SDN, предлагая устойчивость и адаптивность для обнаружения и

смягчения неизвестных и сложных атак, тем самым укрепляя безопасность сети в средах SDN. Недавние исследования подчеркнули эффективность NIDS на основе машинного обучения в сетях малых офисов/домашних офисов (SOHO) в рамках сред SDN.

Интеграция алгоритмов на основе машинного обучения с масштабируемостью SDN привела к значительному улучшению точности систем обнаружения вторжений, подчеркивая ключевую роль методов машинного и глубокого обучения в укреплении безопасности сети в современной сетевой инфраструктуре [8].

VII. Заключение

Интеграция методологий машинного обучения (ML) и глубокого обучения (DL) в архитектуры программно-определяемых сетей (SDN) для систем обнаружения вторжений в сеть (NIDS) обещает значительно усилить безопасность сети перед постоянно меняющимися киберугрозами. В ходе этого обзора мы рассмотрели эффективность различных алгоритмов ML/DL, таких как машины опорных векторов, случайные леса и сверточные нейронные сети, в дополнение к возможностям NIDS. Кроме того, мы проанализировали слияние SDN с методами ML/DL, разъяснив многоаспектные преимущества, включая усиление обеспечения безопасности и улучшение качества обслуживания (QoS). Наше исследование последних достижений и вызовов подчеркивает трансформационное воздействие интеграции методологий ML и DL в рамках архитектуры SDN, открывая путь к более устойчивым и адаптивным системам обнаружения вторжений. Поскольку киберугрозы продолжают эволюционировать, использование возможностей SDN наряду с продвинутыми методами ML/DL становится необходимым для обеспечения целостности и устойчивости сети в современной сетевой инфраструктуре.

Список литературы

[1] Kreutz, Diego, Fernando MV Ramos, Paulo Esteves Verissimo, Christian Esteve Rothenberg, Siamak Azodolmolky, and Steve Uhlig. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103, no. 1 (2014): 14-76. DOI: 10.1109/JPROC.2014.2371999

[2] Aburomman, Abdulla Amin, and Mamun Bin Ibne Reaz. "Survey of learning methods in intrusion detection systems." In 2016 international conference on advances in electrical, electronic and systems engineering (ICAEES), pp. 362-365. IEEE, 2016. DOI: 10.1109/ICAEES.2016.7888070

[3] Mehdi, Syed Akbar, Junaid Khalid, and Syed Ali Khayam. "Revisiting traffic anomaly detection using software defined networking." In Recent Advances in Intrusion Detection: 14th International Symposium, RAID 2011, Menlo Park, CA, USA, September 20-21, 2011. Proceedings 14, pp. 161-180. Springer Berlin Heidelberg, 2011. https://doi.org/10.1007/978-3-642-23644-0_9

[4] Garcia-Teodoro, Pedro, Jesus Diaz-Verdejo, Gabriel Maciá-Fernández, and Enrique Vázquez. "Anomaly-based network intrusion detection: Techniques, systems and challenges." computers & security 28, no. 1-2 (2009): 18-28. https://doi.org/10.1016/j.cose.2008.08.003

[5] Tang, Tuan A., Lotfi Mhamdi, Des McLernon, Syed Ali Raza Zaidi, and Mounir Ghogho. "Deep learning approach for network intrusion detection in software defined networking." In 2016 international conference on wireless networks and mobile communications (WINCOM), pp. 258-263. IEEE, 2016. DOI: 10.1109/WINCOM.2016.7777224

[6] Niyaz, Quamar, Weiqing Sun, and Ahmad Y. Javaid. "A deep learning based DDoS detection system in software-defined networking (SDN)." arXiv preprint arXiv:1611.07400 (2016). https://doi.org/10.48550/arXiv.1611.07400

[7] Sezer, Sakir, Sandra Scott-Hayward, Pushpinder Kaur Chouhan, Barbara Fraser, David Lake, Jim Finnegan, Niel Viljoen, Marc Miller, and Navneet Rao. "Are we ready for SDN? Implementation challenges for software-defined networks." IEEE Communications magazine 51, no. 7 (2013): 36-43. DOI: 10.1109/MCOM.2013.6553676

[8] Hodo, Elike, Xavier Bellekens, Andrew Hamilton, Christos Tachtatzis, and Robert Atkinson. "Shallow and deep networks intrusion detection system: A taxonomy and survey." arXiv preprint arXiv:1701.02145 (2017). https://doi.org/10.48550/arXiv.1701.02145

[9] Kumar, Sailesh. "Survey of current network intrusion detection techniques." Washington Univ. in St. Louis (2007): 1-18. https://www.cse.wustl.edu/~jain/cse571-07/ftp/ids/

[10] Zamani, Mahdi, and Mahnush Movahedi. "Machine learning techniques for intrusion detection." arXiv preprint arXiv:1312.2177 (2013). https://doi.org/10.48550/arXiv.1312.2177

[11] Thaseen, Sumaiya, and Ch Aswani Kumar. "An analysis of supervised tree based classifiers for intrusion detection system." In 2013 international conference on pattern recognition, informatics and Mobile engineering, pp. 294-299. IEEE, 2013. DOI: 10.1109/ICPRIME.2013.6496489

[12] Javaid, Ahmad, Quamar Niyaz, Weiqing Sun, and Mansoor Alam. "A deep learning approach for network intrusion detection system." In Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS), pp. 21-26. 2016. https://doi.org/10.4108/eai.3-12-2015.2262516

[13] Zanero, Stefano, and Sergio M. Savaresi. "Unsupervised learning techniques for an intrusion detection system." In Proceedings of the 2004 ACM symposium on Applied computing, pp. 412-419. 2004. https://doi.org/10.1145/967900.967988

[14] Syarif, Iwan, Adam Prugel-Bennett, and Gary Wills. "Unsupervised clustering approach for network anomaly detection." In Networked Digital Technologies: 4th International Conference, NDT 2012, Dubai, UAE, April 24-26, 2012. Proceedings, Part I 4, pp. 135-145. Springer Berlin Heidelberg, 2012. https://doi.org/10.1007/978-3-642-30507-8_13

[15] Tsai, Chih-Fong, Yu-Feng Hsu, Chia-Ying Lin, and Wei-Yang Lin. "Intrusion detection by machine learning: A review." expert systems with applications 36, no. 10 (2009): 11994-12000. https://doi.org/10.1016/j.eswa.2009.05.029

[16] Ding, Shifei, Zhibin Zhu, and Xiekai Zhang. "An overview on semi-supervised support vector machine." Neural Computing and Applications 28 (2017): 969-978. https://doi.org/10.1007/s00521-015-2113-7

[17] Haweliya, Jyoti, and Bhawna Nigam. "Network intrusion detection using semi supervised support vector machine." International Journal of Computer Applications 85, no. 9 (2014). https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=cbdd844a44b25e97e16a2d8804d588bd07c14f41

[18] Chen, Chuanliang, Yunchao Gong, and Yingjie Tian. "Semi-supervised learning methods for network intrusion detection." In 2008 IEEE international conference on systems, man and cybernetics, pp. 2603-2608. IEEE, 2008. DOI: 10.1109/ICSMC.2008.4811688

[19] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." nature 521, no. 7553 (2015): 436-444. https://doi.org/10.1038/nature14539

[20] Alom, Md Zahangir, VenkataRamesh Bontupalli, and Tarek M. Taha. "Intrusion detection using deep belief networks." In 2015 National Aerospace and Electronics Conference (NAECON), pp. 339-344. IEEE, 2015. DOI: 10.1109/NAECON.2015.7443094

[21] Hughes, Thad, and Keir Mierle. "Recurrent neural networks for voice activity detection." In 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 7378-7382. IEEE, 2013. DOI: 10.1109/ICASSP.2013.6639096

[22] Salama, Mostafa A., Heba F. Eid, Rabie A. Ramadan, Ashraf Darwish, and Aboul Ella Hassanien. "Hybrid intelligent intrusion

detection scheme." In Soft computing in industrial applications, pp. 293-303. Springer Berlin Heidelberg, 2011. https://doi.org/10.1007/978-3-642-20505-7_26

[23] Fiore, Ugo, Francesco Palmieri, Aniello Castiglione, and Alfredo De Santis. "Network anomaly detection with the restricted Boltzmann machine." Neurocomputing 122 (2013): 13-23. https://doi.org/10.1016/j.neucom.2012.11.050

[24] Heba, F. Eid, Ashraf Darwish, Aboul Ella Hassanien, and Ajith Abraham. "Principle components analysis and support vector machine based intrusion detection system." In 2010 10th international conference on intelligent systems design and applications, pp. 363-367. IEEE, 2010. DOI: 10.1109/ISDA.2010.5687239

[25] Wang, Lidong, and Randy Jones. "Big data analytics for network intrusion detection: A survey." International Journal of Networks and communications 7, no. 1 (2017): 24-31.

[26] Nunes, Bruno Astuto A., Marc Mendonca, Xuan-Nam Nguyen, Katia Obraczka, and Thierry Turletti. "A survey of software-defined networking: Past, present, and future of programmable networks." IEEE Communications surveys & tutorials 16, no. 3 (2014): 1617-1634. DOI: 10.1109/SURV.2014.012214.00180

[27] Bakhshi, Taimur. "State of the art and recent research advances in software defined networking." Wireless Communications and Mobile Computing 2017 (2017). https://doi.org/10.1155/2017/7191647

[28] Yan, Qiao, F. Richard Yu, Qingxiang Gong, and Jianqiang Li. "Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges." IEEE communications surveys & tutorials 18, no. 1 (2015): 602-622. DOI: 10.1109/COMST.2015.2487361

[29] Braga, Rodrigo, Edjard Mota, and Alexandre Passito. "Lightweight DDoS flooding attack detection using NOX/OpenFlow." In IEEE local computer network conference, pp. 408-415. IEEE, 2010. DOI: 10.1109/LCN.2010.5735752

[30] Prete, Ligia Rodrigues, Ailton Akira Shinoda, Christiane Marie Schweitzer, and Rogerio Leao Santos De Oliveira. "Simulation in an SDN network scenario using the POX Controller." In 2014 IEEE Colombian Conference on Communications and Computing (COLCOM), pp. 1-6. Ieee, 2014. DOI: 10.1109/ColComCon.2014.6860403

[31] Kaur, Sukhveer, Japinder Singh, and Navtej Singh Ghumman. "Network programmability using POX controller." In ICCCS International conference on communication, computing & systems, IEEE, vol. 138, p. 70. sn, 2014. DOI:10.13140/RG.2.1.1950.6961

[32] Jankowski, Damian, and Marek Amanowicz. "On efficiency of selected machine learning algorithms for intrusion detection in software defined networks." International Journal of Electronics and Telecommunications 62, no. 3 (2016). DOI:10.1515/eletel-2016-0033