ВЕСТНИК
ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА
МОРСКОГО И РЕЧНОГО ФЛОТА ИМЕНИ АДМИРАЛА С. О. МАКАРОВА

АВТОМАТИЗАЦИЯ И УПРАВЛЕНИЕ ТЕХНОЛОГИЧЕСКИМИ ПРОЦЕССАМИ И ПРОИЗВОДСТВАМИ

DOI: 10.21821/2309-5180-2024-16-3-456-466

OPTIMIZATION OF THE PENETRATION TESTING PROCESS IN AUTOMATED PROCESS CONTROL SYSTEMS USING MACHINE LEARNING ALGORITHMS

A. P. Nyrkov, E. S. Yumasheva, A. V. Kirikov

Admiral Makarov State University of Maritime and Inland Shipping,
St. Petersburg, Russian Federation

The process of widespread implementation of automated information management systems in industry, energy and transport is studied in the paper. It is noted that an increase in their complexity inevitably leads to the emergence of various kinds of vulnerabilities in these systems, the presence of which allows attackers to penetrate automated control systems, take control of them, and also disrupt the normal operation of the technological processes they control. It is emphasized that over the past decade, successful cyber attacks have been recorded in the energy sector, including nuclear, in maritime shipping, in port transshipment complexes, as well as in other systems. A preventive approach to ensuring the security of automated control systems is to identify and exploit existing vulnerabilities by simulating possible cyber attacks. It is noted that automation of such a rather labor-intensive process as "penetration testing" allows reducing time, financial costs and other resources. The main methods for identifying vulnerabilities, including the use of artificial intelligence, have been studied. The presented approach to optimizing the penetration testing process in automated process control systems uses machine learning algorithms. Preference is given to machine learning with reinforcement, which is based on the Deep Q-learning algorithm. The integration of network scanning methods, building an attack graph and training neural networks to effectively identify vulnerabilities and risks in network infrastructures is proposed in the paper. To build an attack graph, the MITER ATT&CK knowledge base using the GBVA Framework is utilized, and the Deep Q-learning algorithm is used to select optimal actions during testing.

Keywords: vulnerabilities, information security, penetration testing, machine learning, Deep Q-learning.

For citation:
Nyrkov, Anatoliy P., Elena S. Yumasheva, and Anton V. Kirikov. "Optimization of the penetration testing process in automated process control systems using machine learning algorithms." *Vestnik Gosudarstvennogo universiteta morskogo i rechnogo flota imeni admirala S. O. Makarova* 16.3 (2024): 456–466. DOI: 10.21821/2309-5180-2024-16-3-456-466.

УДК 004.56

ОПТИМИЗАЦИЯ ПРОЦЕССА ТЕСТИРОВАНИЯ НА ПРОНИКНОВЕНИЕ В АСУ ТЕХНОЛОГИЧЕСКИМИ ПРОЦЕССАМИ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ

А. П. Ныркov, Е. С. Юмашева, А. В. Кириков

ФГБОУ ВО «ГУМРФ имени адмирала С. О. Макарова»,
Санкт-Петербург, Российская Федерация

Темой работы является исследование процесса широкого внедрения автоматизированных информационно-управляющих систем в промышленности, энергетике и на транспорте. Отмечается, что повышение их сложности неизбежно приводит к возникновению различного рода уязвимостей в этих системах, наличие которых позволяет злоумышленникам проникать в автоматизированные управляющие системы, брать их под свой контроль, а также нарушать нормальный режим работы управляемых ими

технологических процессов. Подчеркивается, что в течение последнего десятилетия успешные кибератаки были зафиксированы в энергетике, в том числе атомной, в морском судоходстве, в портовых перегрузочных комплексах, а также в других системах. Превентивный подход к обеспечению безопасности автоматизированных управляющих систем заключается в выявлении и использовании существующих уязвимостей путем имитации возможных кибератак. Отмечается, что автоматизация такого достаточно трудоемкого процесса, как «тестирование на проникновение», позволяет сократить время, финансовые затраты и другие ресурсы. Исследованы основные методы выявления уязвимостей, в том числе с применением искусственного интеллекта. В представленном подходе к оптимизации процесса тестирования на проникновение в автоматизированные системы управления технологическими процессами использованы алгоритмы машинного обучения. Предпочтение отдано машинному обучению с подкреплением, основу которого составляет алгоритм Deep Q-learning. Предлагается интеграция методов сканирования сети, построения графа атак и обучения нейронных сетей для эффективного выявления уязвимостей и рисков в сетевых инфраструктурах. Для построения графа атак используются базы знаний MITRE ATT&CK с применением GBVA Framework, для выбора оптимальных действий в процессе тестирования — алгоритм Deep Q-learning.

Ключевые слова: уязвимости, информационная безопасность, тестирование на проникновение, машинное обучение, Deep Q-learning.

Для цитирования:
Ныркov А. П. Оптимизация процесса тестирования на проникновение в АСУ технологическими процессами с использованием алгоритмов машинного обучения / А. П. Ныркov, Е. С. Юмашева, А. В. Кириков // Вестник Государственного университета морского и речного флота имени адмирала С. О. Макарова. — 2024. — Т. 16. — № 3. — С. 456–466. DOI: 10.21821/2309-5180-2024-16-3-456-466.

Введение (Introduction)

Одной из актуальных тенденций современности является возрастающее количество атак на промышленный сегмент сети, в частности, на автоматизированные системы управления технологическими процессами (далее — АСУ ТП). Большинство предприятий промышленности давно перешли на автоматизированное производство, при этом регуляторы и крупные вендоры только относительно недавно (в морской отрасли с 2012 г.) начали изучать вопрос защиты таких систем. Нормативно-правовая база в данной области не является пока окончательно сформировавшейся. Несмотря на то, что с момента выхода одного из основополагающих федеральных законов в этой области — ФЗ № 187 от 26.07.2017 «О безопасности критической информационной инфраструктуры» [1], прошло 7 лет, лишь 2–3 года назад организации начали уделять более пристальное внимание реализации требований закона: началась разработка концепций, появились требования к развертыванию АСУ ТП и т. д.

Причиной особого внимания к информационной безопасности на объектах критической информационной инфраструктуры явился вирус Stuxnet [2], который впервые в истории преодолел барьер между виртуальным миром и реальным. Этот вирус является точечной разработкой, цель которой состоит в отключении ключевой части иранской ядерной программы путем уничтожения центрифуг, используемых для обогащения урана. После заражения компьютера вирус Stuxnet проверяет возможность подключения к программируемым логическим контроллерам (далее — ПЛК). Если доступ к ПЛК обнаружен, то Stuxnet начинает постепенно подменять часть STEP7, отвечающую за прошивку кода ПЛК. Подмена кода позволяет прослушивать сеть Profibus-DP, генерировать / подменять необходимые для дальнейшего закрепления сетевые пакеты, читать входы и управлять выходами ПЛК, подключаться к базе данных WinCC. Для получения успешных результатов злоумышленникам пришлось использовать комплексный подход к атаке, в основе которого применение социальной инженерии и программно-аппаратной части с использованием уязвимостей операционной системы.

Цифровые технологии становятся все более важными в морской сфере. Степень сложности и разнообразие технологий, используемых на современных судах, включая навигационные, информационно-коммуникационные и эксплуатационные системы, допускают различные возможные уязвимости. Любой сбой, вызванный кибератакой, может иметь не только серьезные

экономические и экологические последствия, но и приводить к человеческим жертвам [3], [4]. Порты в этом отношении также не следует игнорировать, поскольку переход на цифровые технологии в перегрузочных и складских операциях создает дополнительные угрозы, что делает эти территории привлекательными для злонамеренных действий со стороны субъектов угроз [5].

За последние 7 лет в этой области было зарегистрировано несколько атак. Примечательным примером является кибератака в июне 2017 г. на глобальную судоходную компанию Maersk, которая привела к массовым сбоям в работе и значительным финансовым потерям [6]. В ходе другого инцидента, произошедшего в феврале 2017 г., киберпреступники взяли под контроль ИТ-системы контейнеровоза, принадлежащего Германии [7]. В декабре 2019 г. итальянская базовая станция АИС также подверглась значительным помехам, когда в районе о. Эльба внезапно появились сотни фальшивых кораблей, что привело к нарушению мониторинга судоходства в этом районе [7].

В данной статье исследован комплексный подход к вопросам формирования системы информационной безопасности на подобных объектах, а также повышения общего уровня ее защищенности за счет внедрения автоматизированного процесса тестирования на проникновение.

Актуальность этой работы обусловлена возросшим количеством кибератак на АСУ ТП, что наглядно отображено в ежегодных отчетах крупных вендоров (Kaspersky, Positive Technologies) [8], а также нехваткой квалифицированных специалистов в области тестирования, так как эта задача является довольно трудоемкой.

Методы и материалы (Methods and Materials)

Методы выявления уязвимостей. Одним из ключевых методов в современной оценке степени защищенности объекта, при котором выполняется всесторонний анализ и исследование системы, является тестирование на проникновение (Penetration Testing (PT)). Целью моделирования реальных сценариев атак является оценка потенциальных рисков и повышение общего уровня защищенности системы путем проведения упреждающих мероприятий по устранению уязвимостей, а именно: анализ архитектуры системы, выявление слабых мест и определение последующих векторов развития потенциальных атак. Однако это довольно ресурсозатратная задача как с финансовой точки зрения, так и с точки зрения человеческих ресурсов, поэтому автоматизация поиска уязвимостей не является новой технологией в этой области.

Разработано большое количество инструментов, облегчающих работу тестировщиков и способствующих повышению их эффективности. Эти инструменты включают сканеры сетей и уязвимостей, а также базы известных уязвимостей под каждую систему. Наиболее популярным в настоящее время является Metasploit, содержащий огромную базу известных эксплойтов системных уязвимостей. Подобные инструменты помогают специалистам работать на более высоком уровне абстракции, позволяющем сосредоточиться на поиске уязвимостей и выборе эксплойтов вместо рутинной работы на низком уровне ручного тестирования, однако использование данных инструментов без высококвалифицированного специалиста не имеет смысла.

Один из подходов к решению задачи проведения эффективного и надежного тестирования заключается в применении методов машинного обучения из области искусственного интеллекта. Первичные концепции строились лишь на создании «графов атак», которые должны были моделировать существующую сеть как граф подключенных компьютеров, где атаки могут быть построены в сети на основе известных уязвимостей и эксплойтов. При этом графы атак в реализации классического машинного обучения могут быть полезны лишь при полном знании системы, что накладывает определенные ограничения.

Другой концепцией подходов является моделирование атак как частично наблюдаемого марковского процесса принятия решений (MDP). Такое моделирование позволяет исключить предположение о том, что конфигурация каждого хоста заранее известна, и дает возможность моделировать наблюдения за конфигурациями по мере продвижения атаки. Однако данный способ плохо масштабируется и применим лишь на одной хост-машине.

Penetration Testing в рамках одного аудита может иметь довольно большой спектр целей и задач, поэтому данная область довольно хорошо регламентирована международными стандартами и методиками, начиная с первичного взаимодействия аудитора и заказчика и заканчивая порядком испытаний тестируемых объектов. Суммируя концепции наиболее популярных методологий и стандартов, таких как Open Source Security Testing (OSSTMM), Open Web Application Security Project (OWASP), National Institute of Standards and Technology (NIST), Penetration Testing Methodologies and Standards (PTES), Information System Assessment Framework (ISSAF), обобщенный процесс пентеста можно представить в виде следующих этапов:

– планирование и определение области тестирования;
– сбор информации о сетевой инфраструктуре;
– сканирование, обнаружение и оценка сетевых уязвимостей;
– предоставление отчета о проведенной работе и определение контрмер.

В данном исследовании будем опираться на стандарт PTES, разработанный в 2009 г., точнее на его обновленную версию 1.1, выпущенную в 2017 г. Этот стандарт включает пять основных технических аспектов проведения тестирования на проникновение, приведенных в следующей таблице.

Этапы тестирования на проникновение

Аспекты	Описание	Пример реализации
Сбор информации об объекте тестирования (Intelligence Gathering)	Сбор как можно большего количества информации из общедоступных источников.	Open source intelligence (OSINT): анализ физической инфраструктуры, сканирование сети, анализ используемого ПО и т. д.
Анализ уязвимостей / сканирование (Enumeration / Scanning)	Обнаружение приложений и сервисов, работающих в системе.	Анализ уязвимостей ОС, БД, VPN, транспортных протоколов, беспроводных сетей и т. д.
Эксплуатация уязвимостей (Exploitation)	Формирование векторов атаки, ориентированных на проникновение в защищенный периметр, за счет выявленных уязвимостей	DOS-атаки, атаки на протоколы WEP и WAP, атаки на сетевые протоколы и т. д.
Повышение привилегий (Privilege Escalation)	Расширение прав доступа в системе, горизонтальное или вертикальное повышение привилегий.	Эксплуатация уязвимостей с целью получения root-прав
Постэксплуатация (Post-exploitation)	Сбор дополнительной информации для расширения области тестирования, зачистка следов присутствия в системе.	Уязвимости ОС, ПО, формирование закладок для последующей эксплуатации, обход внутренних средств защиты; и т. д.

Стандарт PTES рекомендует проводить регулярные сессии планирования для переоценки собранной информации и продолжать пункт Post-exploitation до тех пор, пока не будут достигнуты поставленные цели.

Анализ алгоритмов машинного обучения. Машинное обучение (Machine Learning (ML)) — один из разделов / ветвей области ИИ, основная цель которого состоит не в создании чего-либо нового, а в прогнозировании, запоминании, принятии наилучшего выбора на основе обучающей выборки и алгоритмов. ML имеет три основные ветви: классическое обучение, обучение с подкреплением нейросети и глубокое обучение. Каждая из ветвей имеет свои преимущества и недостатки в различных ситуациях.

Классическое машинное обучение используется во многих поисковых системах интернета при рекомендации ссылок на страницы сайтов, кинофильмов, музыки, статей и т. п. Классическое ML бывает двух типов: *контролируемое обучение* (Supervised Learning) или *обучение без контролера* (Unsupervised Learning). Для первого типа заранее должен быть подготовлен большой мас-

сив размеченных данных для обучения и дальнейшего сопоставления результатов. Примерами алгоритмов этой категории являются: наивный Байес, деревья решений, логическая регрессия, метод опорных векторов и др. Такие алгоритмы показывают хорошие результаты в задачах категорирования объектов по заранее известным признакам (например, определение аномальных транзакций, прогнозирование поведения рынка ценных бумаг, диагностирование медицинских заболеваний и т. д.).

Для задачи автоматизации Penetration Testing данная ветвь ML не подходит, так как содержит некоторые ограничения, а именно:

– сбор и разметка большого объема данных для обучения алгоритма, что, в свою очередь, влечет огромные финансовые и трудовые затраты;

– классические алгоритмы, подразумевающие работу с теми же данными, на которых базировались обучающие выборки, и изучающие только базовые сигнатуры и закономерности в предоставленной обучающей выборке.

Алгоритмы обучения без контролера используются в основном в качестве метода анализа данных. К ним относятся метод *k*-средних, метод главных компонент, сингулярное разложение и др. Эти методы используют для сегментации рынка, сжатия изображений, построения детекторов аномального поведения, но при этом повышается риск ложных срабатываний. Однако применение таких методов также неоправданно при построении алгоритма, автоматизирующего процесс пентеста, ввиду следующих причин:

– наличие высокого риска ложно положительных срабатываний, неправильное выявление уязвимостей, отчеты о нахождении связей, которых на самом деле нет;

– обучение требует хоть и не размеченного, но также большого объема данных для обучающей выборки;

– необходимо большое количество итераций для получения итоговых результатов.

Машинное обучение с подкреплением (Reinforcement Learning (RL)) — это метод ML, в котором система вырабатывает способ принятия решений для достижения наиболее оптимальных результатов.

Обучение с подкреплением состоит из двух основных объектов: *Агента* и *Окружающей среды* (рис. 1). *Агент* — это в данном случае алгоритм, на который постоянно воздействует окружающая среда S_t. Благодаря ей он получает информацию через признаки, состояния или наблюдения, совершает определенные действия A_t, получает за их выполнение вознаграждения R_t и переходит в следующее состояние. Цикл будет повторяться до тех пор, пока окружение не отправит признак окончания работы.

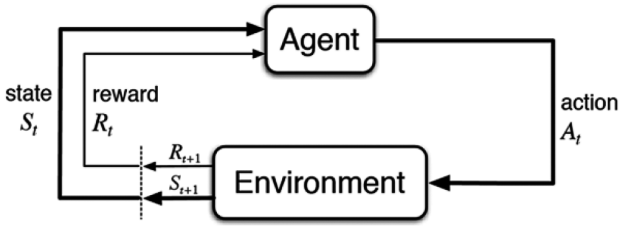

Рис. 1. Взаимодействие агента с окружающей средой [9]
Fig. 1. Agent interaction with the environment [9]

RL похоже на имитацию игры на выживание, т. е. *Агент* пытается обобщить ситуации, для того чтобы выйти из них с максимальной выгодой. В 2016 г. машина обыграла человека в одну из древнейших игр ГО [10], несмотря на то, что ранее была доказана невозможность пересчета всех комбинаций [11], так как их общее число превышает количество атомов во Вселенной. Победа оказалась возможной не из-за пересчета всех возможных комбинаций, а благодаря выбору наилучшего выхода из каждой конкретной ситуации. Именно на этой идее базируется алгоритм *Q-learning*, т. е. *Агент* учится выбирать наиболее качественные решения, воспринимая и запоминая все ситуации подобно марковскому процессу принятия решений [9]. Также наиболее существенным «плюсом»

таких алгоритмов является то, что потребность в обучающей выборке заменяется поощрениями (reward) за выполненные действия от окружающей среды. Цель *Агента* состоит не в нахождении единственно верного ответа, а в максимизации своей «награды» за выполненную работу.

Основой этого метода является алгоритм *Deep Q-learning* (DQL), который использует нейронные сети для аппроксимации функции действий, что позволяет более гибко моделировать сложные задачи и зависимости между состояниями и действиями. Например, пусть Q (s, a) — Q-значная функция, где s — состояние; a — действие. Цель DQL состоит в том, чтобы обучить параметризованную сеть $Q(s, a, \theta)$, где θ — весовой вектор.

Архитектура Q-сети состоит из нескольких слоев нейронов (рис. 2). Входом может быть состояние среды, на выходе каждый нейрон представляет оценку Q-значений для соответствующего действия.

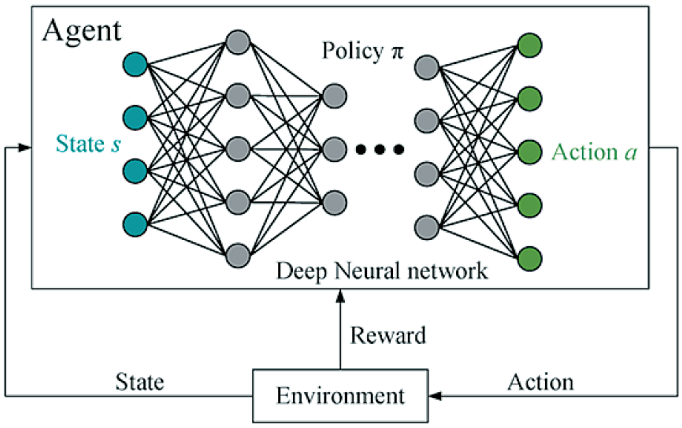

Рис. 2. Схема глубокого обучения с подкреплением
Fig. 2. Deep Reinforcement Learning diagram

Обновление весов θ происходит при помощи минимизации следующей функции потерь:

$$L(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - Q^{\theta}(s_i, a_i)\right)^2, \quad (1)$$

где y_1 — целевое значение, к которому необходимо приблизиться.

Используем для этого алгоритм обновления Q-значения на основе уравнения Беллмана:

$$y_j = \begin{cases} r_j; \\ r_j + \gamma \max_{a'} Q^{\theta}(s',\ a'), \end{cases} \quad (2)$$

где r_j — награда за выполнение действия;

γ — коэффициент дисконтирования.

Далее с помощью метода стохастического градиентного спуска (SGD) будем обновлять веса нейронной сети таким образом, чтобы значения Q-функции приблизить к целевым значениям:

$$\theta = \theta - \alpha\nabla_{\theta} L(\theta), \quad (3)$$

где α — скорость обучения, принимающая значения между 0 и 1.

Таким образом, суть алгоритма DQL состоит в том, чтобы последовательно обновлять веса сети, используя полученные данные об обучении и вычисленные значения Q-функции, минимизируя ошибку между текущими предсказаниями и целевыми Q-значениями.

Результаты (Results)

Предлагаемый подход к тестированию на проникновение. Для использования DQL в задаче тестирования на проникновение система тестирования должна быть построена как математическая модель MDP, описывающая принятие решений в среде, где результаты действий частично

случайны и зависят только от текущего состояния и выполненного действия. Данную модель можно описать как упорядоченную пятерку:

$$\langle S, A, P, R, \gamma \rangle,$$

где S — множество возможных состояний среды, в которых может находиться система во время тестирования (например, информация об уязвимостях, запущенных службах, идентификация хоста, его привилегии и др.);

A — множество действий, которые может выполнить агент в каждом состоянии (могут включать в себя сканирование уязвимостей и их эксплуатацию и т. п.);

P — матрица перехода между состояниями, определяющая вероятность перехода из одного состояния в другое при выполнении конкретного действия;

R — функция вознаграждения, определяющая количество вознаграждения или штрафа для каждой пары «состояние – действие» (вознаграждение может быть присвоено за обнаружение уязвимости и / или ее успешную эксплуатацию, штраф — за обнаружение ложной уязвимости или неудачные попытки ее эксплуатации);

γ — коэффициент дисконтирования, который определяет вес будущих вознаграждений по сравнению с текущими (используется для балансировки краткосрочных или долгосрочных целей агента).

Агент выбирает свои действия на основе заданной стратегии, а стратегия определяет, какие действия предпринимать в каждом состоянии среды для максимизации ожидаемых вознаграждений.

Ожидаемые вознаграждения представим в виде суммы всех будущих наград R_t, начиная с момента t до бесконечности:

$$R_t = R_{t+1} + \gamma R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}. \quad (4)$$

Таким образом, использование принципов MDP позволяет агенту принимать обоснованные решения на основе получаемых данных, эффективно управлять рисками и ожидаемыми вознаграждениями, и создавать адаптивную стратегию, которая обеспечит наилучшие результаты при обнаружении и предотвращении уязвимостей в системе.

Алгоритм сканирования сети. На первом этапе выполняется сканирование сети с помощью инструмента Nmap, используемого также для обнаружения устройств, открытых портов и сервисов, работающих на этих портах. Результаты сканирования сохраняются в формате, который можно анализировать и использовать на следующих этапах.

Вторым этапом является построение *дерева атак* с помощью GBVA Framework на основе MITRE ATT&CK [12] — базы знаний, разработанной MITRE Corporation, которая описывает типичные тактики, методы и техники злоумышленников при выполнении кибератак. Сначала алгоритм извлекает информацию о методах атаки и уязвимостях из базы, далее на основе полученной информации происходит построение графа атак, представляющего потенциальные пути атаки на целевую сеть. Затем система применяет алгоритм Deep First Search (DFS) для эффективного исследования потенциальных векторов [13], который обеспечивает исследование каждого вектора до конца, прежде чем переходит к альтернативным направлениям, что позволяет обнаруживать наиболее глубокие и потенциально опасные уязвимости сети.

На третьем этапе происходит инициализация нейронной сети и начало обучения. Алгоритм Deep Q-learning используется для выбора оптимальных действий *Агента* в каждом состоянии графа атак. Система стремится минимизировать сумму потерь и максимизировать сумму вознаграждений, получаемых за успешно выполненные атаки. Q-функция оценивает ожидаемую сумму вознаграждений, которые получит агент, выбрав действие a в состоянии s, т. е. оптимально выполняя процедуру до конца итераций. Ее обновление происходит на основе ошибки между

текущей и целевой оценкой, рассчитанной с учетом полученного вознаграждения и оценки состояния на следующем шаге:

$$Q(s,a) = Q(s,a) + \alpha\left(r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right), \quad (5)$$

где $Q(s,a)$ — текущая оценка функции для состояния s и действия a;

r — полученное вознаграждение;

α — скорость обучения;

γ — дисконт фактор, учитывающий влияние будущих наград;

s' — состояние, в которое агент переходит после выполнения действия a' в состоянии s;

$\max_{a'} Q(s',a')$ — максимальное значение Q-функции для всех возможных действий a' в состоянии s'.

Параметры нейронной сети обновляются с использованием градиентного спуска и обратного распространения ошибки [14, 15]. Функция потерь для обновления параметров Q-сети вычисляется как среднеквадратическая ошибка (MSE) между текущей оценкой Q-функции и ее целевым значением:

$$L(\theta_i) = MSE\left(y - Q(s_t, a_t, \theta_t)\right)^2. \quad (6)$$

Матрица переноса P_{nn} представляет собой матрицу, описывающую возможные переходы между состояниями в графе атак, обновление которой происходит в соответствии с текущими результатами атак и обучением нейронной сети:

$$P_{nn} = \begin{bmatrix} p_{11}\ p_{12}\ \cdots\ p_{1n} \\ p_{21}\ p_{22}\ \cdots\ p_{2n} \\ \cdots\ \cdots\ p_{ij}\ \cdots \\ p_{m1}\ p_{m2}\ \cdots\ p_{mn} \end{bmatrix} \quad (7)$$

где p_{ij} — вероятность перехода из состояния s_i в состояние s_j при выполнении конкретного действия a_k.

Псевдокод алгоритма PT:

1. Получение информации о топологии сети.
2. Выявление чувствительных значений хоста.
3. Использование GBVA Framework для создания дерева атак.
4. Поиск векторов графа атак с использованием алгоритма Deep First Search.
5. Создание матрицы переноса (7).
6. Инициализация нейронных сетей.
7. **for** шаг_итерации **from** 1 **to** Max **do.**
8. **for** t **from** 1 **to** T **do**.
9. Выбор действия Агента a в соответствии со значениями Q-функции (5).
10. Выполнение действия a и получение последующего состояния среды s', вознаграждения r.
11. Добавление кортежа перехода (s, a, r, s') в D.
12. $s = s'$.
13. **if** $|D|$ > размера пакета **then**.
14. Выборка и построение целевого значения Q в соответствии с уравнением (2).
15. Выполнение шага градиентного спуска с функцией потерь (6).
16. Замена весовых параметров θ.
17. **end if.**
18. **end for.**
19. **end for.**

Заключение (Conclusion)

В работе рассмотрен подход к оптимизации процесса тестирования на проникновение в АСУ ТП с использованием алгоритмов машинного обучения. Предложено использование итерации методов сканирования сети (Nmap), построения графа атак на основе базы знаний MITRE ATT&CK с применением GBVA Framework, обучения нейронных сетей для эффективного выявления уязвимостей и рисков в сетевых инфраструктурах, а также алгоритм Deep Q-learning для выбора оптимальных действий в процессе тестирования на проникновение.

Несмотря на то, что экспериментальная часть работы еще не выполнена, теоретический анализ предложенного подхода позволяет предположить его достаточную эффективность и перспективность в контексте оптимизации процесса тестирования на проникновение в АСУ ТП. Ожидается, что предложенная методика позволит автоматизировать процесс выявления уязвимостей и повысить эффективность действий испытателя на проникновение, что, в свою очередь, позволит сократить время и затраты на обнаружение и устранение уязвимостей в сетевых системах.

Дальнейшие исследования будут направлены на завершение создания необходимых для работы системы тестирования модулей и проведение экспериментальной проверки предложенного подхода с целью подтверждения его эффективности.

СПИСОК ЛИТЕРАТУРЫ

1. Федеральный закон «О безопасности критической информационной инфраструктуры» от 26.07.2017 № 187 [Электронный ресурс]. — Режим доступа: http://www.consultant.ru/document/cons_doc_LAW_220885/ (дата обращения 21.10.2023).

2. An Unprecedented Look at Stuxnet, the World's First Digital Weapon [Электронный ресурс]. — Режим доступа: https://www.wired.com/2014/11/countdown-to-zero-day-stuxnet/ (дата обращения: 03.11.2023).

3. *Когтев А. В.* Обоснование необходимости создания автоматизированной информационной системы оценки и прогнозирования киберугроз на морских судах под флагом РФ / А. В. Когтев // Современные тенденции и перспективы развития водного транспорта России: матер. межвуз. науч.-практ. конф. аспирантов, студентов и курсантов. 20 мая 2021 г. — СПб.: Изд-во ГУМРФ им. адм. С. О. Макарова, 2021. — 2 часть. — С. 37–41.

4. *Kardakova M.* Cyber security on sea transport / M. Kardakova, I. Shipunov, A. Nyrkov, T. Knysh // Energy Management of Municipal Transportation Facilities and Transport. — Cham: Springer International Publishing, 2018. — Pp. 481–490. DOI: 10.1007/978-3-030-19756-8_46.

5. *Наташова К. В.* К вопросу о категорировании объектов критической информационной инфраструктуры морских портов / К. В. Наташова, С. С. Соколов, О. Н. Губернаторов, А. П. Нырков, А. В. Кириков // Безопасность информационных технологий. — 2020. — Т. 27. — № 2. — С. 35–46. DOI: 10.26583/bit.2020.1.03.

6. *Alcaide J. I.* Critical infrastructures cybersecurity and the maritime sector / J. I. Alcaide, R. G. Llave // Transportation Research Procedia. — 2020. — Vol. 45. — Pp. 547–554. DOI: 10.1016/j.trpro.2020.03.058

7. *Androjna A.* Assessing cyber challenges of maritime navigation / A. Androjna, T. Brcko, I. Pavic, H. Greidanus // Journal of Marine Science and Engineering. — 2020. — Vol. 8. — Is. 10. — Pp. 776. DOI: 10.3390/jmse8100776.

8. Attacks on industrial sector hit record in second quarter of 2023 [Электронный ресурс]. — Режим доступа: https://www.kaspersky.com/about/press-releases/2023_attacks-on-industrial-sector-hit-record-in-second-quarter-of-2023 (дата обращения: 20.11.2023).

9. *Sutton R. S.* Reinforcement Learning: An Introduction / R. S. Sutton, A. G. Barto. — Second edition. — Cambridge: MIT Press, 2020. — 526 p.

10. *Воронцов Н.* AlphaGo победила человечество в го [Электронный ресурс] / Н. Воронцов. — Режим доступа: https://nplus1.ru/news/2017/05/25/now-it-is-official (дата обращения: 30.11.2023).

11. *Воронцов Н.* Точное количество разрешенных комбинаций в го оказалось больше числа атомов во Вселенной [Электронный ресурс] / Н. Воронцов. — Режим доступа: https://nplus1.ru/news/2016/01/25/mathematical (дата обращения: 22.11.2023).

12. ATT&CK Matrix for Enterprise [Электронный ресурс]. — Режим доступа: https://attack.mitre.org (дата обращения: 22.11.2023).

13. *Нырков А. П.* Дискретная математика: кодирование и обработка дискретных структур данных / А. П. Нырков, А. Ю. Кузнецов, Е. В. Зуров, А. В. Башмаков. — СПб.: ГУМРФ им. адм. С. О. Макарова, 2022. — 92 с.

14. *Zhilenkov A. A*. Intelligent autonomous navigation system for UAV in randomly changing environmental conditions / A. A. Zhilenkov, S. G. Chernyi, S. S. Sokolov, A. P. Nyrkov // Journal of Intelligent & Fuzzy Systems. — 2020. — Vol. 38. — Is. 5. — Pp. 6619–6625. DOI: 10.3233/JIFS-179741.

15. *Sokolov S.* Hybrid neural networks in cyber physical system interface control systems / S. Sokolov, A. Zhilenkov, S. Chernyi, A. Nyrkov, N. Glebov // Bulletin of Electrical Engineering and Informatics. — 2020. — Vol. 9. — No. 3. — Pp. 1268–1275. DOI: 10.11591/eei.v9i3.1293.

REFERENCES

1. Federal'nyi zakon "O bezopasnosti kriticheskoi informatsionnoi infrastruktury" ot 26.07.2017 № 187. Web. 21 Oct. 2023 <http://www.consultant.ru/document/cons_doc_LAW_220885/>.

2. An Unprecedented Look at Stuxnet, the World's First Digital Weapon. Web. 3 Nov. 2023 <https://www.wired.com/2014/11/countdown-to-zero-day-stuxnet/>.

3. Kogtev A. V. "Justification of the need to create an automated information system for assessing and predicting cyber threats on sea vessels under the flag of the Russian Federation." *Sovremennye tendentsii i perspektivy razvitiya vodnogo transporta Rossii: mater. mezhvuz. nauch.-prakt. konf. aspirantov, studentov i kursantov*. Vol. 2. SPb.: Izd-vo GUMRF im. adm. S. O. Makarova, 2021. 37–41.

4. Kardakova, Maria, Ilya Shipunov, Anatoly Nyrkov, and Tatyana Knysh. "Cyber security on sea transport." *Energy Management of Municipal Transportation Facilities and Transport*. Cham: Springer International Publishing, 2018. 481–490. DOI: 10.1007/978-3-030-19756-8_46.

5. Natashova, K. V., S. S. Sokolov, O. N. Gubernatorov, A. P. Nyrkov, and A. V. Kirikov. "On the issue of categorization of objects of critical information infrastructure of seaports." *IT Security (Russia)* 27.2 (2020): 35–46. DOI: 10.26583/bit.2020.1.03.

6. Alcaide, Juan Ignacio, and Ruth Garcia Llave. "Critical infrastructures cybersecurity and the maritime sector." *Transportation Research Procedia* 45 (2020): 547–554. DOI: 10.1016/j.trpro.2020.03.058.

7. Androjna, Andrej, Tanja Brcko, Ivica Pavic, and Harm Greidanus. "Assessing Cyber Challenges of Maritime Navigation." *Journal of Marine Science and Engineering* 8.10 (2020): 776. DOI: 10.3390/jmse8100776.

8. Attacks on industrial sector hit record in second quarter of 2023. Web. 20 Nov. 2023 <https://www.kaspersky.com/about/press-releases/2023_attacks-on-industrial-sector-hit-record-in-second-quarter-of-2023>.

9. Sutton, R.S., and A. G. Barto. *Reinforcement Learning: An Introduction*. Second edition. Cambridge: MIT Press, 2020.

10. Vorontsov, N. AlphaGo defeated humanity in Go. Web. 30 Nov. 2023 <https://nplus1.ru/news/2017/05/25/now-it-is-official>.

11. Vorontsov, N. The exact number of allowed combinations in Go turned out to be greater than the number of atoms in the Universe. Web. 22 Nov.2023 <https://nplus1.ru/news/2016/01/25/mathematical>.

12. ATT&CK Matrix for Enterprise. Web. 22 Nov. 2023 <https://attack.mitre.org>.

13. Nyrkov, A.P., A. Yu. Kuznetsov, E. V. Zurov, and A. V. Bashmakov. *Diskretnaya matematika: kodirovanie i obrabotka diskretnykh struktur dannykh*. SPb.: GUMRF im. adm. S. O. Makarova, 2022.

14. Zhilenkov, Anton A., S. G. Chernyi, S. S. Sokolov, and A. P. Nyrkov. "Intelligent autonomous navigation system for UAV in randomly changing environmental conditions." *Journal of Intelligent & Fuzzy Systems* 38.5 (2020): 6619–6625. DOI: 10.3233/JIFS-179741.

15. Sokolov, Sergei, Anton Zhilenkov, Sergei Chernyi, Anatoliy Nyrkov, and Nikolay Glebov. "Hybrid neural networks in cyber physical system interface control systems." *Bulletin of Electrical Engineering and Informatics* 9.3 (2020): 1268–1275. DOI: 10.11591/eei.v9i3.1293.

ИНФОРМАЦИЯ ОБ АВТОРАХ

Ныркov Анатолий Павлович —
доктор технических наук, профессор
ФГБОУ ВО «ГУМРФ имени адмирала
С. О. Макарова»
198035, Российская Федерация, Санкт-Петербург,
ул. Двинская, 5/7
e-mail: *kaf.koib@gmail.com*,
nyrkowap@gumrf.ru

INFORMATION ABOUT THE AUTHORS

Nyrkov, Anatoliy P. —
Dr. of Technical Sciences, professor
Admiral Makarov State University of Maritime
and Inland Shipping
5/7 Dvinskaya Str., St. Petersburg, 198035,
Russian Federation
e-mail: *kaf.koib@gmail.com*,
nyrkowap@gumrf.ru

Юмашева Елена Сергеевна — аспирант
Научный руководитель:
Нырков Анатолий Павлович
ФГБОУ ВО «ГУМРФ имени адмирала
С. О. Макарова»
198035, Российская Федерация, Санкт-Петербург,
ул. Двинская, 5/7
e-mail: *yumasheva.lena@list.ru*, *kaf_koib@gumrf.ru*
Кириков Антон Викторович — аспирант
Научный руководитель:
Нырков Анатолий Павлович
ФГБОУ ВО «ГУМРФ имени адмирала
С. О. Макарова»
198035, Российская Федерация, Санкт-Петербург,
ул. Двинская, 5/7
e-mail: *tony-68@yandex.ru*, *kaf_koib@gumrf.ru*

Yumasheva, Elena S. — Postgraduate
Supervisor:
Nyrkov, Anatoliy P.
Admiral Makarov State University of Maritime
and Inland Shipping
5/7 Dvinskaya Str., St. Petersburg, 198035,
Russian Federation
e-mail: *yumasheva.lena@list.ru*, *kaf_koib@gumrf.ru*
Kirikov, Anton V. — Postgraduate
Supervisor:
Nyrkov, Anatoliy P.
Admiral Makarov State University of Maritime
and Inland Shipping
5/7 Dvinskaya Str., St. Petersburg, 198035,
Russian Federation
e-mail: *tony-68@yandex.ru*, *kaf_koib@gumrf.ru*

Статья поступила в редакцию 24 мая 2024 г.
Received: May 24, 2024.