DOI 10.36622/1682-7813.2025.28.1.002
УДК 621.372.004

АВТОМАТИЗИРОВАННЫЙ ПОИСК СООТВЕТСТВИЯ УЯЗВИМОСТЕЙ, ШАБЛОНОВ И ТЕХНИК КИБЕРАТАК

А.Г. Остапенко, Ю.В. Макаров, Д.А. Нархов, А.Е. Нетребин, Н.А. Исаев, Д.Р. Зинченко, Д.И. Чаусов

В статье предлагается программно формировать процедуру установления соответствия шаблонов и техник уязвимостей, которая зачастую разработчика систем защиты информации приводит к неадекватным решениям при поиске вышеуказанного соответствия, являющегося важнейшим во всех последующих расчетах и вытекающих из них проектных реакциях. Разрешение перечисленных противоречий предлагается осуществить через разработку соответствующего методического и алгоритмического обеспечения, ориентированного на машинное обучение имеющимся в данном пространстве датасетом с последующим использованием интеллектуальных возможностей нейросетей. В этой связи приобретение и обработка данных о существующих уязвимостях, а также их отношениях с известными техниками атак становятся ключевыми аспектами для формирования надежной автоматизированной системы анализа. Экспериментальные результаты, основанные на применении предложенных алгоритмов машинного обучения, продемонстрировали значительное повышение точности идентификации соответствий между уязвимостями и техниками, минимизируя риск неэффективных решений при проектировании систем защиты информации.

Ключевые слова: уязвимость, техники, шаблон атаки, программно-аппаратные ошибки, CAPEC, CWE, CVE, нейросеть.

Введение

На основе сопоставления уязвимостей с техниками формируется представление о процессе реализации сценария. В качестве исходных параметров для соответствия выступает способ эксплуатации программно-аппаратной ошибки, последствия реализации и объект (компонент) уязвимости [1].

Цель работы заключается в разработке и автоматизации алгоритма сопоставления шаблонов с техниками уязвимостей, а также в реализации данного алгоритма с использованием технологий машинного обучения для повышения точности и эффективности анализа уязвимостей в информационных системах [2].

Информационное обеспечение

В условиях быстроразвивающейся информационной среды эффективное управление уязвимостями становится ключевой задачей для организаций, стремящихся защитить свои информационные системы. В данной работе предлагается использовать несколько авторитетных баз данных для создания надежной системы автоматизированного поиска соответствий между шаблонами атак, программно-аппаратными ошибками и уязвимостями.

MITRE ATT&CK – это общедоступная база знаний о тактике и методах злоумышленников, основанная на наблюдениях в реальных условиях. База знаний ATT&CK используется в качестве основы для разработки конкретных моделей угроз и методологий в частном секторе, в правительстве, а также в сообществе разработчиков продуктов и услуг в сфере кибербезопасности. Она помогает определить соответствия шаблонов атак с программно-аппаратными ошибками, что позволяет специалистам по кибербезопасности не только идентифицировать угрозы, но и разрабатывать методы защиты на основе известных сценариев атак [3].

MITRE CWE (Common Weakness Enumeration) классифицирует программно-

аппаратные ошибки, работающие как потенциальные уязвимости. Использование данной базы позволяет составить список ключевых уязвимостей, актуальных для конкретной системы или приложения. Это помогает разработчикам и аналитикам быстрее выявлять и устранять ошибки в программном обеспечении, что снижает риски, связанные с уязвимостями [4].

База данных уязвимостей (БДУ) ФСТЭК содержит сведения об уязвимостях программного и программно-аппаратного обеспечения, а также перечень и описание угроз безопасности информации для информационных систем различного назначения. Этот ресурс поддерживает процессы обнаружения уязвимостей и оценки уровня риска, связанного с каждой из них [5].

MITRE EMB3D - это модель угроз для встроенных устройств, используемых в таких отраслях, как критически важная инфраструктура, Интернет вещей, автомобилестроение, здравоохранение, производство и многих других. Модель угроз предназначена для того, чтобы помочь поставщикам, владельцам/операторам активов, тестирующим организациям и специалистам по безопасности повысить общую безопасность аппаратного и программного обеспечения встроенных устройств. Эта модель угроз призвана служить центральным хранилищем информации, определяющим известные угрозы для встроенных устройств и их уникальные функции/свойства, которые позволяют совершать определённые действия. Сопоставляя угрозы с соответствующими функциями/свойствами устройства, пользователь может легко определить степень подверженности угрозам на основе известных функций устройства [6].

Для сопоставления программно-аппаратных ошибок с техниками сценариев используется MITRE EMB3D, что позволит установить взаимосвязь между типами атак и конкретными уязвимостями в программных системах. Установление таких связей обеспечивает комплексный подход к безопасности, позволяя прогнозировать и предотвращать возможные угрозы.

Предполагается использование существующих инцидентов из базы данных MITRE EMB3D, которые демонстрируют конкретные атаки и соответствующие им программно-аппаратные ошибки. Эти инциденты будут отправной точкой для анализа и сопоставления с уязвимостями в программных системах. Такой подход помогает не только выявлять уязвимости, но и разрабатывать меры по их устранению, основываясь на знаниях о реальных сценариях использования уязвимостей, а также на основе примеров реализации реальных атак обучить нейросеть.

Интеграция представленных ресурсов создаёт эффективную основу для автоматизированного анализа и повышения уровня безопасности информационных систем.

Методическое обеспечение

В рамках данного исследования предполагается разработка методического обеспечения для автоматизированного сопоставления уязвимостей с техниками атак. Основой для этого послужат методы, представленные на сайтах CVE MAPPING METHODOLOGY и MITRE [7-8]. Данный подход предполагает несколько ключевых этапов, направленных на глубокий анализ сценариев атак и их последствий для информационных систем.

На первоначальном этапе будут исследованы существующие методики сопоставления уязвимостей с техниками на платформах CVE MAPPING METHODOLOGY и MITRE. Эти исследования позволят выделить основные принципы и методы, применяемые для анализа сценариев реализации уязвимостей.

В данных методах уделяется внимание таким факторам, как тип эксплуатации уязвимости и возможные последствия, которые могут возникнуть после реализации сценария атаки. На основании результатов может быть разработана система для оценки ущерба, который может быть нанесен информационной системе. Ущерб оценивается с учетом первичных (непосредственно от использования уязвимости) и вторичных (долгосрочные последствия, такие как утечка данных или нарушение работы систем) факторов.

На основе исследования представленных методов сопоставления предлагается использовать уже существующие техники реализации сценариев атак, которые связаны с программно-аппаратными ошибками, на основании данной информации появляется возможность обучить нейросеть на реальных инцидентах, без ручного создания dataset. Для этого будет использована база данных MITRE EMB3D, содержащая сведения о инцидентах, где каждый инцидент ассоциируется с конкретной программно-аппаратной ошибкой (CWE) и набором техник, применяемых для успешной реализации сценария атаки. Этот этап включает разработку алгоритма для автоматического сопоставления между выявленными уязвимостями и соответствующими техниками на основе информации из MITRE EMB3D.

На основе данного метода предусмотрено тестирование её эффективности на реальных сценариях кибератак, чтобы продемонстрировать способность методики не только идентифицировать уязвимости, но и предлагать адекватные меры по их устранению. Будет проведен ретроспективный анализ инцидентов, описанных на MITRE EMB3D.

С учетом динамичности угроз предложенная автоматизация с помощью машинного обучения в дальнейшем предоставит возможность наиболее в гибкой форме оценивать существующие инциденты и предлагать наиболее адекватные меры по смягчению и устранению последствий, за счет его самообучения.

В результате, разработанный метод позволит создать адаптивную систему для автоматизированного сопоставления уязвимостей с техниками атак, что существенно повысит уровень безопасности информационных систем и оперативность реагирования на возникающие угрозы.

Лингвистическое образование и технологии

Для успешной реализации поставленных задач в сфере лингвистического образования необходима надежная программная база, способная эффективно выполнять все требуемые функции. Начальный этап включает «выгрузку» информации из существующих актуальных баз данных, а последующие этапы сосредоточены на анализе и компоновке этих данных. Для этого потребуются ряд простых скриптов, с учетом особенности ресурсов с необходимой информации, которые будут работать с такими типами данных, как XML, CSV, JSON, TXT и другими форматами, в которых обычно доступна информация из существующих хранилищ (например, БДУ ФСТЭК, NVD NIST, MITRE ATT&CK и др.).

Наиболее удобным подходом для написания такого скрипта будет использование библиотеки Pandas языка программирования Python. Эта библиотека предлагает широкий функционал для корректного считывания и обработки данных, что делает её основным инструментом для реализации задуманного.

Собранные данные будут заноситься в единую базу данных, основанную на PostgreSQL - доступной системе управления базами данных с открытым исходным кодом. Поскольку информация из источников, с которых производится выгрузка, обычно относится только к одному элементу цепочки CAPEC-CWE-CVE-Техники, целесообразно проектировать в базе данных отдельные таблицы для хранения полной информации об этих элементах, таких как CAPEC, CWE и CVE. Важно также создавать совместные таблицы, отражающие факты соответствия между CAPEC и CWE, а также между CAPEC и CVE, что будет полезно для анализа.

Такой подход к проектированию базы данных поможет избежать перегрузки таблиц избыточной информацией. При использовании SQL-запросов можно будет эффективно собирать необходимые данные из нескольких таблиц и выводить их в одном окне. На рис. 1 изображена структура базы данных.

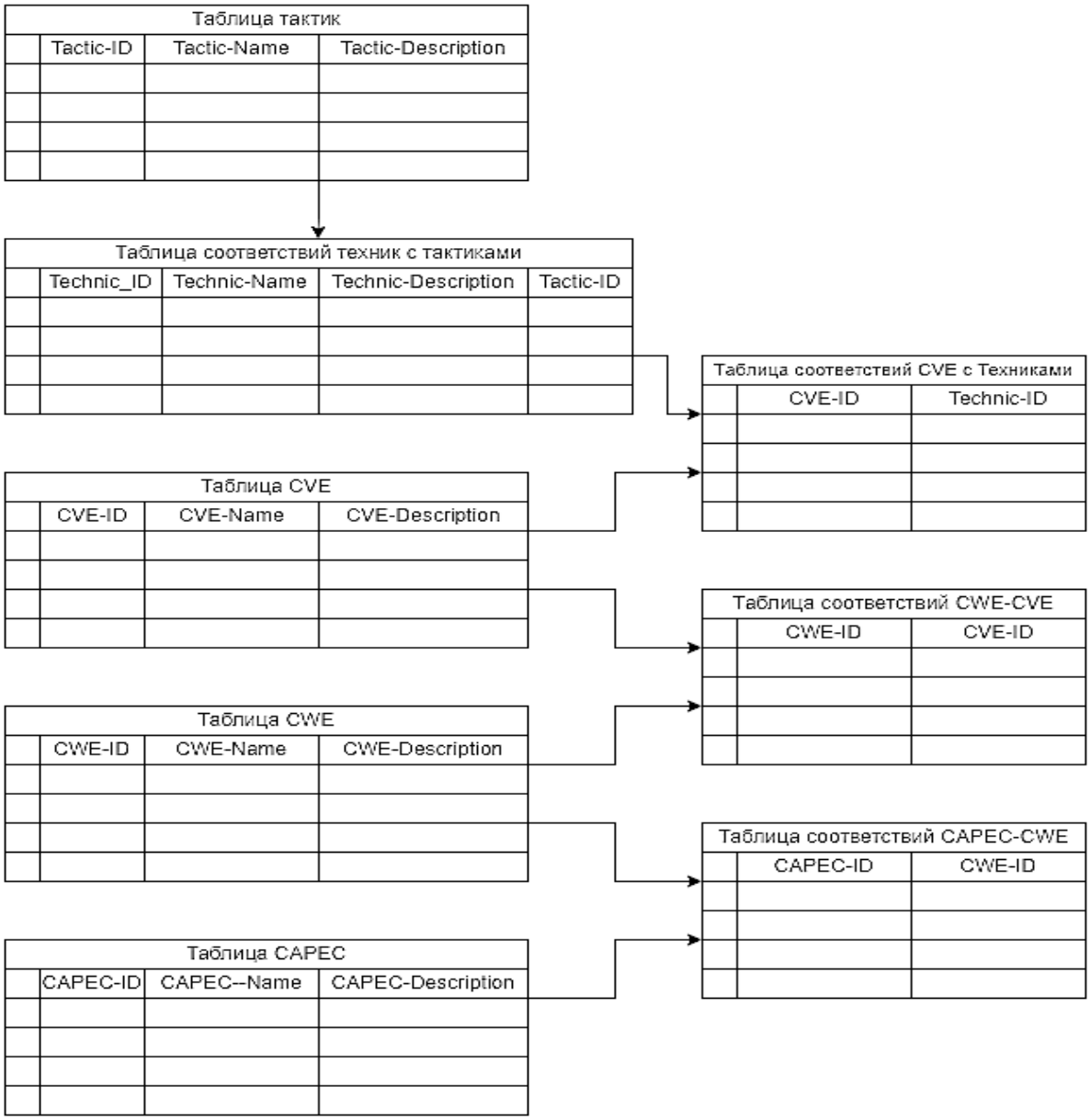

Рис. 1. Структура базы данных и связей между таблицами

Разработка программной части

Для организации взаимодействия с базами данных используется программа, написанная на языке программирования Python с применением фреймворка FastAPI. Python был выбран как наиболее удобный и эффективный язык для работы с большими данными благодаря своему простому синтаксису и автоматическому управлению памятью. Кроме того, Python обладает мощными библиотеками для обработки данных и машинного обучения, что делает его идеальным для задач анализа данных.

FastAPI - это современный фреймворк, который упрощает разработку веб-приложений на Python, позволяя создавать приложения, доступные через браузер. Такой подход предлагает несколько преимуществ. Во-первых, все ответы сервера будут представлены в виде веб-страниц, что позволяет визуальной части программы отображать данные в удобном для чтения формате, отличном от стандартного JSON. Во-вторых, использование FastAPI значительно упрощает развертывание сервера, превращая разработанное приложение в полноценный веб-сайт, что открывает возможности для взаимодействия с данными в реальном времени.

На рис. 2-5 показаны результаты, когда пользователь вводит нужный идентификатор CAPEC и получает связанные с ним CWE,

соответствующие CVE и техники. Пользователь может также кликнуть на любой из атрибутов, чтобы их идентификатор, название и описание. С описанием и названием CWE также выводятся сопоставленные нейросетью техники и обоснование, почему именно данные техники вошли в перечень.

Рис. 2. Поле ввода для взаимодействия с программой

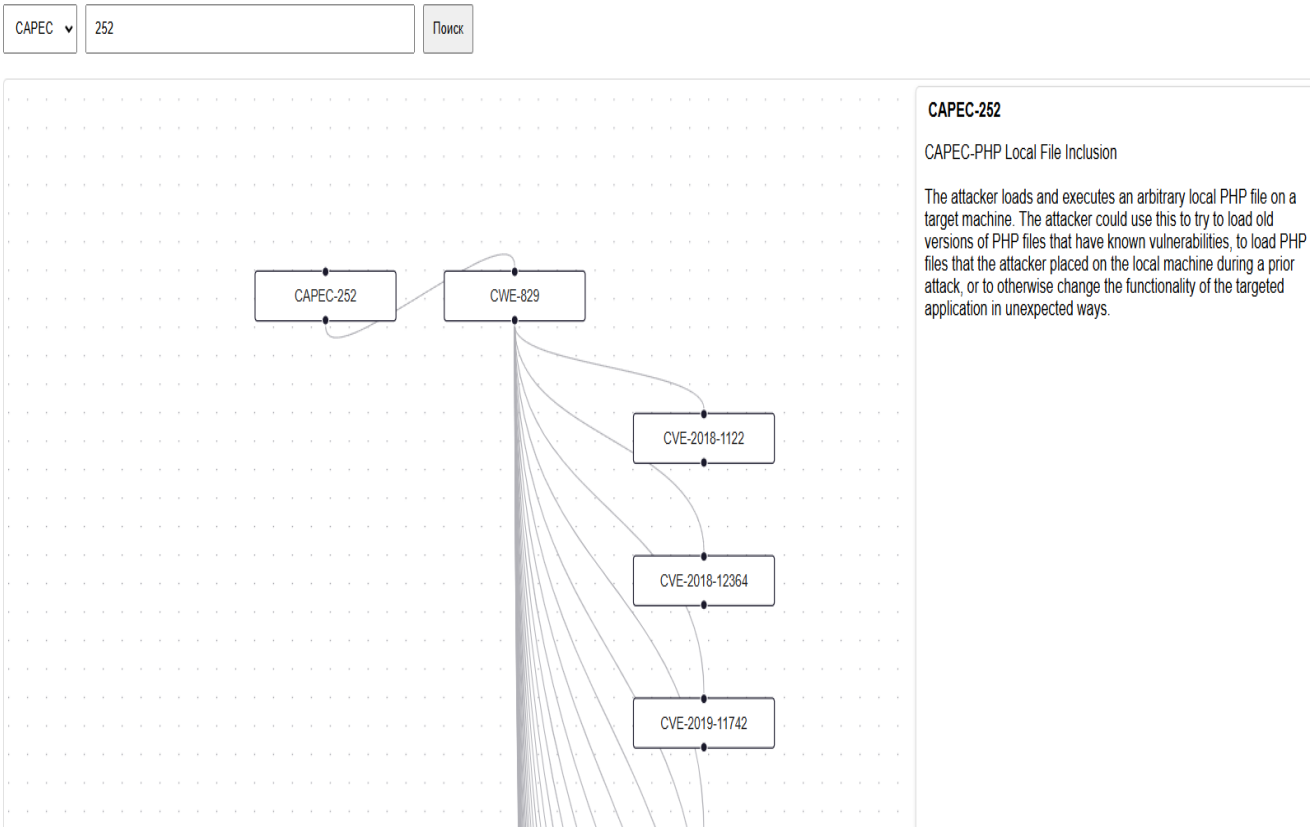

Рис. 3. Демонстрация выходных данных, когда пользователь вводит нужный идентификатор CAPEC

На рис. 6-8 показаны результаты, когда пользователь вводит нужный идентификатор CWE и получает связанные с ним CAPEC, CVE и техники с возможностью получить информацию о всех атрибутах.

На рис. 9-11 показаны результаты, когда пользователь вводит нужный идентификатор CWE и получает связанные с ним CAPEC, CVE и техники с возможностью получить информацию о всех атрибутах.

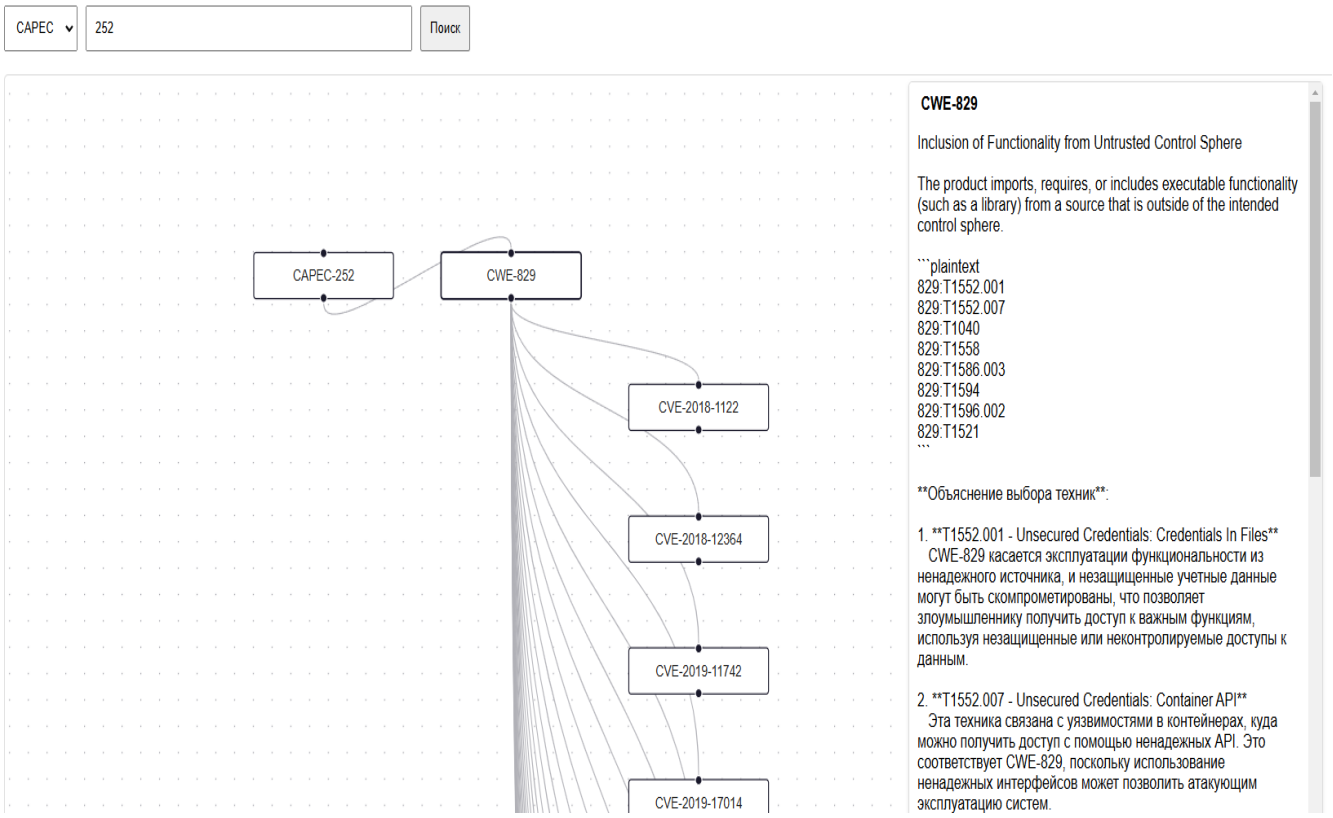

Рис. 4. Демонстрация выходных данных, когда пользователь вводит нужный идентификатор CAPEC

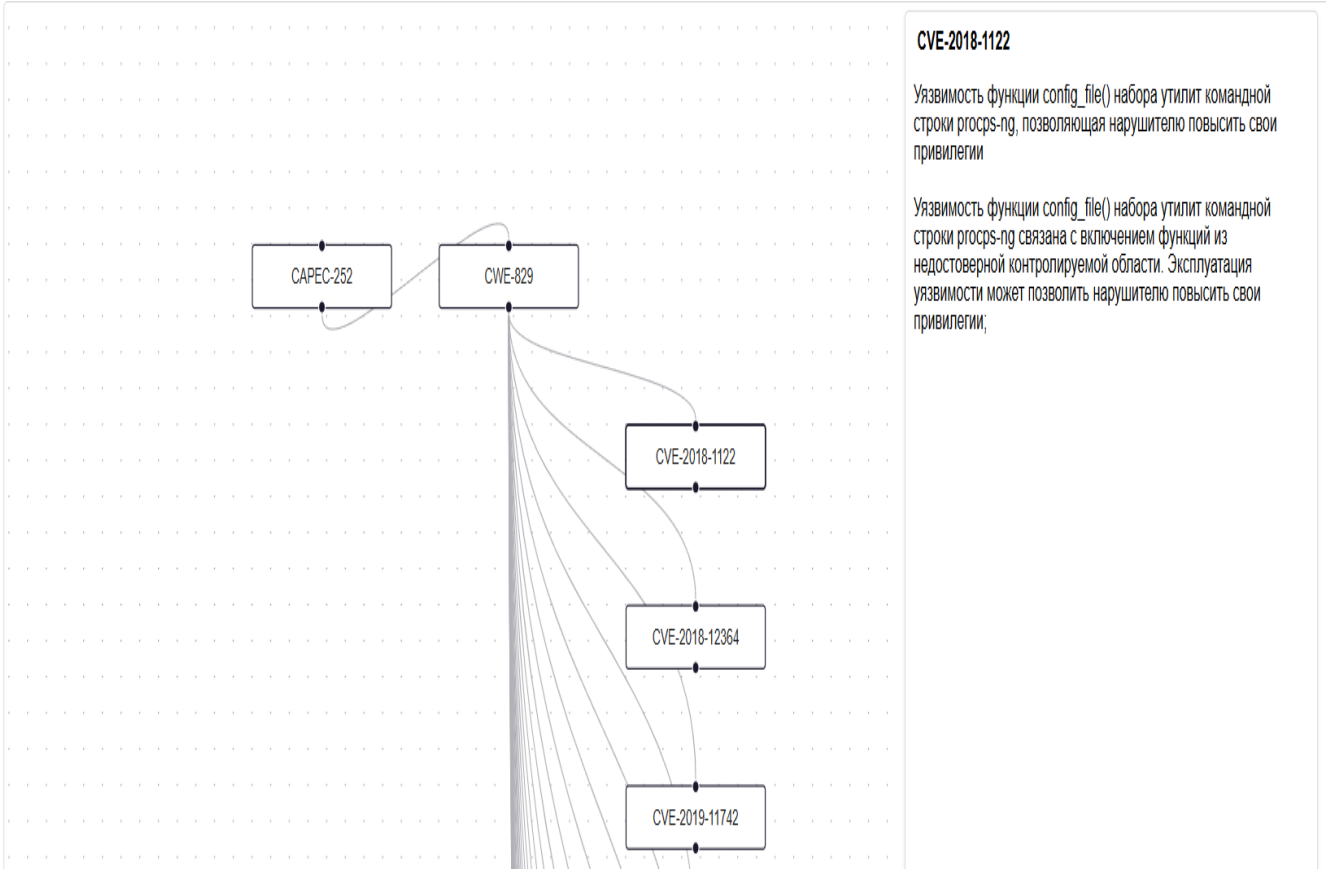

Рис. 5. Демонстрация выходных данных, когда пользователь вводит нужный идентификатор CAPEC

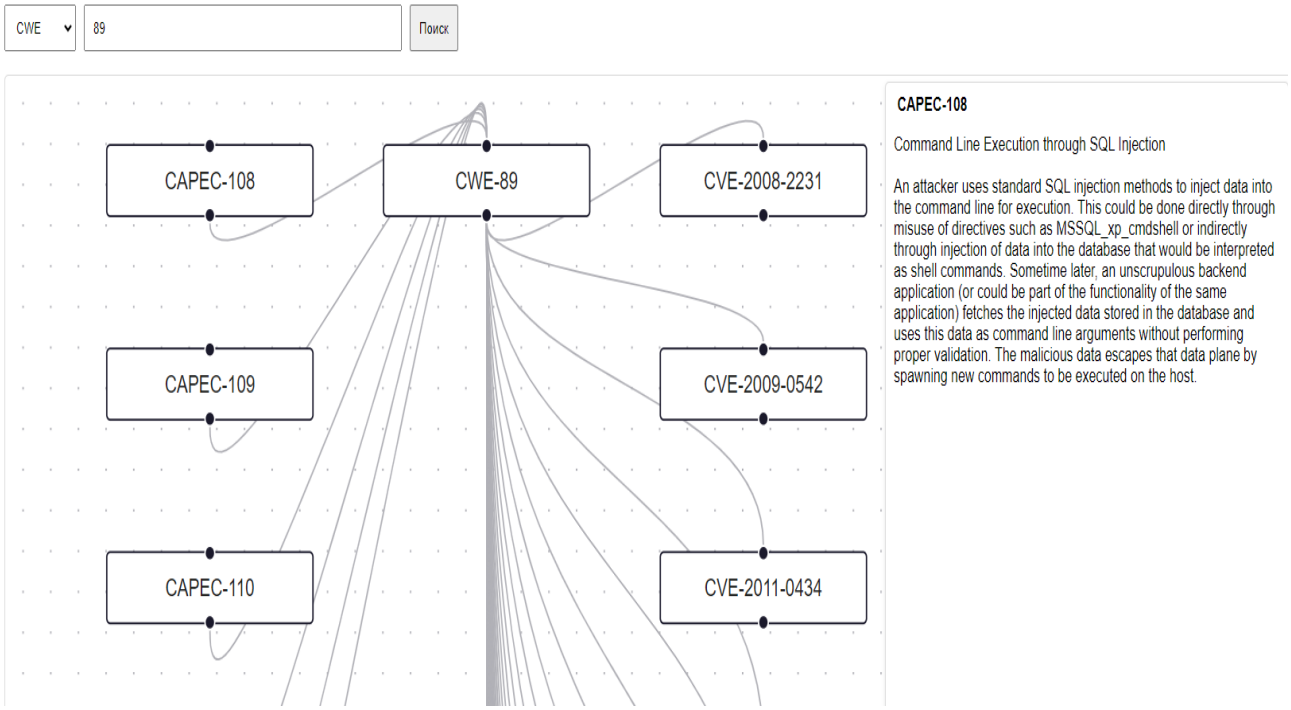

Рис. 6. Демонстрация выходных данных, когда пользователь вводит нужный идентификатор CWE

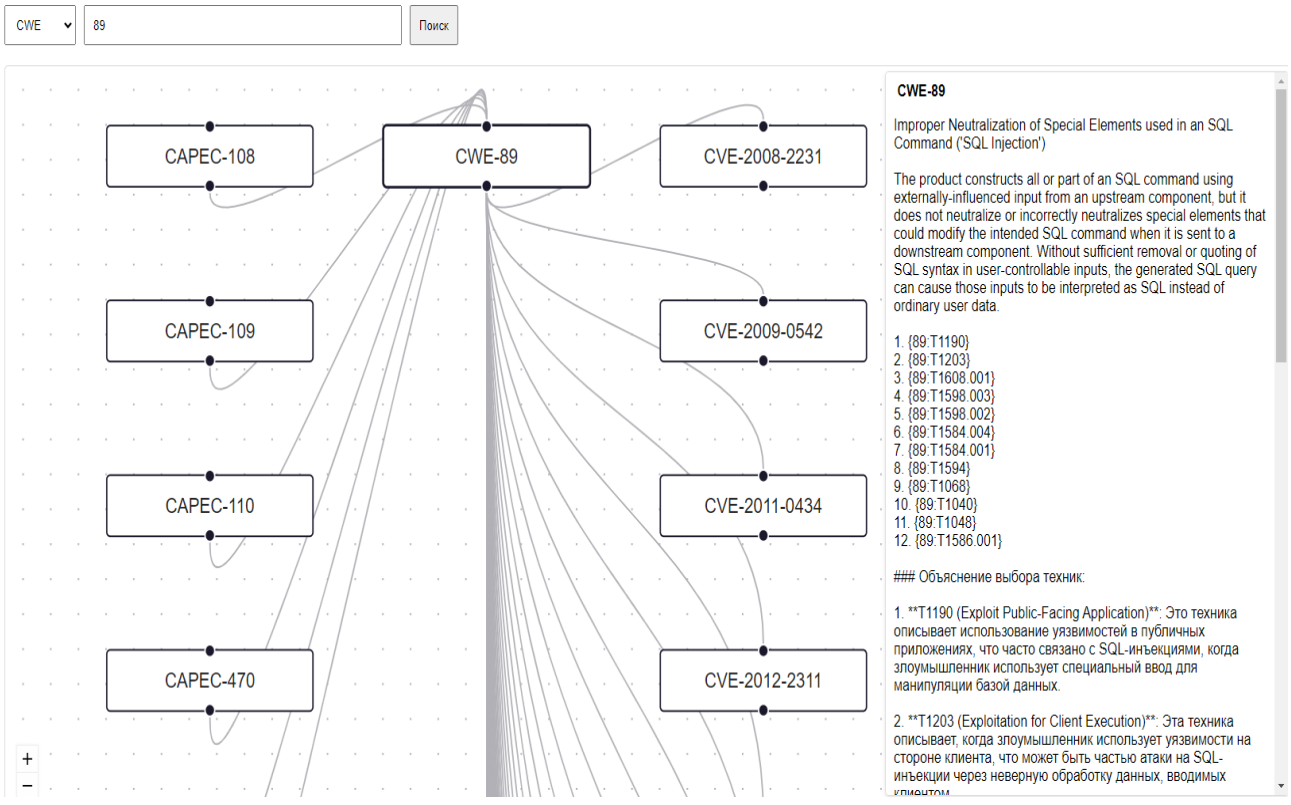

Рис. 7. Демонстрация выходных данных, когда пользователь вводит нужный идентификатор CWE

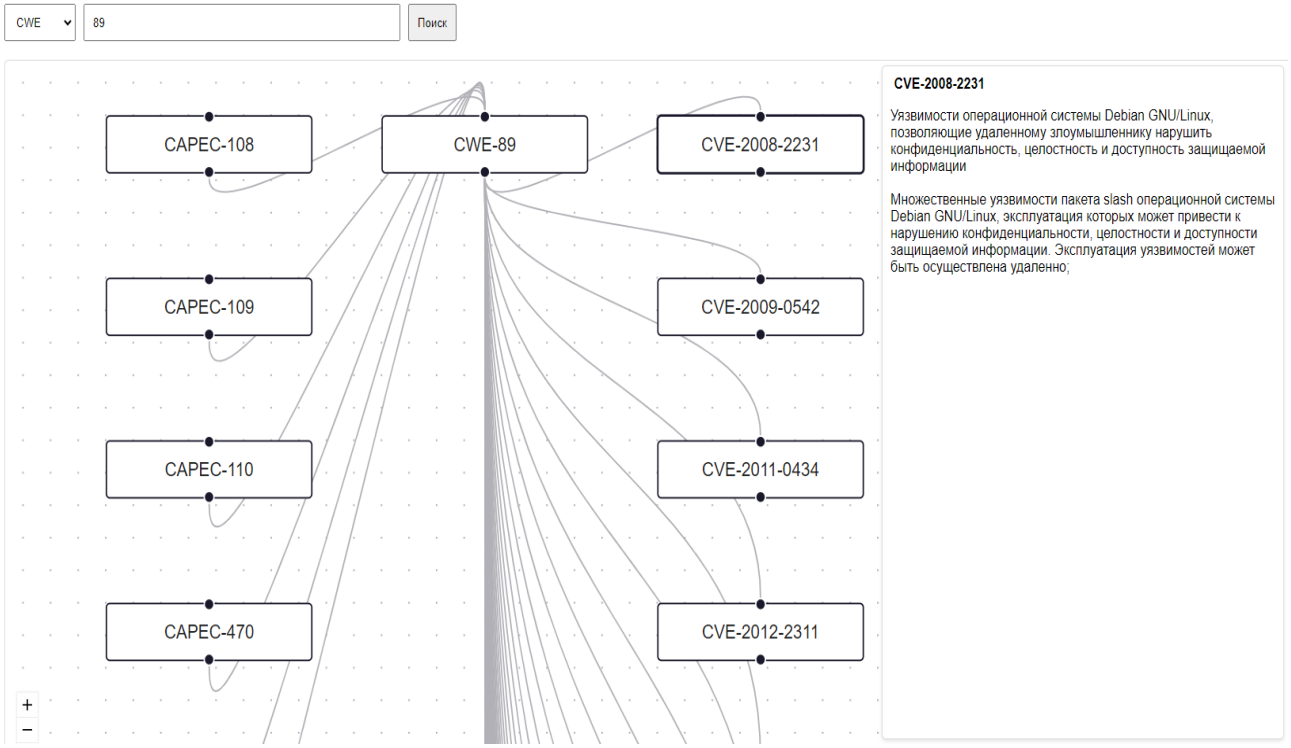

Рис. 8. Демонстрация выходных данных, когда пользователь вводит нужный идентификатор CWE

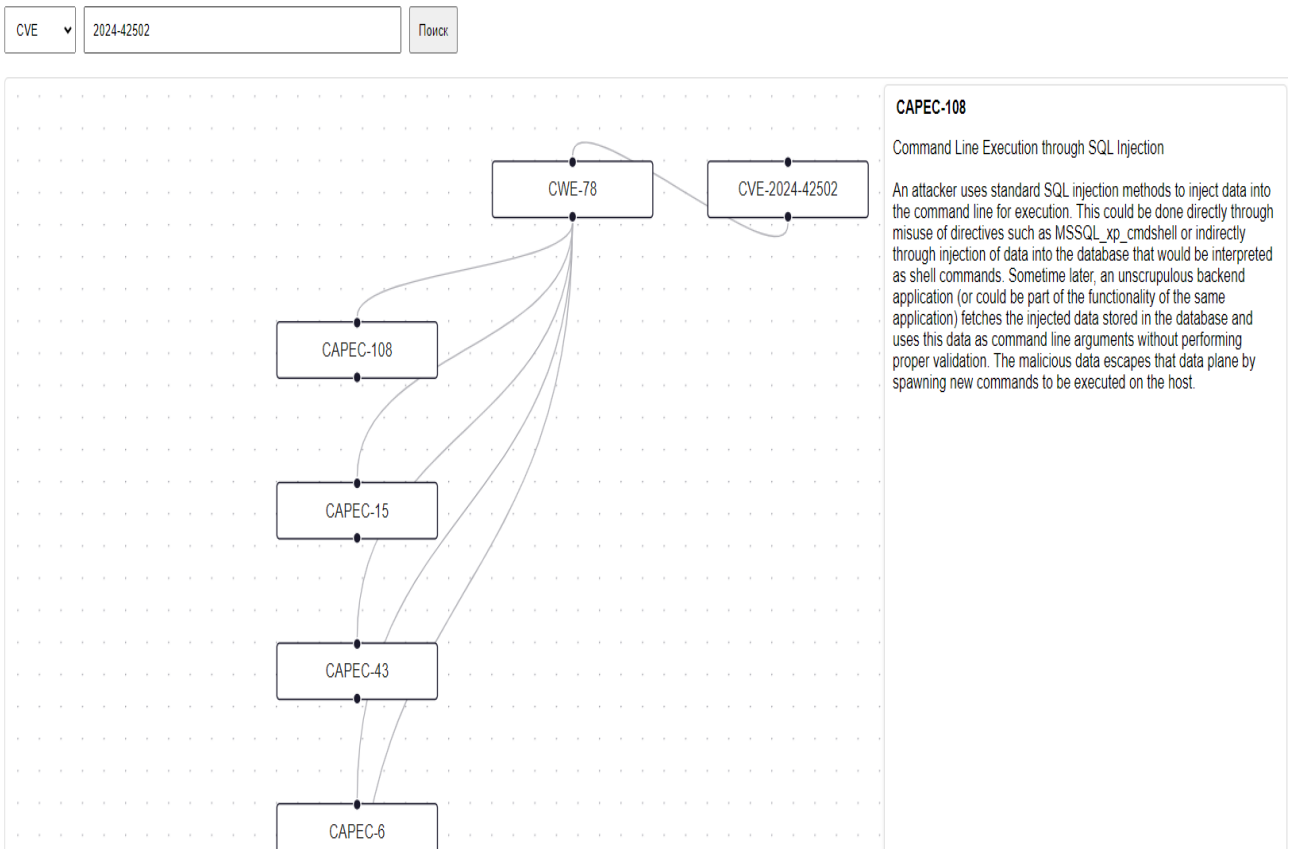

Рис. 9. Демонстрация выходных данных, когда пользователь вводит нужный идентификатор CVE

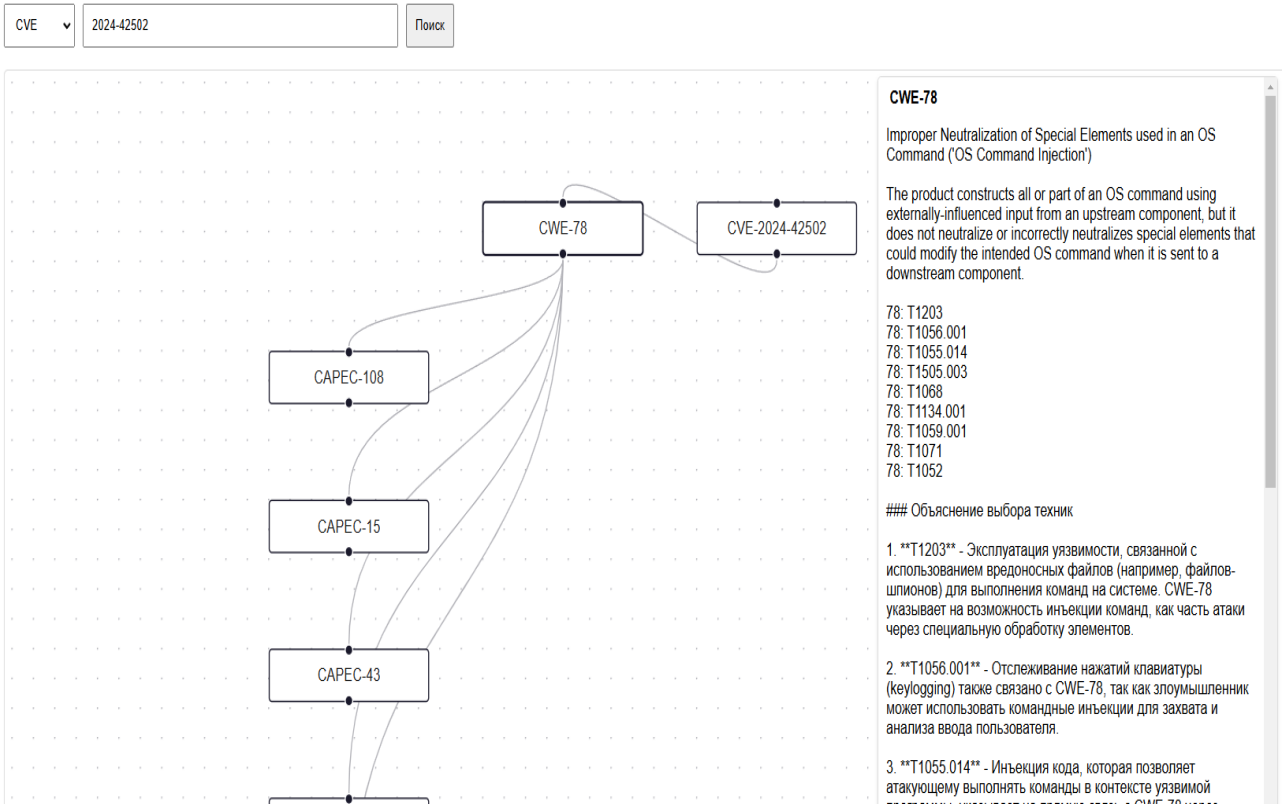

Рис. 10. Демонстрация выходных данных, когда пользователь вводит нужный идентификатор CVE

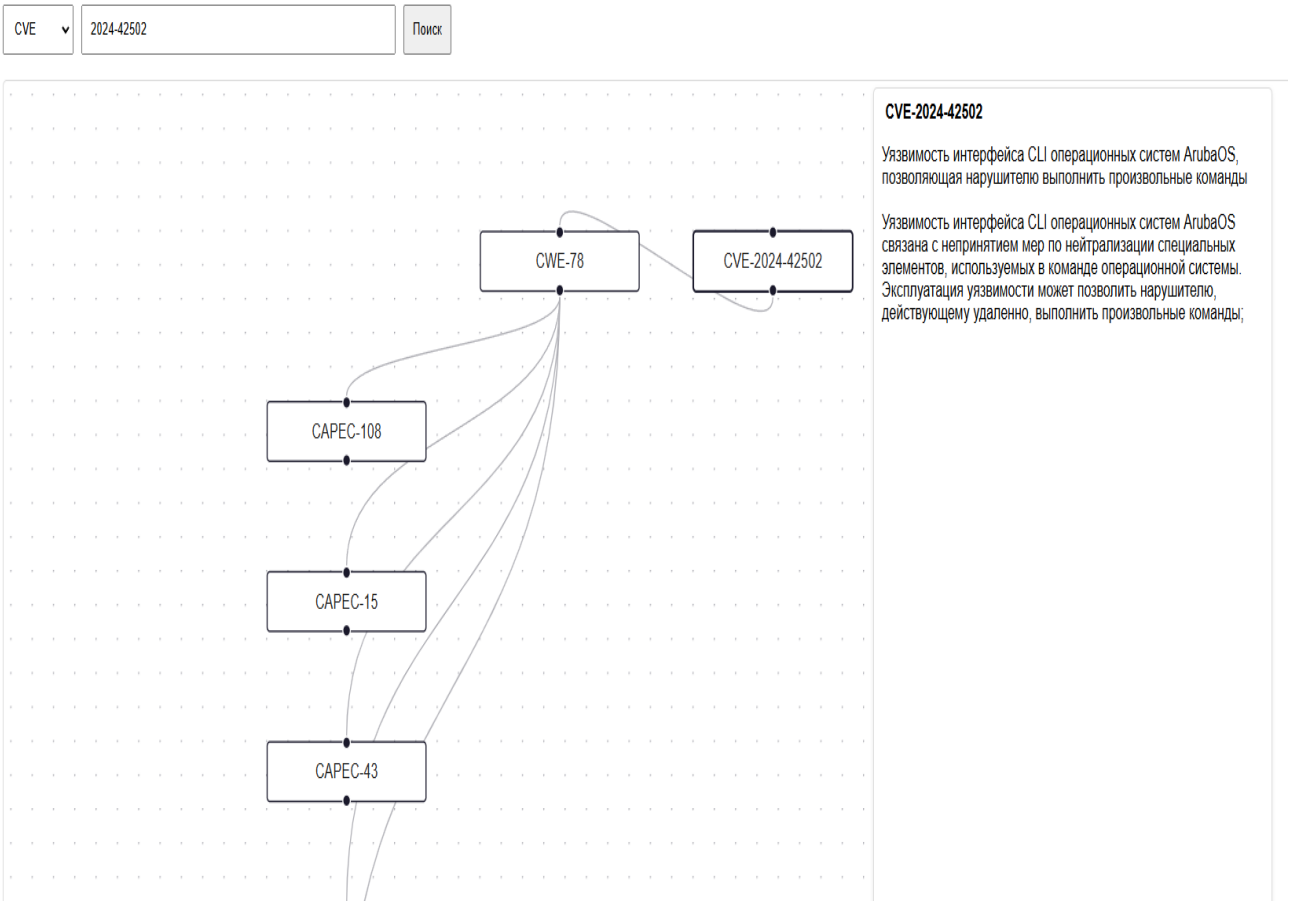

Рис. 11. Демонстрация выходных данных, когда пользователь вводит нужный идентификатор CVE

Все данные извлекаются непосредственно из базы данных, что обеспечивает их актуальность и достоверность, без генерации информации извне. Эта архитектура также позволяет легко расширять базу данных, добавляя описания, дополнения и другую информацию по мере появления новых данных.

Описание сценариев атак может быть дополнено подразделом, позволяющим рассчитывать вероятность реализации конкретных сценариев на основе аналитических данных о системе. Это обеспечивает возможность математической оценки состояния безопасности, что открывает путь для разработки адаптивных политик безопасности и регламентов по обнаружению, регистрации, реагированию и ликвидации последствий инцидентов нарушения сетевой безопасности. Кроме того, взаимодействие со сценариями атаки может быть обогащено графом реализации уязвимостей, который иллюстрирует последовательность действий злоумышленника, а также уязвимости и техники, используемые для их эксплуатации, что позволяет глубже понять механизмы атак и повысить уровень защиты.

Информация о CVE может быть расширена добавлением раздела для расчета вероятностей и ущербов эксплуатации уязвимости. Выполнение расчетов возможно на основании калькуляторов уязвимостей, в частности версии CVSS 3.1, предоставляющей данные о критичности уязвимости, но не учитывающей индивидуальные особенности системы. И версии калькулятора CVSS 4.0, который имеет расширенный список метрик, в первую очередь ориентированный на проведение анализа системы специалистом. Кроме того, описание может быть расширено опциональным описанием известных истории случаев реализации выбранных уязвимостей. Так как в критических ситуациях у специалиста, использующего приложение, времени на ознакомление с данной информацией может не быть, эти данные в первую очередь будут полезны для нейронной сети, которая в кратчайшие сроки может предоставить информацию о ситуации, сравнить данные о текущем состоянии системы с данными из известных случаев эксплуатации уязвимости, и предоставить краткую сводку специалисту с рекомендациями по противодействию.

Аналогичным образом может быть расширена информация о CWE. Так как их описания содержат информацию в том числе о техниках эксплуатации, они могут быть дополнены за счет информации о них в базе MITTRE&ATACK.

Для реализации вышеописанных функций в программе предусмотрены функции, обеспечивающие её комфортное масштабирование. Описания CAPEC, CWE и CVE могут быть расширены путем создания новых таблиц в базе данных и связывания их с уже существующими, а также добавления новых столбцов для хранения необходимых данных. Отображение новой информации можно реализовать через добавление соответствующих вкладок в описания CAPEC/CWE/CVE, что станет возможным благодаря модульной архитектуре исходного кода приложения. Кроме того, выполнение расчетов вероятности реализации сценариев атак, а также вероятностей и ущерба от эксплуатации уязвимостей может быть реализовано в отдельной вкладке, где будут применяться соответствующие формулы и методики, предложенные экспертами в области информационной безопасности. Такой подход не только упростит доступ к аналитическим данным, но и обеспечит интуитивно понятный интерфейс для пользователей, что повысит общую эффективность работы с приложением.

На данный момент приложение использует языковую модель GPT-4 для сопоставления уязвимостей и техник их эксплуатации. Однако, несмотря на свои сильные стороны, модель имеет потенциал для доработки и улучшения, что позволит еще более эффективно решать задачи в области информационной безопасности.

С учетом широкого распространения бесплатных языковых моделей, открытых к обучению, представляется целесообразным рассмотреть интеграцию более специализированных решений. Одним из таких вариантов является языковая модель SMEET, которая, как показали исследования, является наиболее подходящей для задач

маппинга уязвимостей и техник. Это связано с её высокой точностью в анализе и сопоставлении данных [10].

Одним из значительных преимуществ SMEET является её открытый исходный код, который предоставляет возможность модификации модели согласно конкретным целям проекта. Например, можно адаптировать модель под специфические запросы и требования пользователей, что сделает её более эффективной в контексте задач компании.

Для улучшения работы с данными можно внедрить функции, отвечающие за автоматическое обновление базы данных, содержащей информацию о новейших уязвимостях и техниках их эксплуатации. Это можно реализовать через API, позволяющее получить данные из авторитетных источников, таких как NVD (National Vulnerability Database) или MITRE. Такой подход обеспечит актуальность информации и позволит пользователям получать самые свежие данные о существующих и потенциальных угрозах.

Еще одной важной функцией, которую стоит рассмотреть, являются механизмы машинного обучения. Они могут анализировать предыдущие инциденты и предсказывать вероятные уязвимости на основе текущих трендов. Это не только повысит информированность пользователей, но и позволит проактивно предотвращать возможные атаки.

Если по каким-то причинам SMEET не удовлетворит требования проекта, существует также возможность продолжать использование языковой модели GPT-4. Исходный код модели также доступен для редактирования, что открывает пространство для доработки и интеграции дополнительных модулей, обеспечивающих более глубокий анализ. Например, можно создать функции, которые будут фокусироваться на специфических аспектах информационной безопасности, таких как анализ угроз или построение сценариев атак. Это сделает работу с данными более интуитивной и обоснованной.

Кроме того, стоит задуматься о добавлении интерфейса визуализации данных. Такой интерфейс позволит пользователям более наглядно представлять результаты анализа, используя графики, диаграммы и тепловые карты. Эти инструменты могут продемонстрировать распределение уязвимостей и их потенциальное воздействие на организацию, что значительно упростит процесс принятия решений.

Таким образом, разработанная система представляет собой эффективный инструмент для анализа и получения информации, существенно упрощая работу с большими объемами данных. Она также гарантирует гибкость для будущих расширений. Интеграция различных языковых моделей и внедрение новых функций значительно повысит точность и глубину получаемых данных, делая систему более адаптивной к постоянно меняющемуся ландшафту киберугроз.

Алгоритм работы программы

Для наглядного представления алгоритма, воспользуемся структурно-функциональной схемой это графическое представление системы или процесса, которое отображает как структуру (составные части), так и функции (процессы) этой системы, которая изображена на рис. 12.

Программа является веб-приложением, предоставляющим пользователю возможность выбрать идентификатор для поиска объекта. В качестве идентификатора может выступать CAPEC/CWE/CVE. В качестве объекта – номер идентификатора.

На основе выбранных данных программа формирует ответ. В случае, если выбран CAPEC, в базе данных осуществляется поиск по введенному идентификатору. В результате программа выводит граф, состоящий из CAPEC, соответствующих ему CWE и связанных с ними CVE. Нажимая на любой из полученных элементов графа, пользователь получает краткое описание данного элемента. При этом, если это CAPEC, то будет выведено лишь его краткое описание. Этот же результат будет получен, если нажать на элемент графа, содержащий CVE.

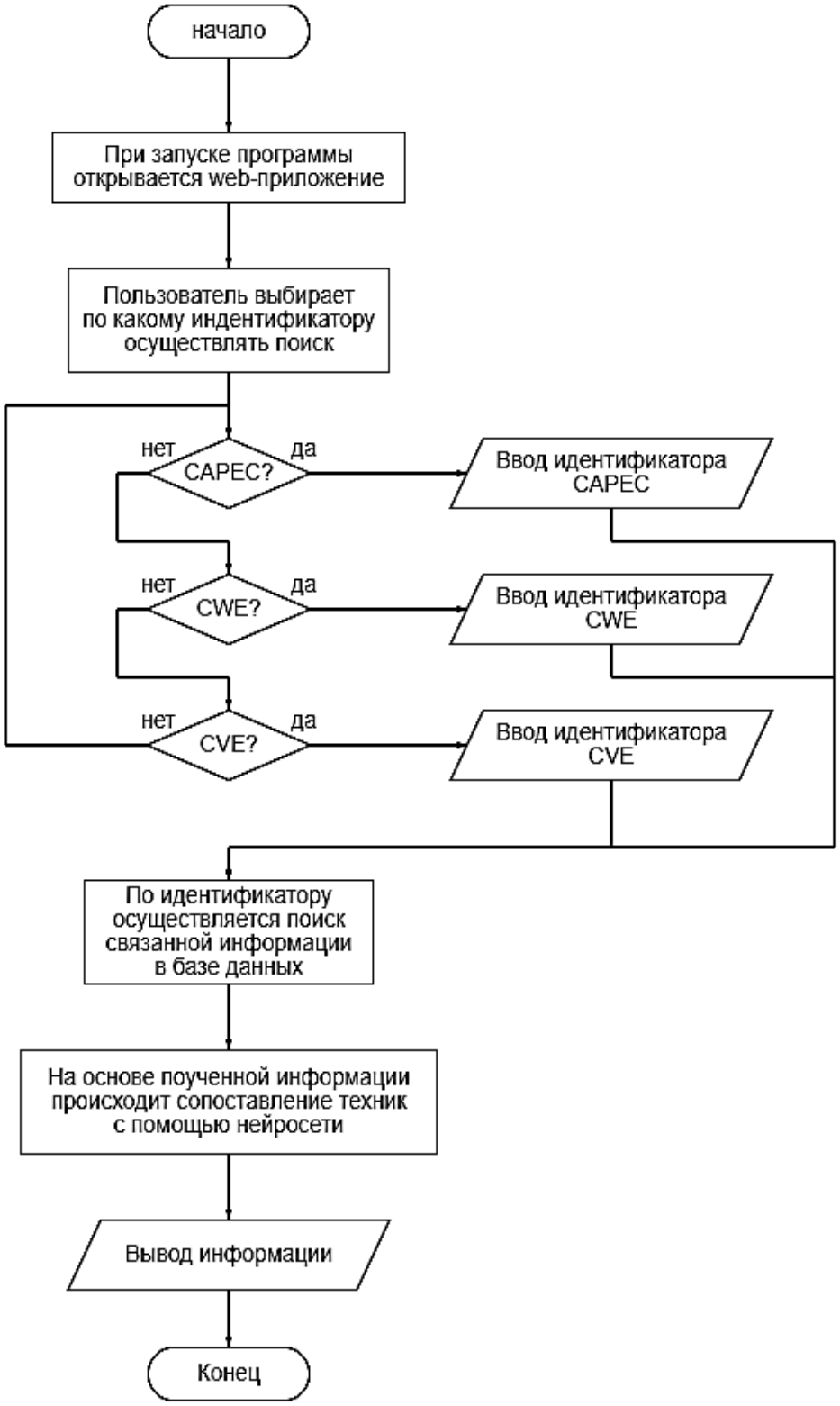

Рис. 12. Структурно-функциональная схема программы

Элементы, содержащие CWE, помимо краткого описания содержат информацию о реализуемых для данной программной ошибки техниках, а также обоснование их выбора.

Если же пользователь выбирает в качестве идентификатора CWE, то пользователь получает граф, в котором представлена выбранная CWE, а также связанные с ней CAPEC и CVE. Выбранная программная ошибка может содержать несколько уязвимостей, а также на неё могут ссылаться несколько шаблонов атаки, что отражено в графе поиска.

Выбранная CVE (Common Vulnerabilities and Exposures) генерирует в рабочей области граф, который связан с соответствующими CWE (Common Weakness Enumerations). Каждая CWE может быть связана с несколькими шаблонами атаки, что также отображается на графе. Пользователь имеет возможность выбрать любой из элементов графа, чтобы получить детальную информацию о нем.

Независимо от выбранного идентификатора, нейросетевая модель, встроенная в приложение, осуществляет сопоставление техник эксплуатации и выбранного идентификатора уязвимости. Сопоставление проводится на основании баз данных, содержащих информацию о техниках эксплуатации и связанных с ними уязвимостях. В дальнейшем предполагается доработка языковой модели, которая будет включать данные о соотнесении уязвимостей и техник эксплуатации, разработанные вручную. Это позволит повысить точность и полноту анализа, улучшая механизм генерации рекомендаций по устранению уязвимостей и минимизации рисков, связанных с кибератаками. Данная интеграция обеспечит более глубокое понимание взаимосвязей между уязвимостями и техниками их эксплуатации, что в свою очередь окажет влияние на эффективность стратегий защиты и предотвращения инцидентов в будущем.

Заключение

В ходе исследования была разработана автоматизированная система для поиска соответствий между уязвимостями, шаблонами и техниками кибератак, что значительно повышает уровень безопасности информационных систем. Использование машинного обучения и интеграция авторитетных баз данных, таких как MITRE ATT&CK, MITRE CWE, БДУ ФСТЭК и MITRE EMB3D, позволили создать эффективные методические и алгоритмические решения для анализа уязвимостей.

Автоматизация процесса сопоставления не только улучшает точность идентификации уязвимостей и связанных с ними атак, но также минимизирует риск принятия неэффективных решений при проектировании систем защиты информации. Экспериментальные результаты подтвердили, что предложенные методы могут существенно сократить время, необходимое для анализа, и повысить нужды специалистов по кибербезопасности.

Перспективы дальнейших исследований заключаются в постоянном обновлении методических подходов, необходимых для адаптации к новым угрозам и уязвимостям, а также в разработке механизмов для автоматического обучения системы на основе новых данных. Это сделает инструменты по анализу уязвимостей более гибкими и эффективными, способствуя созданию безопасной информационной среды

Список литературы

1. Mapping Vulnerability Description to MITRE ATT&CK Framework by LLM// URL: https://assets-eu.researchsquare.com/files/rs-4341401/v1_covered_59acb6fd-68d0-425e-8041-666108000204.pdf?c=1716358479 (дата обращения: 12.01.2025).

2. Mapping MITRE ATT&CK to CVE using NLP, BERT and PyTorch URL: https://systemweakness.com/mapping-mitre-att-ck-to-cve-using-nlp-bert-and-pytorch-63b8d495589a (дата обращения: 12.01.2025).

3. База знаний о тактике и методах злоумышленников MITRE ATT&CK. // URL: https://attack.mitre.org/ (дата обращения: 12.01.2025).

4. Классификация программно-аппаратных ошибок MITRE CWE. // URL: https://cwe.mitre.org/ (дата обращения: 12.01.2025).

5. База данных уязвимостей (БДУ) ФСТЭК. // URL: https://bdu.fstec.ru/vul (дата обращения: 12.01.2025).

6. Модель угроз для встроенных устройств MITRE EMB3D. // URL: https://emb3d.mitre.org/ (дата обращения: 12.01.2025).

7. CVE mapping meth // URL: https://center-for-threat-informed-defense.github.io/mappings-explorer/about/methodology/cve-methodology/ (дата обращения: 12.01.2025).

8. MITRE. Mapping ATT&CK to CVE for Impact. // URL: https://mitre-

engenuity.org/cybersecurity/center-for-threat-informed-defense/our-work/mapping-attck-to-cve-for-impact/ (дата обращения: 12.01.2025).

9. Система управления базами данных PostgreSQL. // URL: https://www.postgresql.org/ (дата обращения: 12.01.2025).

Воронежский государственный технический университет
Voronezh State Technical University

Поступила в редакцию 19.01.2025

Информация об авторах

Остапенко Александр Григорьевич – д-р техн. наук, профессор, заведующий кафедрой, Воронежский государственный технический университет, e-mail: alexostap123@gmail.com

Макаров Юрий Вадимович – студент, Воронежский государственный технический университет, e-mail: makarov0130@gmail.com

Нархов Дмитрий Андреевич – аспирант, Воронежский государственный технический университет, e-mail: alexanderostapenkoias@gmail.com

Нетребин Александр Евгеньевич – студент, Воронежский государственный технический университет, e-mail: alexandr.netrebin@mail.ru

Исаев Никита Андреевич – студент, Воронежский государственный технический университет, e-mail: dodde99@yandex.ru

Зинченко Данил Романович – студент, Воронежский государственный технический университет, e-mail: dzinchenko568@gmail.com

Чаусов Дмитрий Игоревич – студент, Воронежский государственный технический университет, e-mail: dm.chausov@mail.ru

AUTOMATED SEARCH FOR MATCHING VULNERABILITIES, PATTERNS AND TECHNIQUES OF CYBER ATTACKS

A.G. Ostapenko, Yu.V. Makarov, D.A. Narkhov, A.E. Netrebin, N.A. Isaev, D.R. Zinchenko, D.I. Chausov

The article suggests programmatically forming a procedure for determining the compliance of vulnerability patterns and techniques, which often leads to inadequate solutions for the information security system developer when searching for the above-mentioned compliance, which is the most important in all subsequent calculations and the resulting design reactions. It is proposed to resolve these contradictions through the development of appropriate methodological and algorithmic software focused on machine learning with the dataset available in this space, followed by the use of the intellectual capabilities of neural networks. In this regard, the acquisition and processing of data on existing vulnerabilities, as well as their relationship with known attack techniques, are becoming key aspects for the formation of a reliable automated.

Keywords: vulnerability, techniques, attack pattern, software and hardware errors, CAPEC, CWE, CVE, neural network.

Submitted 19.01.2025

Information about the authors

Alexsandr G. Ostapenko – Dr. Sc. (Technical), Professor, Head of Department, Voronezh State Technical University, e-mail: alexostap123@gmail.com

Yuriy V. Makarov – student, Voronezh State Technical University, e-mail: makarov0130@gmail.com

Dmitry A. Narhov – graduate student, Voronezh State Technical University, e-mail: alexanderostapenkoias@gmail.com

Alexsandr E. Netrebin – student, Voronezh State Technical University, e-mail: alexandr.netrebin@mail.ru

Nikita A. Isaev – student, Voronezh State Technical University, e-mail: dodde99@yandex.ru

Danil R. Zinchenko – student, Voronezh State Technical University, e-mail: dzinchenko568@gmail.com

Dmitriy I. Chausov – student, Voronezh State Technical University, e-mail: dm.chausov@mail.ru