БЕЗОПАСНОСТЬ ЦИФРОВЫХ ТЕХНОЛОГИЙ. – 2025. – № 3 (118). – 40–61

МЕТОДЫ И СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ, ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

УДК 004 DOI: 10.17212/2782-2230-2025-3-40-61

РАЗРАБОТКА СИСТЕМЫ МОДЕЛИРОВАНИЯ КИБЕРАТАК ДЛЯ ВЫЯВЛЕНИЯ УЯЗВИМОСТЕЙ *

Д.А. ФОМИН[1], И.Л. РЕВА[2]

[1] *630559, РФ, г. Новосибирск, наукоград Кольцово, 35, ЗАО «ЦФТ», инженер по безопасной разработке программного обеспечения. E-mail: evtdanilf@yandex.ru*
[2] *630073, РФ, г. Новосибирск, пр. К. Маркса, 20, Новосибирский государственный технический университет, доцент кафедры защиты информации. E-mail: reva@corp.nstu.ru*

Поддержание необходимого уровня защищенности информационных систем остается ключевой задачей для организаций любых масштабов. Традиционные методы, такие как тестирование на проникновение (пентест), позволяют обнаружить уязвимости до их эксплуатации злоумышленниками, однако часто проводятся нерегулярно и не охватывают все векторы атак. Решением этой задачи является использование систем моделирования кибератак, которые могут использоваться как для выявления уязвимостей в IT-инфраструктуре, так и для тестирования эффективности средств защиты информации.

В статье предложена и реализована методика автоматизированного моделирования кибератак (Breach and Attack Simulation, BAS) на основе поэтапного сканирования веб-приложений. Описаны этапы разведки, фаззинг директорий и шаблонное сканирование уязвимостей с помощью различных инструментов и сканеров, каждый из которых играет свою роль в процессе моделирования кибератаки.

Ключевые слова: информационная безопасность, моделирование кибератак, уязвимости, кибербезопасность, пентест, тестирование на проникновение, анализ защищенности

ВВЕДЕНИЕ

Информационные технологии стали неотъемлемой частью современного мира, обеспечивают эффективную работу различных организаций и предприятий. Однако их повсеместное использование увеличивает риски, связанные

* Статья получена 20 июня 2025 г.

с угрозами информационной безопасности. Утрата конфиденциальности, целостности или доступности данных может привести к серьезным финансовым и репутационным потерям.

По данным Positive Technologies, в IV квартале 2024 года количество инцидентов увеличилось на 5 % по сравнению с предыдущим кварталом и на 13 % в сравнении с аналогичным периодом прошлого года [1].

Эти тенденции обостряют задачу своевременного обнаружения уязвимостей и повышения готовности IT-инфраструктуры к отражению атак. Для предотвращения подобных рисков необходимо регулярно оценивать защищенность информационных систем.

Отчет группы компаний «Солар» показывает, что в I квартале 2025 года зафиксировано 801,2 млн веб-атак на сайты российских компаний – это в 2 раза больше, чем за аналогичный период прошлого года [2]. Уязвимости в веб-приложениях способны компрометировать не только отдельные сайты, но и всю инфраструктуру компании, поэтому автоматизация их регулярного сканирования и анализа приобретает первоочередное значение.

Традиционно организации выполняют сканирование на уязвимости, тесты на проникновение (пентесты) и имитацию комплексных сценариев атак, которые проводятся специализированными «красными» командами (red team). Однако подобные мероприятия зачастую организуются нерегулярно. Одной из актуальных задач в этом направлении становится автоматизация анализа защищенности и систематического запуска симуляции действий злоумышленников.

Новым подходом в этой области стали системы моделирования кибератак, известные как BAS (Breach and Attack Simulation). Эти системы предназначены для автоматического проведения имитационных атак, что позволяет обнаруживать уязвимости, а также проверять эффективность всех средств защиты в комплексе.

Целью исследования является создание системы моделирования кибератак, объединяющей последовательные этапы сканирования веб-приложений с целью выявления уязвимостей.

Для реализации поставленной задачи необходимо решить следующие задачи:

- изучить актуальные уязвимости веб-приложений;
- проанализировать современные решения BAS;
- разработать программное обеспечение, провести его апробацию.

1. УЯЗВИМОСТИ ВЕБ-ПРИЛОЖЕНИЙ

Кибератаки – это умышленные действия злоумышленников, направленные на нарушение работы информационных систем, кражу данных или нарушений их целостности.

Типы атак делятся на несколько категорий.

1. Сетевые атаки – это, например, сканирование портов (при помощи сетевых сканеров), атаки типа DDoS (отказ в обслуживании), перехват трафика (Man-in-the-Middle).

2. Атаки на веб-приложения – это любые способы злоупотребления функциональностью клиент-серверного взаимодействия для получения несанкционированного доступа, модификации или кражи данных и дестабилизации работы сервиса. Классическим примером является внедрение вредоносного кода через пользовательский ввод: межсайтовый скриптинг (XSS), SQL-инъекции, подделка межсайтовых запросов (CSRF и SSRF). Любой интерактивный элемент веб-приложения – форма ввода, поле поиска, загрузка файла – потенциально может быть использован злоумышленником для атаки.

3. Социальная инженерия – это фишинг (подделка сайтов и рассылка вредоносных ссылок), распространение троянов и другого вредоносного ПО на устройства пользователей.

4. Внутренние угрозы – это случаи, когда атака исходит от недобросовестного сотрудника или по неосторожности администратора (например, из-за слабых паролей или ошибок в конфигурации).

Успешная атака обычно включает в себя несколько последовательных этапов, которые объединены в концепцию Kill Chain.

1. Внешняя разведка – сбор общей информации о целевой сети и сервисах (например, сканирование портов).

2. Вооружение – подготовка вредоносных модулей (например, формирование эксплойтов и оснащение вредоносным ПО).

3. Доставка – передача вредоносного ПО жертве через выбранный вектор (например, фишинговое письмо).

4. Заражение – активация эксплойта в целевой системе.

5. Установка – развертывание удаленного доступа.

6. Получение управления – установление канала связи между зараженным узлом и сервером злоумышленника.

7. Выполнение действий у жертвы – отправка собранных данных, вывод из строя IT-активов, выявление других целей и дальнейшее продвижение по сети [3].

В современных условиях часто целью злоумышленников становятся именно веб-приложения, так как они открыты извне, обрабатывают пользовательские данные и связаны с критически важными сервисами и базами данных.

Наиболее распространенные и опасные уязвимости веб-приложений представлены в OWASP Top 10 – списке уязвимостей, который публикует открытое сообщество OWASP [4].

1. *Недостатки контроля доступа* (*Broken Access Control*)

Уязвимости, позволяющие злоумышленникам обходить авторизацию. Классический пример – изменение идентификатора ресурса в URL (IDOR), приводящее к просмотру или изменению чужих данных.

2. *Криптографические сбои* (*Cryptographic Failures*)

Это риск, который связан с тем, что конфиденциальная информация хранится или передается без должного шифрования, либо используются устаревшие алгоритмы. Сюда относится, например, использование устаревших алгоритмов хеширования md5 или sha-1, в том числе без соли. В результате критичные данные (пароли, токены) могут быть скомпрометированы.

3. *Инъекции* (*Injection*)

Уязвимости, при которых необработанный ввод пользователя вставляется в команды или запросы, что позволяет выполнять произвольный код или читать и изменять данные. Примером могут служить SQL-инъекции, которые иногда могут приводить к компрометации данных в базах данных [5].

4. *Небезопасный дизайн* (*Insecure Design*)

Проблемы, вытекающие из изначально неправильной проработки бизнес-логики или требований безопасности. Например, подтверждение личности для восстановления учетных данных через «секретный вопрос». Такой подход не является безопасным, так как ответы могут знать несколько человек.

5. *Небезопасные настройки безопасности* (*Security Misconfiguration*)

Параметры серверов, фреймворков или компонентов, оставшиеся по умолчанию или открытые избыточно (например, включенный режим отладки в веб-фреймворке Django, раскрытие каталога «.git» системы контроля версий Git).

6. *Уязвимые и устаревшие компоненты* (*Vulnerable and Outdated Components*)

Использование зависимостей или модулей с известными уязвимостями без своевременного обновления. Пример – использование уязвимых версий

библиотеки Apache Log4j. Одна из таких уязвимостей – CVE-2021-4422, которая приводит к удаленному выполнению вредоносного кода.

7. *Ошибки аутентификации и идентификации* (*Identification and Authentication Failures*)

Недостатки в механизмах входа в систему и управления сессионными токенами: предсказуемые идентификаторы или отсутствие ограничения числа попыток авторизации.

8. *Нарушения целостности ПО и данных* (*Software and Data Integrity Failures*)

Отсутствие проверки подлинности загружаемых модулей или обновлений, при которых злоумышленник может внедрить вредоносный код через обновление ПО.

9. *Сбои в мониторинге и журналировании* (*Security Logging and Monitoring Failures*)

Отсутствие или неполнота регистрации событий безопасности (неудачные попытки входа, запросы с подозрительными параметрами), что затрудняет своевременное обнаружение и расследование инцидентов.

10. *Подделка запросов на стороне сервера* (*SSRF*)

Уязвимость, при которой веб-сервер по команде злоумышленника выполняет запросы к внутренним или внешним ресурсам, что может приводить к обходу сетевых фильтров, доступу к внутренним сервисам и их компрометации.

Основную часть уязвимостей из OWASP Top 10 призваны обнаруживать сканеры уязвимостей. Но не все уязвимости можно обнаружить при помощи сканеров, и для их поиска тоже нужен ручной поиск [6]. Так, обнаружение уязвимостей бизнес-логики и небезопасного дизайна архитектуры приложения (A04) тяжело автоматизировать, так как нужно учитывать процессы работы приложения. К примеру, к уязвимостям бизнес-логики можно отнести ситуацию, когда злоумышленник многократно применяет один и тот же промокод, тем самым искусственно увеличивая размер скидки, или многократно отменяет оплаченные бонусными баллами заказы для наращивания бонусного баланса.

Также сложности вызывает проверка контроля доступа (A01), поскольку корректная валидация прав пользователя чаще всего зависит от сложной логики маршрутизации и ролей, что бывает трудно свести к набору шаблонных HTTP-запросов.

Кроме того, автоматизированному сканированию недоступна верификация целостности программного обеспечения и данных (A08) – здесь необхо-

дима проверка цифровых подписей и безопасности цепочек поставок. Не менее критичным является и оценка систем журналирования и мониторинга (A09): для понимания полноты и корректности логов требуется ручной аудит настроек лог-систем и анализа событий.

Поэтому для полного покрытия всех векторов атак необходим комбинированный подход, сочетающий автоматизированное сканирование и экспертный ручной анализ.

2. АНАЛИЗ СОВРЕМЕННЫХ РЕШЕНИЙ BAS

Breach and Attack Simulation (BAS) – это системы для автоматизированного непрерывного моделирования кибератак на IT-инфраструктуру организации. BAS-системы позволяют запускать заранее подготовленные сценарии атак, имитирующие действия злоумышленников, включая разведку, проникновение, движение по сети и эксфильтрацию данных [7].

Преимущества BAS-систем заключаются в возможности регулярно проверять защищенность без прерывания рабочего процесса. Благодаря автоматизации запусков сценариев имитации атак можно получать актуальную картину состояния безопасности в режиме реального времени и сфокусироваться на слабых местах. Автоматизация в данном случае позволяет выполнять рутинные задачи быстрее, что снижает нагрузку на специалистов по информационной безопасности и минимизирует человеческий фактор при регулярных проверках.

Типичные компоненты решений BAS:

- репозиторий сценариев атак – возможные векторы атак, которые заложены разработчиком в систему. Они имитируют атаки на веб-приложения, инфраструктуру, эскалацию привилегий и многое другое, в том числе атаки на основе матрицы MITRE ATTACK;
- агенты-исполнители – модули, развертываемые на серверах, контейнерах или виртуальных машинах для локального запуска сценариев атак и сбора результатов;
- оркестратор – центральный сервис, распределяющий задачи по агентам, управляет расписанием запусков и проверками;
- панель результатов и аналитики – веб-интерфейс для оценки обнаруженных уязвимостей и результатов атак.

Также у некоторых решений поддерживается интеграция с SIEM для передачи данных об атаках.

BAS отличается от традиционного пентеста и сканирования уязвимостей тем, что моделирует реальные атаки во множестве разнообразных векторов.

При этом BAS может работать непрерывно и в автоматическом режиме, мгновенно уведомляя о новых проблемах [8].

Однако не все решения BAS реализуют полную цепочку Kill Chain. Некоторые из систем сосредоточены на конкретных атаках, большинство являются платными, а также имеют другие особенности, которые представлены в табл. 1.

Таблица 1

Сравнение популярных решений BAS

Название	Особенности	Полное покрытие Kill Chain	Модель распространения
MITRE Caldera	Интеграция с MITRE ATTACK, поддержка пользовательских сценариев, фокус на Red Team	Да	Открытый исходный код
Cymulate	Облачная платформа, симуляция фишинга, интеграция с SIEM	Да	Коммерческая подписка
SafeBreach	Визуализация по MITRE ATTACK	Да	Коммерческая подписка
AttackIQ	Поддержка MITRE ATTACK, визуализация рисков, интеграция с SIEM, EDR	Да	Коммерческая подписка
Randori	Фокус на Red Team, составление поверхности атаки для приоритизации целей	Частичное (внешняя разведка и заражение)	Коммерческая подписка

Несмотря на высокую степень автоматизации, BAS пока не в состоянии полностью заменить ручные пентесты, особенно в части поиска уязвимостей, которые сложно автоматизировать, и векторов с социальной инженерией [9].

3. РЕАЛИЗАЦИЯ СИСТЕМЫ МОДЕЛИРОВАНИЯ КИБЕРАТАК

3.1. ТЕХНОЛОГИИ РЕАЛИЗАЦИИ СИСТЕМЫ

Благодаря огромному числу библиотек, а также простоте разработки и универсальности Python стал естественным выбором для разработки системы. Кроме того, Python широко используется в области информационной

безопасности, большое количество инструментов разработано на этом языке. Для реализации логики API и клиентской части был выбран Flask, так как это легкий фреймворк, который позволяет быстро реализовать маршрутизацию и обработку запросов.

Для выполнения длительных операций требуется запуск их в асинхронном режиме, чтобы не останавливать работу самой системы.

Для этой цели был выбран Celery – это распределенная система управления задачами с открытым исходным кодом и активной поддержкой сообщества. Она позволяет отправлять функции на выполнение в отдельные потоки или процессы, используя брокер очередей (например, RabbitMQ или Redis) для передачи сообщений.

Celery позволяет легко масштабироваться за счет возможности добавлять обработчики, что позволит выполнять больше задач за единицу времени.

Redis выбран в качестве брокера сообщений, для доставки заданий очереди заданий, так как проще в освоении, чем RabbitMQ, и не требует сложной настройки, а также позволяет легко интегрировать отправку сообщений в функционал динамического обновления данных в веб-интерфейс.

В качестве хранилища результатов сканирования и метаданных задач выбрана нереляционная СУБД MongoDB, которая хранит данные в формате JSON-подобных документов. Это удобно для результатов сканирования, поскольку схема может гибко меняться по ходу работы (добавление новых полей, разных типов данных). Отсутствие жесткой схемы в MongoDB обеспечивает гибкость, а встроенная поддержка репликации и шардирования (размещения коллекций с данными на разных хостах) упрощает масштабирование.

Для обеспечения изоляции модулей между собой было решено использовать Docker, что также позволит обеспечить единую среду выполнения на всех этапах жизненного цикла приложения – от разработки до эксплуатации.

3.2. ВЫБОР ИНСТРУМЕНТОВ И СКАНЕРОВ

Для реализации системы моделирования кибератак были выбраны инструменты, обеспечивающие разведку целей и обнаружение уязвимостей.

В качестве сетевого сканера портов был выбран Nmap. В качестве аналога можно рассмотреть Masscan, но он не подходит, так как рассчитан на сканирование больших сетей, само сканирование хоть и проходит быстрее, чем у Nmap, но результат менее точный и ограничен базовым функционалом, то есть утилита не имеет возможности, например, определять версии служб.

Nmap же поддерживает гибкую настройку, имеет расширенный функционал за счет встроенных скриптов определения сервисов, их версий и даже уязвимостей.

Для поиска поддоменов используется два подхода. Сначала выполняется пассивный сбор. Для этого была выбрана утилита Subfinder, предназначенная для пассивного поиска поддоменов и оптимизированная для высокой скорости работы и легкой интеграции, что явилось главной причиной выбора именно этой утилиты. Она обращается к публичным API (DNS-ресурсы, онлайн-сервисы, собирающие информацию о поддоменах) и формирует список обнаруженных поддоменов.

Второй подход заключается в активном поиске поддоменов. Для этого используется утилита для фаззинга, которая перебирает по словарю различные поддомены для цели и фиксирует корректные ответы сервера, что позволяет обнаружить скрытые поддомены, которые невозможно найти пассивным подходом.

С целью активного HTTP-сканирования и разведки веб-сервисов была выбрана утилита httpx. Она превосходит привычную утилиту curl за счет встроенной поддержки массовых параллельных запросов, гибкой настройки по различным параметрам и структурированного вывода, упрощающего интеграцию в систему и обработку результатов.

Данная утилита позволяет обнаруживать веб-сервисы на ранее найденных сетевых портах, собирать информацию о веб-сайтах, а также проверять доступность веб-сервиса, что позволяет, например, проверить результат поиска поддоменов.

Для поиска уязвимостей был выбран подход с шаблонными сканерами, что позволит описывать специфический вектор атаки или проверку на уязвимость в виде отдельного шаблона. Для этих целей был выбран сканер Nuclei. Это легкий и настраиваемый сканер уязвимостей с открытым исходным кодом, основанный на шаблонах. Сканер быстро работает за счет того, что написан на языке программирования Go, позволяет детектировать уязвимости на разных протоколах (HTTP, TCP, DNS и другие) с минимальным числом ложных срабатываний. Также он легко обновляется через публичный репозиторий.

Выбор в пользу Nuclei обусловлен его активным развитием и наличием большого сообщества, что гарантирует регулярное обновление шаблонов и оперативное устранение ошибок, а также возможностью добавлять новые сценарии проверки путем разработки пользовательских шаблонов.

При сравнении с другими бесплатными решениями следует отметить следующее:

- Wapiti практически не поддерживается сообществом и редко получает обновления, из-за чего база проверок быстро устаревает;
- OWASP ZAP требует значительных системных ресурсов, часто порождает большое количество ложных срабатываний и затрудняет автоматизацию за счет сложной настройки;

• Nikto обладает ограниченной возможностью добавления собственных правил, что не позволит легко расширять функционал.

Для фаззинга директорий и активного перебора поддоменов используется ffuf. В отличие от Gobuster или Dirb, ffuf поддерживает гибкую настройку, автоматическую подстановку заголовков и большое количество форматов словарей. Его структура вывода позволяет легко обрабатывать результаты и интегрировать в разрабатываемую систему.

3.3. АРХИТЕКТУРА СИСТЕМЫ

Система моделирования кибератак спроектирована по модульному принципу и реализована в клиент-серверной архитектуре. Такое разделение обеспечивает независимое развитие и масштабирование каждого компонента, а также упрощает добавление новых модулей.

Связь модулей системы представлена на рис. 1.

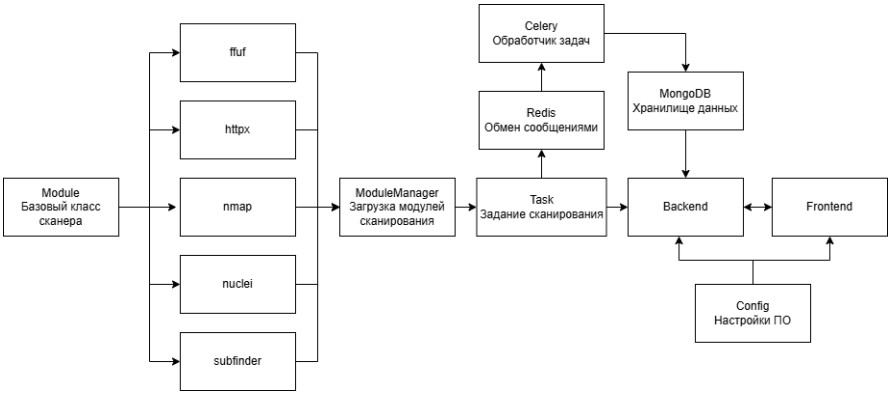

Рис. 1. Связь модулей системы

Система состоит из следующих компонентов:

• компонент Module представляет собой базовый класс для модулей сканирования;

• модуль ModuleManager отвечает за обнаружение и загрузку модулей сканирования;

• компонент Task инкапсулирует все параметры одного задания (цели, список модулей, опции), ставит его в очередь, отслеживает выполнение каждого этапа и обновляет накопленные результаты и статус;

- Redis выступает брокером сообщений (обеспечивает связь между ПО и обработчиками Celery) и каналом для публикации данных в режиме реального времени;
- Celery распределяет задачи между доступными обработчиками и гарантирует их асинхронное выполнение;
- MongoDB хранит информацию о задачах, а также результаты сканирований;
- Config централизованно управляет настройками приложения и отдельных сканирующих модулей;
- модуль Backend является серверной частью, реализующей REST API, авторизацию для API и обработку задач;
- компонент Frontend является веб-интерфейсом системы.

3.4. АЛГОРИТМ МОДЕЛИРОВАНИЯ КИБЕРАТАК

На рис. 2 показан алгоритм симуляции кибератак на веб-приложения.

Рис. 2. Алгоритм симуляции кибератак

Поэтапное сканирование выполняется различными инструментами, которые автоматизируют внешнюю разведку и поиск уязвимостей.

Этапы работы системы

1. *Поиск поддоменов*

Пассивный поиск поддоменов выполняется с Subfinder. Утилита делает запросы к онлайн-источникам, которые позволяют оперативно получить список известных поддоменов. Активный поиск заключается в том, что по заранее подготовленному словарю проводится фаззинг возможных поддоменов. Корректные ответы сервера фиксируются и добавляются в перечень исследуемых целей.

2. *Поиск открытых портов*

Для каждой обнаруженной цели выполняется поиск открытых сетевых портов и определение версий сервисов при помощи сетевого сканера Nmap.

3. *Обнаружение HTTP-сервисов*

На обнаруженных сетевых портах при помощи httpx выполняется поиск веб-служб, собирается базовая информация о них, такая как заголовок сайта, код ответа и т. д.

4. *Фаззинг директорий веб-приложения*

С помощью словарей запускается перебор каталогов и файлов на обнаруженных веб-сервисах, что позволяет выявить скрытые и нестандартные интерфейсы, административные панели, временные файлы и другие ценные точки входа.

5. *Шаблонное сканирование на уязвимости*

Все найденные URL и веб-сервисы сканируются при помощи Nuclei, что позволяет обнаружить уязвимости в веб-приложениях.

3.5. РЕАЛИЗАЦИЯ ВЕБ-ИНТЕРФЕЙСА

При входе в систему пользователя встречает страница авторизации. Для входа используется имя пользователя и пароль, которые были указаны в конфигурационном файле при запуске системы.

После успешной авторизации появится главная страница веб-интерфейса (рис. 3). Веб-интерфейс имеет боковую панель, которая позволяет легко переходить между страницами. Также на этой странице есть две функциональные кнопки и три панели.

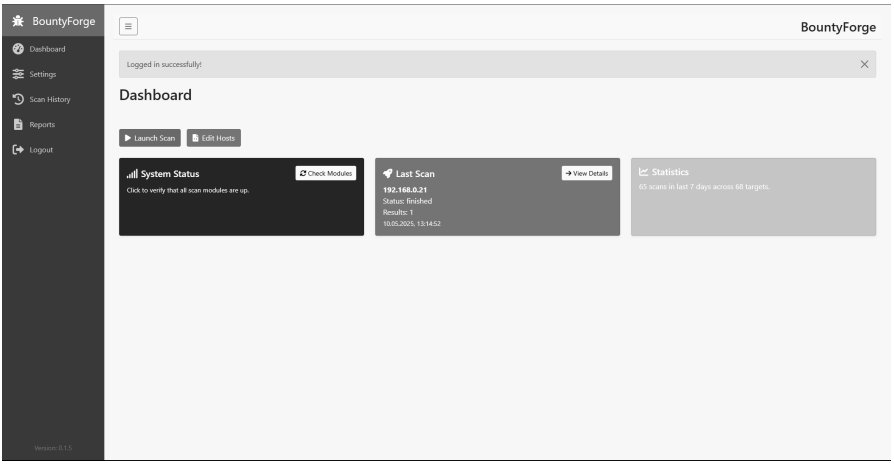

Рис. 3. Главная страница

Кнопка Launch Scan позволяет запустить сканирование, при нажатии на кнопку появляется окно запуска сканирования (рис. 4). Здесь можно указать цель для атаки и цели, которые необходимо исключить из сканирования, дополнительные заголовки, например, для авторизации, а также выбрать инструменты, которые будут запущены.

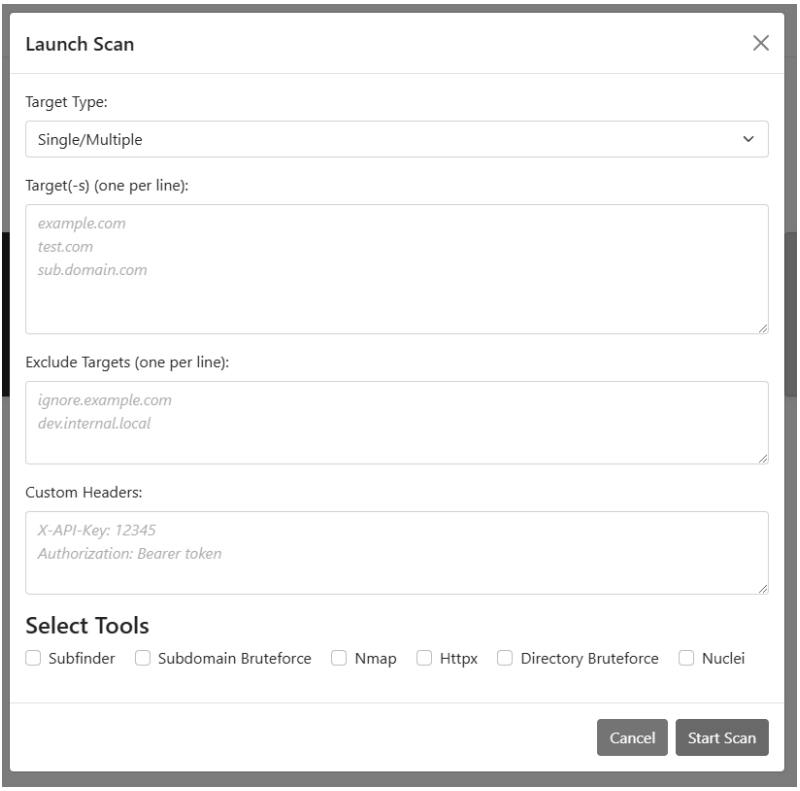

Рис. 4. Окно запуска задачи

Кнопка Edit Hosts позволяет открыть окно для редактирования файла /etc/hosts для контейнера, в котором запускаются инструменты и сканеры (рис. 5). Это позволяет, например, указать виртуальный хост, который недоступен при обращении по прямому адресу без его явного указания в локальном файле соответствия IP-адреса к доменному имени.

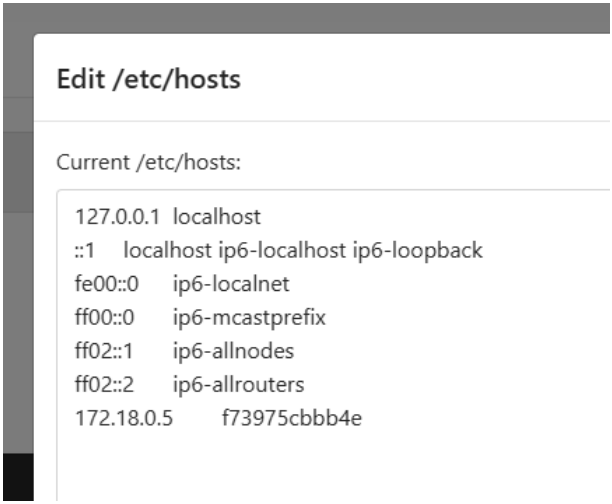

Рис. 5. Редактирование файла хостов

Панель System Status позволяет проверить работоспособность модулей системы, а также узнать их версию при нажатии на соответствующую кнопку.

Панель Last Scan показывает краткую сводку по последнему запуску и имеет кнопку для навигации на соответствующую страницу с информацией.

Панель Statistics показывает количество запусков и целей за последние 7 дней.

На странице настройки модулей Settings пользователь может гибко управлять параметрами каждого из интегрированных сканеров и инструментов (рис. 6). Интерфейс сгруппирован для каждой утилиты отдельно, позволяет быстро находить нужный модуль и задавать для него конфигурацию. Доступен выбор режима работы модулей, словарей, шаблонов для Nuclei, а также есть возможность указать дополнительные флаги командной строки для более гибкой настройки. Помимо этого, есть общие параметры настройки, такие как количество отправляемых в секунду запросов и время ожидания ответа на запросы.

Страница с деталями сканирования представляет собой страницу с информацией о процессе выполнения задачи в виде вывода каждого модуля (рис. 7).

Рис. 6. Страница настройки модулей

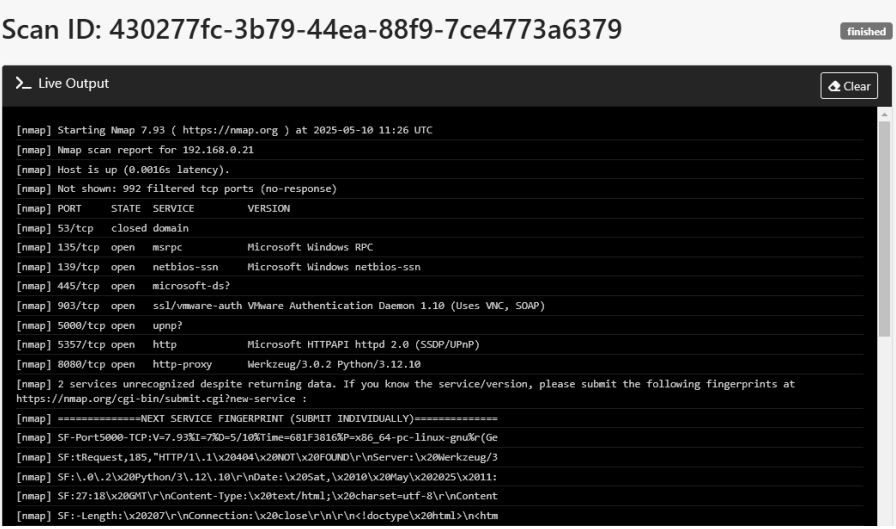

Рис. 7. Страница с деталями задачи

Страница Scan Report позволяет просматривать результаты поэтапного сканирования: слева отображаются найденные веб-сайты, их пути, а также поддомены. С правой стороны находится визуальная статистика по критичности уязвимостей, а также список уязвимостей (рис. 8).

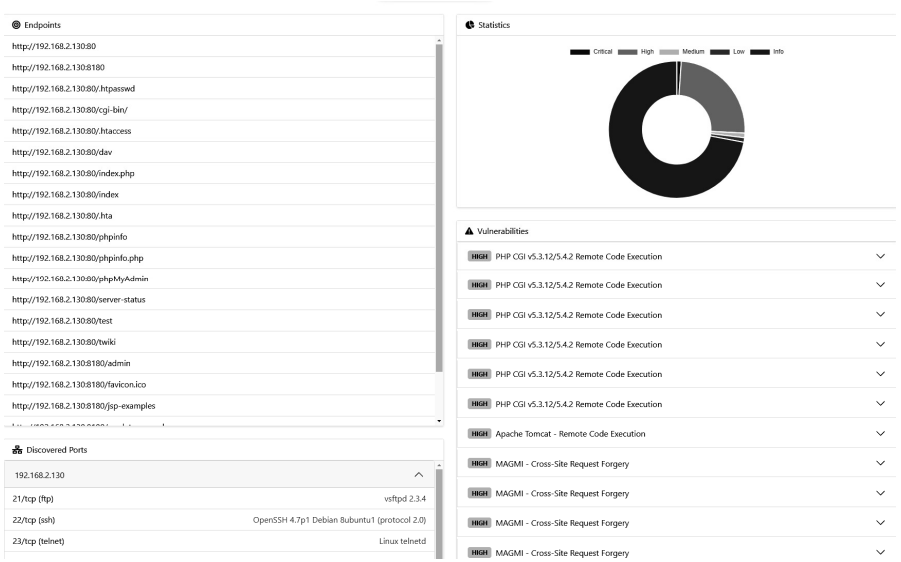

Рис. 8. Страница результатов выполнения задачи

3.6. АПРОБАЦИЯ СИСТЕМЫ

При проведении тестирования системы была использована специальная тестовая среда для тестирования на проникновение – виртуальная машина metasploitable2, которая содержит преднамеренно уязвимые сетевые сервисы (включая уязвимый веб-сайт).

В ходе тестирования нагрузка на сканирующие модули не должна превышать 100 запросов в секунду, все остальные настройки были оставлены по умолчанию.

Использование оперативной памяти не превысило 2 Гб, поэтому можно судить о том, что система получилась нетребовательной к системным ресурсам.

Пиковое использование системных ресурсов во время этапа обнаружения уязвимостей представлено на рис. 9.

```
NAME                CPU %     MEM USAGE / LIMIT
bf-frontend         0.22%     87.14MiB / 7.463GiB
bf-celery-worker    9.56%     1.344GiB / 7.463GiB
bf-backend          0.44%     98.63MiB / 7.463GiB
bf-mongo            0.34%     206.6MiB / 7.463GiB
bf-redis            0.44%     10.52MiB / 7.463GiB
```

Рис. 9. Потребление ресурсов

После завершения работы системы был сформирован отчет по результатам всех этапов сканирования, который отображен на рис. 10.

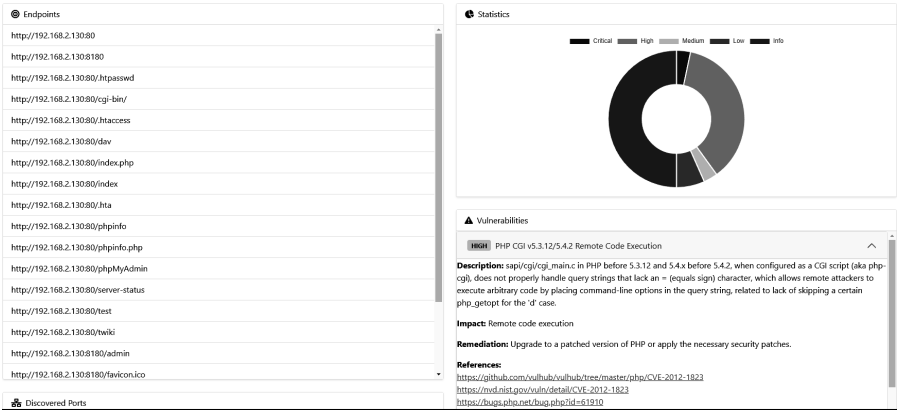

Рис. 10. Визуализация результатов в веб-интерфейсе

Основные показатели по итогам эксперимента представлены в табл. 2.

Таблица 2

Результаты тестирования

Метрика	Значение
Общее время сканирования в секундах	399
Общее число найденных уязвимостей	30
Общее число обнаруженных конечных точек	21
Общее число обнаруженных открытых сетевых портов	22

В табл. 3 представлено соотношение обнаруженных уязвимостей по критичности. Было проведено сравнительное сканирование при помощи сканера Nuclei без этапа разведки, а также дополнительно запускался сканер уязвимостей OWASP Zap.

Таблица 3

Статистика по обнаруженным уязвимостям

Критичность уязвимостей	Обнаружено уязвимостей системой	Обнаружено уязвимостей при помощи Nuclei	Обнаружено уязвимостей при помощи OWASP Zap
Критическая	1	1	0
Высокая	11	9	0
Средняя	1	1	6
Низкая	2	2	8
Информационная	15	10	7

ЗАКЛЮЧЕНИЕ

В результате исследования была разработана система моделирования кибератак для автоматизированного выявления уязвимостей. Эта система позволяет повысить уровень защищенности IT-инфраструктуры компаний.

Разработанная система показывает эффективность относительно обычных сканеров уязвимостей веб-приложений, так как смогла обнаружить критические уязвимости, в том числе благодаря этапу разведки.

Практическая значимость работы заключается в том, что внедрение разработанной системы позволяет автоматизировать регулярные проверки веб-приложений, а также проверять эффективность внедренных СЗИ.

Перспективы развития заключаются в добавлении планировщика для автоматического запуска задач по расписанию, в расширении набора модулей сканирования, внедрении механизма автоматизированной оценки приоритетности обнаруженных уязвимостей на основе их потенциального риска и вероятности эксплуатации, что позволит сделать систему более адаптивной и универсальной.

СПИСОК ЛИТЕРАТУРЫ

1. *Голушко А.* Актуальные киберугрозы: IV квартал 2024 года – I квартал 2025 года // Positive Technologies: сайт. – 2025, 20 марта. – URL: https://ptsecurity.com/ru-ru/research/analytics/aktualnye-kiberugrozy-iv-kvartal-2024-goda-i-kvartal-2025-goda (дата обращения: 02.09.2025).

2. ГК «Солар»: число веб-атак на сайты российских компаний за год выросло в 2 раза // Solar: сайт. – URL: https://rt-solar.ru/events/news/5498/ (дата обращения: 02.09.2025).

3. *Бойченко О.В.* Модель многоступенчатой кибератаки Cyber Kill Chain // Дистанционные образовательные технологии: материалы VII Международной научно-практической конференции, Ялта, 20–22 сентября 2022 года. – Симферополь, 2022. – С. 246–250. – EDN UGZVWV.

4. OWASP Top Ten | OWASP Foundation // Owasp. – URL: https://owasp.org/www-project-top-ten (accessed: 02.09.2025).

5. *Рыбаков Н.С., Сушко М.А.* Анализ уязвимостей веб-приложений // Студент: наука, профессия, жизнь: материалы XI Всероссийской студенческой научной конференции с международным участием. В 5 ч. Ч. 3, Омск, 22–26 апреля 2024 года. – Омск, 2024. – С. 228–232. – EDN HQPOJG.

6. *Мисбахов Н.И., Лукьянов Э.Р., Степанов М.О.* Обзор классификации OWASP Top 10 // Актуальные проблемы науки и образования в условиях современных вызовов: сборник материалов XXVIII Международной научно-практической конференции, Москва, 01 марта 2024 года. – СПб., 2024. – С. 51–55. – EDN TECDOF.

7. *Ли Д.* Системы BAS (Breach and Attack Simulation): комплексная симуляция кибератак на инфраструктуру // Anti-Malware.ru: сайт. – 2021, 4 мая. – URL: https://www.anti-malware.ru/analytics/Technology_Analysis/Breach-and-Attack-Simulation (дата обращения: 03.09.2025).

8. *Швидченко С.А., Решетняк А.Р.* Аналитический обзор существующих инструментов для тестирования программного обеспечения на предмет // Интеллектуальные информационные технологии и математическое моделирование: труды Международной научной конференции, пос. Дивноморское, Краснодарский край, 26–29 августа 2022 года. – Ростов н/Д., 2022. – С. 142–147. – EDN ITLMWS.

9. *Шамрицкий К.Г.* Обзор решений Breach and Attack Simulation по практической информационной безопасности // Вестник по безопасности: материалы Всероссийской научно-практической конференции по безопасности, Тольятти, 20–21 декабря 2019 года. Вып. 12. – Тольятти, 2019. – С. 48–55. – EDN EHCZRE.

Фомин Данил Андреевич, инженер по безопасной разработке программного обеспечения ЗАО «ЦФТ». Область научных интересов – разработка автоматизированных систем моделирования атак информационной безопасности. E-mail: evtdanilf@yandex.ru

Рева Иван Леонидович, доцент кафедры защиты информации Новосибирского государственного технического университета. Область научных интересов – информационная безопасность. E-mail: reva@corp.nstu.ru

DOI: 10.17212/2782-2230-2025-3-40-61

Development of a breach and attack simulation system for vulnerability detection *

D.A. Fomin[1], I.L. Reva[2]

[1] *CJSC "CFT", 35 Koltsovo Science City, Novosibirsk, 630559, Russian Federation, application security engineer. E-mail: evtdanilf@yandex.ru*
[2] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, Associate Professor of the Department of Information Protection. E-mail: reva@corp.nstu.ru*

Maintaining an adequate level of information system security remains a critical task for organizations of all sizes. Traditional approaches, such as penetration testing (pentesting), enable the identification of vulnerabilities before they can be exploited by attackers; however, these tests are often conducted irregularly and fail to cover all attack vectors. A promising solution is the adoption of Breach and Attack Simulation (BAS) platforms, which can be used both to detect vulnerabilities within an organization's IT infrastructure and to assess the effectiveness of existing security controls.

This work presents the design and implementation of an automated BAS methodology based on staged scanning of web applications. We detail each phase of the attack simulation– including reconnaissance, directory fuzzing, and template-based vulnerability scanning– employing a suite of specialized tools and scanners. Each tool contributes a distinct capability to the overall simulation process, enhancing coverage and enabling continuous, repeatable security assessments.

Keywords: information security; cyberattack simulation; vulnerabilities; cybersecurity; penetration testing; security assessment

* *Received 20 June 2025.*

REFERENCES

1. Golushko A. Aktual'nye kiberugrozy: IV kvartal 2024 goda – I kvartal 2025 goda [Current Cyber Threats: Q4 2024 – Q1 2025]. *Positive Technologies*. Website, 2025, March 20. (In Russian). Available at: https://ptsecurity.com/ru-ru/research/analytics/aktualnye-kiberugrozy-iv-kvartal-2024-goda-i-kvartal-2025-goda (accessed 02.09.2025).

2. GK «Solar»: chislo veb-atak na saity rossiiskikh kompanii za god vyroslo v 2 raza [Solar Group: Number of Web Attacks on Russian Companies' Sites Doubled over One Year]. *Solar*. Website. (In Russian). Available at: https://rt-solar.ru/events/news/5498/ (accessed 02.09.2025).

3. Boichenko O. V. [Multi-stage cyber attack model: cyber kill chain]. *Distantsionnye obrazovatel'nye tekhnologii* [Distance educational technologies]. Proceedings of the VII International Scientific and Practical Conference]. Simferopol, 2022, pp. 246–250. (In Russian).

4. *OWASP Top Ten | OWASP Foundation*. Available at: https://owasp.org/www-project-top-ten (accessed 02.09.2025).

5. Rybakov N. S., Sushko M. A. [Vulnerability analysis of web applications]. *Student: nauka, professiya, zhizn'* [Student: Science, Profession, Life]. Proceedings of the XI All-Russian student scientific conference with international participation. In 5 pt. Pt. 3. Omsk, 2024, pp. 228–232. (In Russian).

6. Misbakhov N.I., Lukyanov E.R., Stepanov M.O. [OWASP Top 10 Classification Overview]. *Aktual'nye problemy nauki i obrazovaniya v usloviyakh sovremennykh vyzovov* [Current issues of science and education in the context of modern challenges]. Collection of materials of the XXVIII International scientific and practical conference, Moscow, 2024. St. Petersburg, 2024, pp. 51–55. (In Russian).

7. Li D. Sistemy BAS (Breach and Attack Simulation): kompleksnaya simulyatsiya kiberatak na infrastrukturu [Breach and Attack Simulation Systems: Comprehensive Cyber Attack Simulation on Infrastructure]. *Anti-Malware.ru*, 2021, May 4. Available at: https://www.anti-malware.ru/analytics/Technology_Analysis/Breach-and-Attack-Simulation (accessed 03.09.2025).

8. Shvidchenko S.A., Reshetnyak A.R. [Analytical Review of Existing Software Testing Tools for Cyberthreats]. *Intellektual'nye informatsionnye tekhnologii i matematicheskoe modelirovanie* [Intelligent information technologies and mathematical modeling]. Proceedings International Scientific Conference IIT&MM-2022. Rostov-on-Don, 2022, pp. 142–147. (In Russian).

9. Shamritskiy K. G. [Review of Breach and Attack Simulation Solutions for Practical Information Security]. *Vestnik po bezopasnosti* [Security Bulletin].

Proceedings of the All-Russian Scientific and Practical Conference on Security. Tolyatti, 2019, vol. 12, pp. 48–55. (In Russian).

Для цитирования:

Фомин Д.А., *Рева И.Л.* Разработка системы моделирования кибератак для выявления уязвимостей // Безопасность цифровых технологий. – 2025. – № 3 (118). – С. 40–61. – DOI: 10.17212/2782-2230-2025-3-40-61.

For citation:

Fomin D.A, Reva I.L. Razrabotka sistemy modelirovaniya kiberatak dlya vyyavleniya uyazvimostei [Development of a breach and attack simulation system for vulnerability detection]. *Bezopasnost' tsifrovykh tekhnologii = Digital Technology Security*, 2025, no. 3 (118), pp. 40–61. DOI: 10.17212/2782-2230-2025-3-40-61.