Министерство образования Республики Беларусь
Учреждение образования
«НАЦИОНАЛЬНЫЙ ДЕТСКИЙ ТЕХНОПАРК»

ИССЛЕДОВАТЕЛЬСКИЙ ПРОЕКТ

«Программное средство для обнаружения уязвимостей веб-сайтов»

учащихся УО «Национального детского технопарка»

Држевецкого Никиты Александровича
Ромейко Михаила Юрьевича

Научный руководитель

Пулко Татьяна Александровна
к.т.н., доцент
УО «Белорусский государственный университет информатики и радиоэлектроники»

Минск 2023

СОДЕРЖАНИЕ

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Цель проекта: разработать сканер для обнаружения уязвимостей веб-приложений и провести сравнительный анализ результатов сканирования разработанного приложения с другими аналогичными инструментами.

Для достижения цели необходимо выполнить следующие задачи:

1. Провести обзор основных типов уязвимостей веб-приложений
2. Изучить инструменты обнаружения уязвимостей, включая сканеры уязвимостей и их принцип работы
3. Рассмотреть различные инструменты и утилиты для сканирования веб-приложений.
4. Изучить библиотеки языка программирования Python.
5. Спроектировать программное обеспечение сканера веб-уязвимостей «TechnoScan» с использованием нотаций диаграммы UML.
6. Разработать и продемонстрировать функционирование разработанного сканера уязвимостей.
7. Сравнить результаты сканирования, полученные с использованием разработанного инструмента, с результатами других известных и дуступных сканеров уязвимостей.

Объект исследования: уязвимости веб-приложений и способы их нахождения при помощи сканеров уязвимостей.

ВВЕДЕНИЕ

Веб-приложения для многих стали неотъемлемой частью повседневной жизни, обеспечивая доступ к различным сервисам и информации через интернет. Однако, вместе с увеличением функциональности и сложности веб-приложений, возрастает и уровень угроз для их безопасности. Представляющие серьезную опасность как для частных лиц, так и для организаций, уязвимости веб-приложений могут привести к неблагоприятным последствиям, включая утечку конфиденциальных данных, атаки на сервера, и многое другое.

Для предотвращения возможных угроз и выявления уязвимостей веб-приложений на ранних этапах, рекомендуется проводить предварительное сканирование собственного веб-сайта. Этот процесс позволяет выявить потенциальные уязвимости и принять меры по их устранению до возможного воздействия злоумышленников. Для осуществления данной операции применяются специализированные инструменты, такие как сканеры веб-уязвимостей.

В ходе исследования были тщательно рассмотрены вопросы, связанные с уязвимостями веб-приложений и средствами их обнаружения. Актуальность данной темы обосновывается не только широким распространением веб-приложений, но и возрастающим числом кибератак и инцидентов по утечке данных, которые представляют серьезную угрозу для конфиденциальности и целостности информации.

В рамках данного проекта был разработан сканер уязвимостей, обладающий графическим интерфейсом и реализованным на языке программирования Python. Была проведена демонстрация его функциональности, а также выполнен сравнительный анализ результатов сканирования с двумя другими аналогичными инструментами, такими как OWASP ZAP и W9scan.

1. Основные типы уязвимостей веб-приложений

Безопасность веб-сайтов играет важную роль в защите данных и обеспечении безопасного пользовательского опыта. Уязвимости веб-сайтов являются слабыми местами, которые могут быть использованы злоумышленниками для несанкционированного доступа, изменения данных или нарушения нормальной работы сайта. С развитием средств автоматизации проектирования и разработки программного обеспечения все меньше и меньше внимания уделяется вопросам безопасности и качества программного кода разрабатываемого программного обеспечения. В результате чего подавляющее большинство веб-сайтов содержит уязвимости различной степени критичности, определить которые поможет проведение пентеста.

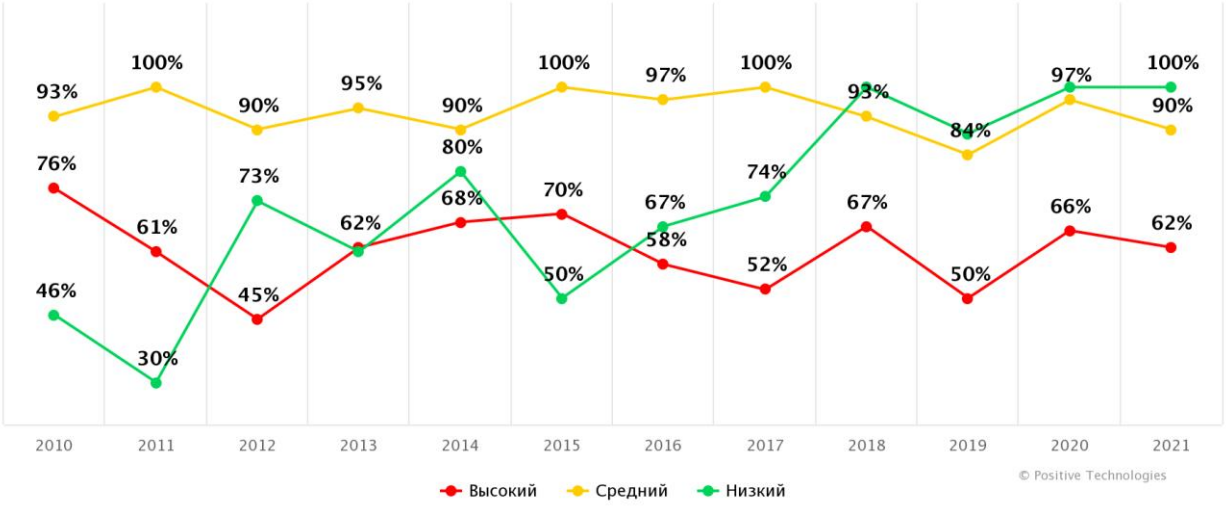

Рисунок 1. Доли веб-приложений с уязвимостями различной степени критичности [1]

В настоящее время уже существует огромное количество возможных уязвимостей и способов их использования злоумышленниками и одним из самых достоверных источников об обнаруживаемых уязвимостях веб-сайтов и связанных с ними угрозах информационной безопасности, на который ссылаются различные эксперты в области информационной безопасности веб-сайтов, является проект Open Web Application Security Project (OWASP) (рисунок 2) [2]:

1. A01 - недостатки контроля доступа: эта уязвимость связана с недостаточным или некорректным контролем доступа к ресурсам или функциональности системы; если не правильно настроить права доступа или контроль пользовательских сессий, злоумышленники могут получить доступ к чужим данным, функциям или привилегиям, нарушая безопасность системы.

2. A05 - некорректная настройка параметров безопасности: эта уязвимость возникает из-за неправильной конфигурации параметров

безопасности; например, если слабые пароли, открытые порты или доступ к неиспользуемым функциям не были должным образом настроены, это может привести к несанкционированному доступу и компрометации системы.

3. A07 - недостатки идентификации и аутентификации: уязвимости связанные с аутентификацией и идентификацией могут позволить злоумышленникам обойти или подменить механизмы аутентификации; например, слабые пароли, отсутствие механизмов двухфакторной аутентификации или небезопасные методы хранения учетных данных могут позволить злоумышленникам получить доступ к аккаунтам пользователей.

4. A04 - небезопасное проектирование: уязвимости, связанные с небезопасным проектированием, возникают, когда приложение или система разрабатываются с недостаточным учетом безопасности; это может привести к созданию уязвимых механизмов аутентификации, неправильному обращению с конфиденциальными данными и другим проблемам, которые могут быть использованы злоумышленниками.

5. A03 - внедрение: эта уязвимость возникает, когда злоумышленники могут внедрить и исполнить вредоносный код на сервере или клиенте; это может произойти из-за недостаточной обработки пользовательского ввода, небезопасного обращения с внешними данными или других слабостей в коде приложения.

6. A02 - криптографические недостатки: уязвимости, связанные с криптографией, возникают, когда используемые криптографические методы не обеспечивают должного уровня защиты; например, слабые алгоритмы шифрования, неправильное использование криптографии или недостаточное управление ключами могут привести к утечкам данных или нарушению конфиденциальности.

7. A06 - уязвимые и устаревшие компоненты: эта уязвимость возникает, когда приложение или система использует компоненты (библиотеки, фреймворки и т.д.), которые содержат известные уязвимости или устарели и не имеют актуальных обновлений безопасности; это делает систему уязвимой для атак, которые могут быть совершены через уязвимые компоненты.

8. A08 - недостатки проверки целостности ПО и данных: уязвимости связанные с проверкой целостности могут позволить злоумышленникам изменять или модифицировать данные или программное обеспечение без уведомления пользователя или системы; неправильная проверка целостности может открыть возможности для вредоносного поведения и нарушения целостности данных.

9. A09 - недостатки журналирования и мониторинга: эта уязвимость связана с недостаточной или неправильной реализацией журналирования и мониторинга в приложении или системе: отсутствие должного контроля и

мониторинга может затруднить обнаружение атак или инцидентов безопасности, что может привести к задержке в реакции и увеличению вреда от атак.

10. A10 - подделка запроса со стороны сервера (SSRF): эта уязвимость позволяет злоумышленникам заставить сервер выполнять запросы на внутренние системы или другие ресурсы, которые должны быть недоступны для публичных запросов; подобные атаки могут быть использованы для сканирования портов, обхода брандмауэров или даже выполнения внутренних операций на компрометированном сервере.

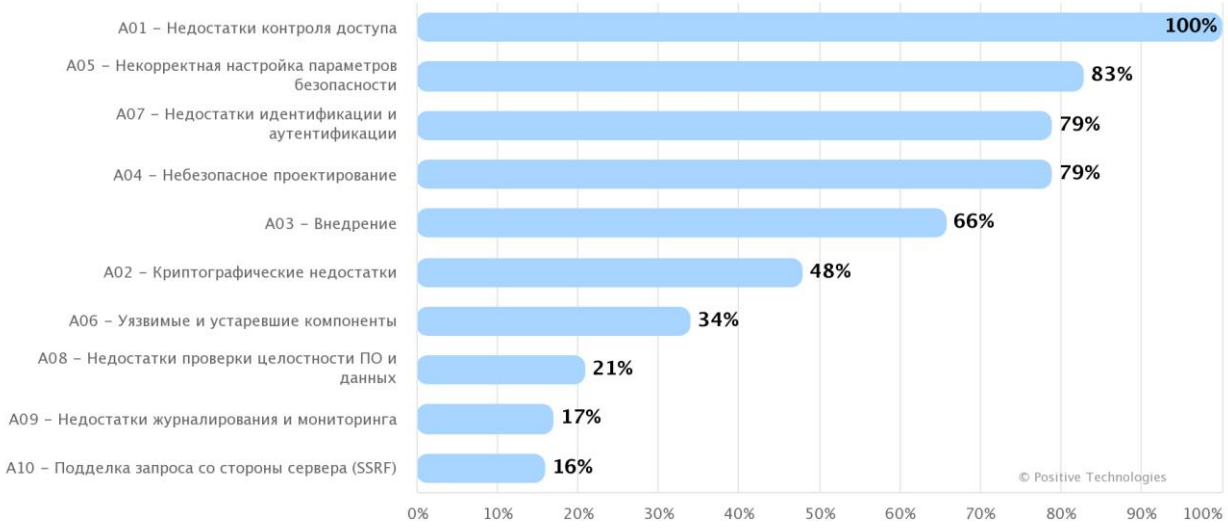

Рисунок 2. Распределение уязвимостей по категориям OWASP Top 10 – 2021 (доля приложений) [1]

Уязвимости веб-сайтов представляют серьезную угрозу для безопасности данных и функциональности сайтов. С развитием технологий и автоматизации разработки программного обеспечения, риск возникновения уязвимостей значительно возрастает. Различные виды уязвимостей, такие как инъекции, проблемы с аутентификацией и авторизацией, уязвимости сессий, XSS-атаки, уязвимости контроля доступа, небезопасные прямые ссылки на объекты и CSRF, могут быть использованы злоумышленниками для несанкционированного доступа к данным и системам пользователей. Предотвращение уязвимостей является важной задачей для организаций, которые стремятся обеспечить безопасность своих веб-приложений.

Правильная реализация мер безопасности, таких как использование проверенных методов аутентификации, обновление программного обеспечения, настройка корректных политик доступа и регулярное проведение пентестов, помогает минимизировать риски и повышает защищенность веб-сайтов. Для достижения надежной защиты от уязвимостей, организации должны сотрудничать с опытными специалистами по

информационной безопасности и применять современные инструменты для обнаружения и решения уязвимостей. Только через систематический анализ и устранение уязвимостей веб-сайтов можно обеспечить безопасность данных и сохранить доверие пользователей к функциональности и защите их персональной информации.

2. Обзор сканеров веб-уязвимостей

С каждым днем появляются новые угрозы и уязвимости, которые могут подвергнуть риску конфиденциальность, целостность и доступность данных. В этом контексте, сканеры уязвимостей играют важную роль в сетевой инфраструктуре, позволяя организациям устранять проблемы безопасности до того, как злоумышленник успеет воспользоваться ими.

Сканеры уязвимостей представляют собой инструменты, которые используются для выявления уязвимостей в сетевых системах, приложениях и устройствах. Они являются важной частью процесса обеспечения безопасности информационных систем. Такие сканеры работают путем сканирования (анализа) сетей, портов и других элементов с целью обнаружить потенциальные уязвимости, которые могут быть использованы злоумышленниками для несанкционированного доступа, атак или других видов злоупотребления [3].

Общий пример работы сканера уязвимостей включает следующие шаги (рисунок 3):

1. Пользователь определяет систему или сеть, которую необходимо проверить на наличие уязвимостей.
2. Сканер активно исследует целевую систему, собирая информацию о ее компонентах, открытых портах, сервисах и других характеристиках.
3. На основе собранной информации сканер проводит тщательное сканирование на предмет известных уязвимостей, используя базу данных уязвимостей и паттернов.
4. После завершения сканирования сканер анализирует результаты и генерирует отчет, который содержит информацию о выявленных уязвимостях и, возможно, рекомендации по их устранению.
5. Администратор или ответственное лицо принимает необходимые меры для исправления обнаруженных уязвимостей и повышения уровня безопасности системы.

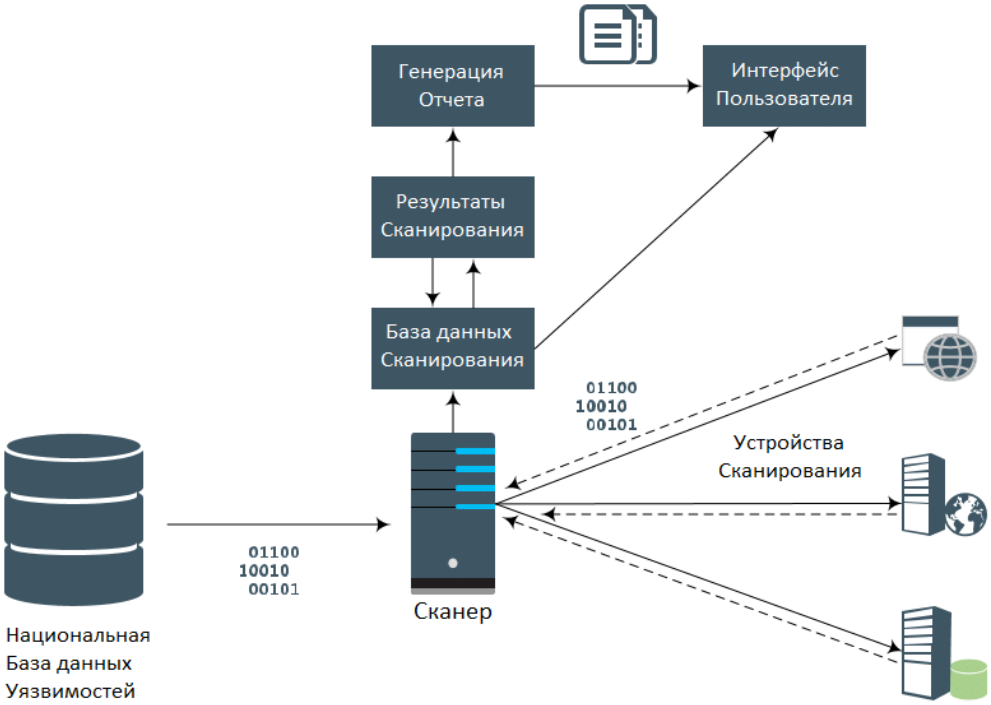

Рисунок 3. Иллюстрации принципа работы сканеров уязвимостей [5].

В принципе работы сканеров существуют два механизма поиска и анализа уязвимостей. Это зондирование и сканирование.

1. Зондирование. При таком способе анализ происходит в активной фазе, то есть инициализирует виртуальные атаки и проверяет, в каких участках системы возникают угрозы. Довольно эффективный, но медленный и рискованный метод, поскольку тестирование может вызвать реальный сбой. По окончании процесса предоставляется подробный отчет с найденными проблемами и рекомендациями по их деактивации [3, 4].

2. Сканирование. В отличии от предыдущего, этот способ работает с максимально возможной скоростью, но на поверхностном уровне, поэтому дает менее точные результаты. Однако очень маловероятно, что при таком режиме может случится сбой системы. Данный метод только предупреждает о найденных проблемах, но не более того [3, 4].

Существуют два основных подхода к сканированию уязвимостей: "белый ящик" и "черный ящик".

1. "Белый ящик" (White Box) сканирование. При этом подходе тестирующий имеет полный доступ к исходному коду приложения или системы. Это позволяет более глубоко анализировать код и обнаруживать скрытые уязвимости, но требует доступа к программному обеспечению [3].

2. "Черный ящик" (Black Box) сканирование. В этом случае тестирующий не имеет предварительной информации о структуре исследуемой системы. Такой подход моделирует действия настоящего

злоумышленника, что позволяет выявлять уязвимости с точки зрения внешнего атакующего [3].

На рынке существует множество сканеров уязвимостей, как коммерческих, так и свободно распространяемых. Пример некоторых наиболее известных зарубежных сканеров:

1. Nessus. Один из наиболее популярных и мощных сканеров с богатым функционалом и обширной базой данных уязвимостей.
2. OpenVAS. Бесплатный и открытый сканер уязвимостей, который является альтернативой Nessus.
3. Acunetix. Коммерческий сканер уязвимостей, сфокусированный на обнаружении уязвимостей в веб-приложениях и сетях.
4. Qualys. Облачный сервис для сканирования уязвимостей и анализа безопасности сетей.

Пример некоторых наиболее известных российских сканеров уязвимостей:

1. XSpider. Продукт компании Positive Technologies, который впервые появился на рынке два десятилетия назад. Если при выборе решения для компании важен отечественный продукт с именем и сложившейся репутацией на рынке, стоит рассмотреть данный инструмент [6].
2. ScanOVAL. Разработана компанией «Алтэкс-Софт» и распространяется на бесплатной основе. Скачать этот сканер уязвимостей можно с официального сайта Федеральной службы по техническому и экспортному контролю. Там же доступна и база уязвимостей для данной программы. В ней две сотни угроз и 43 тысячи уязвимостей [6].
3. RedCheck. Еще одна разработка компании «Алтэкс-Софт». Сканер уязвимости включен в реестр отечественного ПО и имеет сертификат ФСТЭК по четвертому уровню доверия [6].

У каждого из трех инструментов имеются свои преимущества. ScanOVAL – бесплатный и простой базовый вариант. XSpider — максимально гибкий инструмент со складывавшейся десятилетиями репутацией. RedCheck имеет специальные опции для интеграции с Docker и Cubernetes, что дает дополнительные плюсы при использовании этого сканера в сфере логистики [6].

CMS (Content Management System) - это программное обеспечение, позволяющее управлять созданием, редактированием и организацией контента на веб-сайте без необходимости знания программирования. Многие CMS, такие как WordPress, Wix и Joomla, являются популярными среди веб-разработчиков и владельцев сайтов.

Из-за широкой распространенности и популярности CMS, они часто становятся мишенями атак для злоумышленников. Недостаточно

обновленные или некорректно настроенные CMS могут иметь уязвимости, которые могут быть использованы для взлома сайта или внедрения вредоносного кода. Для обнаружения данных уязвимостей можно воспользоваться специальными сканерами для CMS. Вот некоторые из них:

1. WPScan. Специализируется на сканировании веб-сайтов, работающих на платформе WordPress. Этот сканер идентифицирует уязвимости в темах, плагинах и настройках WordPress, что позволяет администраторам принять соответствующие меры для усиления безопасности сайта.

2. CMSMap. Является универсальным сканером уязвимостей, который поддерживает множество различных CMS, включая WordPress, Joomla, Drupal и другие. Этот инструмент автоматически определяет CMS, используемую на веб-сайте, и сканирует его на наличие известных уязвимостей и слабых мест.

Правильное использование сканеров уязвимостей в сочетании с активными мерами обеспечения безопасности позволяет создать надежную защиту для информационных ресурсов и обеспечить безопасность в целом.

3. Исследование сетевых инструментов для поиска уязвимостей

Инструмент SQLmap

Nmap — это инструмент командной строки с открытым исходным кодом для сканирования портов, аудита сетевой безопасности, обнаружения хостов и служб и получения списка открытых портов. Он был запущен как инструмент Linux, а затем включен в Windows, macOS и BSD [7]. Nmap написан на языке программирования Python.

Утилита Nmap в процессе сканирования сети перебирает доступный диапазон портов и пытается подключиться к каждому из них. Если подключение удалось, в большинстве случаев, передав несколько пакетов, программа может даже узнать версию программного обеспечения, которая ожидает подключений к этому порту. Теперь, после того как мы рассмотрели основы, рассмотрим, как пользоваться Nmap для сканирования портов, сети и не только [7].

Запускать nmap можно как в режиме графического интерфейса, так и через командную строку [8].

Инструмент SQLmap

SQL-инъекция или SQLi – уязвимость, которая позволяет атакующему использовать фрагмент вредоносного кода на языке структурированных запросов (SQL) для манипулирования базой данных (далее СУБД) и получения доступа к потенциально ценной информации. Атаки на основе таких уязвимостей – одни из самых распространенных и опасных: они могут быть нацелены на любое веб-приложение или веб-сайт, которые взаимодействуют с базой данных SQL (а подавляющее большинство баз данных реализованы именно на SQL) [9].

Для автоматизирования поиска SQL уязвимостей существует инструмент SQLmap, написанный на языке Python. Эта утилита с открытым исходным кодом для тестирования на проникновение, которая автоматизирует процесс выявления и эксплуатации уязвимостей SQL-инъекций и захватов серверов баз данных. Данный инструмент имеет широкий набор возможностей, начиная от сбора отпечатков баз данных по полученной от них данным, до доступа к файловой системе и выполнения команд в операционной системе [10].

Есть пять основных классов SQL-инъекций, и все их поддерживает инструмент SQLmap:

1. UNION query SQL injection. Классический вариант внедрения SQL-кода, когда в уязвимый параметр передается выражение, начинающееся с "UNION ALL SELECT". Эта техника работает, когда веб-приложения напрямую возвращают результат вывода команды SELECT на страницу.

2. Error-based SQL injection. В случае этой атаки сканер заменяет или добавляет в уязвимый параметр синтаксически неправильное выражение, после чего анализирует HTTP-ответ (заголовки и тело) в поиске ошибок СУБД, в которых содержалась бы заранее известная инъецированная последовательность символов и вывод на подзапрос. Эта техника работает только тогда, когда веб-приложение по каким-то причинам (чаще всего в целях отладки) раскрывает ошибки СУБД.

3. Stacked queries SQL injection. Сканер проверяет, поддерживает ли веб-приложение последовательные запросы, и, если они выполняются, добавляет в уязвимый параметр HTTP-запроса точку с запятой и следом внедряемый SQL-запрос. Этот прием в основном используется для внедрения SQL-команд, отличных от SELECT, например, для манипуляции данными (с помощью INSERT или DELETE).

4. Boolean-based blind SQL injection. Реализация так называемой слепой инъекции: данные из баз данных уязвимого веб-приложения нигде не возвращаются. Прием также называется дедуктивным. SQLmap добавляет в

уязвимый параметр HTTP-запроса синтаксически правильно составленное выражение, содержащее подзапрос SELECT (или любую другую команду для извлечения определенных данных из базы данных) в ожидании получить отрицательный или утвердительный ответ.

5. Time-based blind SQL injection. Полностью слепая инъекция. Точно так же, как и в предыдущем случае, но сканер добавляет подзапрос, который приводит к паузе работы СУБД на определенное количество секунд. Используя эту особенность, сканер может посимвольно извлечь данные из базы данных, сравнивая время ответа на оригинальный запрос и на запрос с внедренным кодом. Здесь также используется алгоритм двоичного поиска [28].

Важно также понимать, что для разных СУБД реализации атаки зачастую сильно отличаются. Все эти случаи умеет обрабатывать SQLmap и на данный момент поддерживает MySQL, Oracle, PostgreSQL, Microsoft SQL Server, Microsoft Access, SQLite, Firebird, Sybase и SAP MaxDB [11].

Существует большое множество опций для работы с SQLmap. Для показа базовой справки можно использовать опцию “-h” или “--help”, а для продвинутой справки “-hh”, которая выведет все возможные опции инструмента SQLmap.

Инструмент XSStrike

Межсайтовый скриптинг (XSS) – это уязвимость, которая заключается во внедрении злоумышленником своего Javascript кода в веб-страницу, которая отображается в браузере пользователя.

После такого внедрения злоумышленник фактически захватывает веб-страницу и может манипулировать данными пользователя, когда он находится на странице. В случае успеха злоумышленник может:

1. Внедрять свои скрипты в веб-страницу
2. Отправлять на свой сервер пользовательские данные - банковские карты, идентификаторы сессий, пароли и тд.
3. Совершать действия от имени пользователя - рассылать спам, совершать денежные переводы [12].

Существует несколько видов XSS:

1. Хранимые (Stored) – вредоносный код сохраняется на сервере и загружается с него каждый раз, когда пользователи запрашивают отображение той или иной страницы.
2. Отображаемые (Reflected) – вредоносная строка является частью запроса жертвы к веб-сайту. Сайт принимает и вставляет эту вредоносную строку в отправляемый ответ обратно пользователю.

3. XSS в DOM-модели (DOM-Based) – представляет собой вариант как хранимой, так и отображаемой XSS-атаки. В этой XSS-атаке вредоносная строка не обрабатывается браузером жертвы, пока настоящий JavaScript веб-сайта не выполнится [13].

Уязвимость возникает из-за недостаточной фильтрации данных, которые выводятся при отображении страницы. Такие уязвимости довольно часто встречаются даже в крупных продуктах, поэтому стоит обязательно тестировать свои веб-приложения на наличие XSS уязвимостей [12].

Возможности инструмента XSStrike:

– мощный двигатель фаззинга;

– технология взлома контекста;

– Интеллектуальная генерация пэйлоадов;

– Поддержка метода GET & POST;

– Поддержка файлов cookie;

– Обнаружение WAF;

– Пэйлоады ручной работы для фильтрации и WAF-уклонения;

– Скрытое обнаружение параметров [14].

Важно отметить, что данный инструмент позволяет работать как с GET-запросами, так и с POST-запросами.

4. Разработка сканера веб-уязвимостей «TechnoScan» на языке программирования Python

Этапы разработки сканера веб-уязвимостей «TechnoScan»

Главной задачей при создании приложения было составление простого и интуитивно понятного интерфейса, объединение трех различных инструментов, а также полная автоматизация сканирования. Тем самым использование сканера «TechnoScan» будет удобно даже для тех, кто не является экспертом в области информационной безопасности. Также благодаря использованию Nmap, SQLmap и XSStrike программа имеет обширный охват уязвимостей различных типов.

Для разработки структурированной архитектуры кода данной программы и для обеспечения более четкой спецификации задачи, была создана UML-диаграмма работы данного приложения [15].

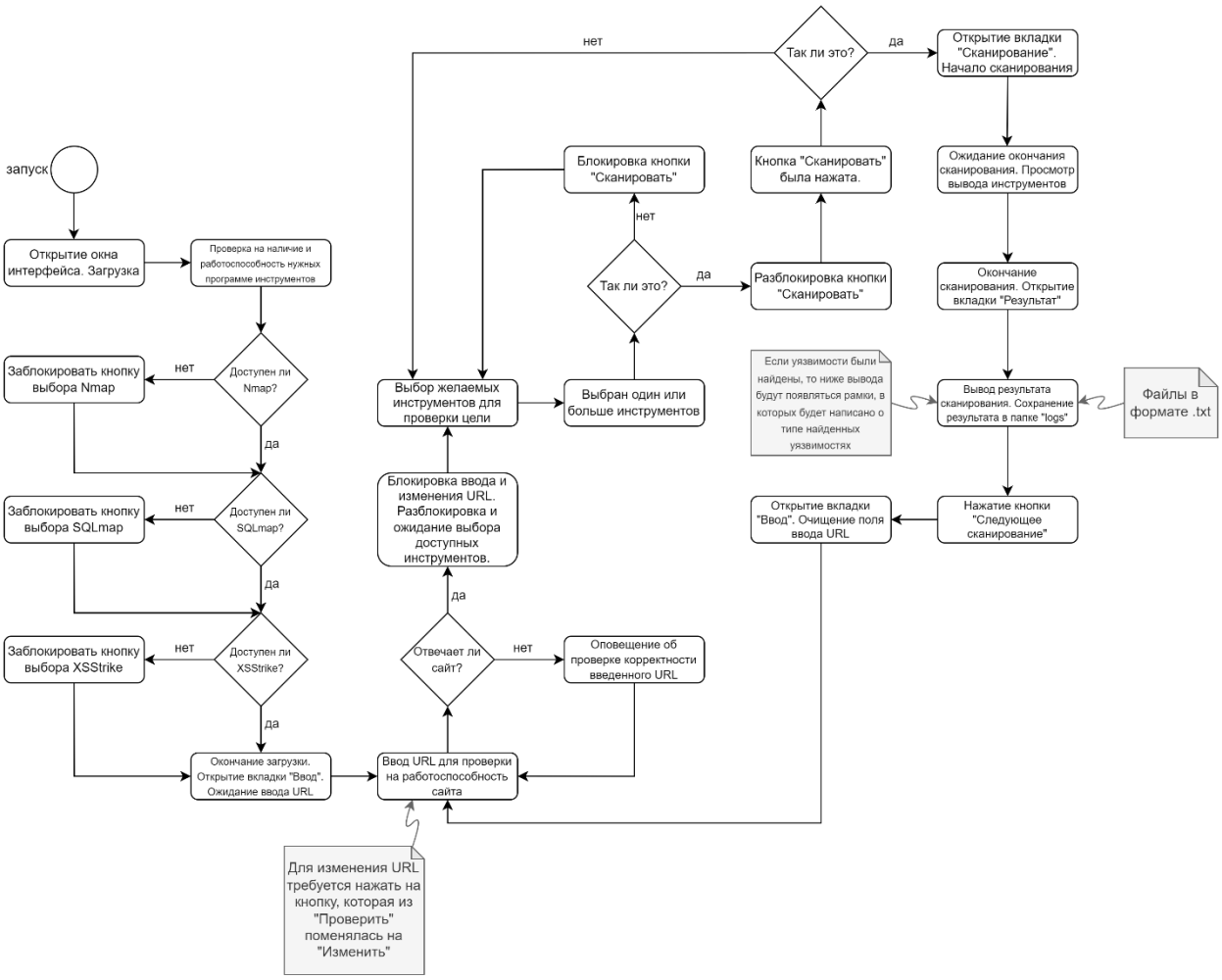

Рисунок 4. UML-диаграмма работы сканера веб-уязвимостей «TechnoScan» [15]

Как видно из диаграммы, для достижения поставленной цели, было решено объединить сразу несколько инструментов (Nmap, SQLmap и XSStrike) для проведения сканирования, чтобы приложение имело обширный охват уязвимостей различных типов. Для программирования использовался язык Python, возможности которого охватывают различные области и позволяют занимать лидирующие позиции в мире программирования [16]. Одни из самых важных отрывков кода по функциональности предоставлен на рисунке 5.

```python
import requests
import subprocess as sp
import customtkinter as ctk
from tkinter import *
from PIL import Image, ImageTk, ImageEnhance
import os
from datetime import datetime
import threading as th
import re
from time import sleep
```

Рисунок 5. Использование библиотек Python в коде программы сканера уязвимости «TechnoScan»

Демонстрация работы сканера веб-уязвимостей «TechnoScan»

Для работы и использования сканера требуется запустить «TechnoScan.exe» из его исходной директории, после чего откроется окно с меню загрузки на нем. Для проверки сайта на уязвимости требуется ввести его адрес, нажать кнопку «Подтвердить», выбрать инструменты для сканирования и нажать кнопку «Сканировать» (рисунок 6).

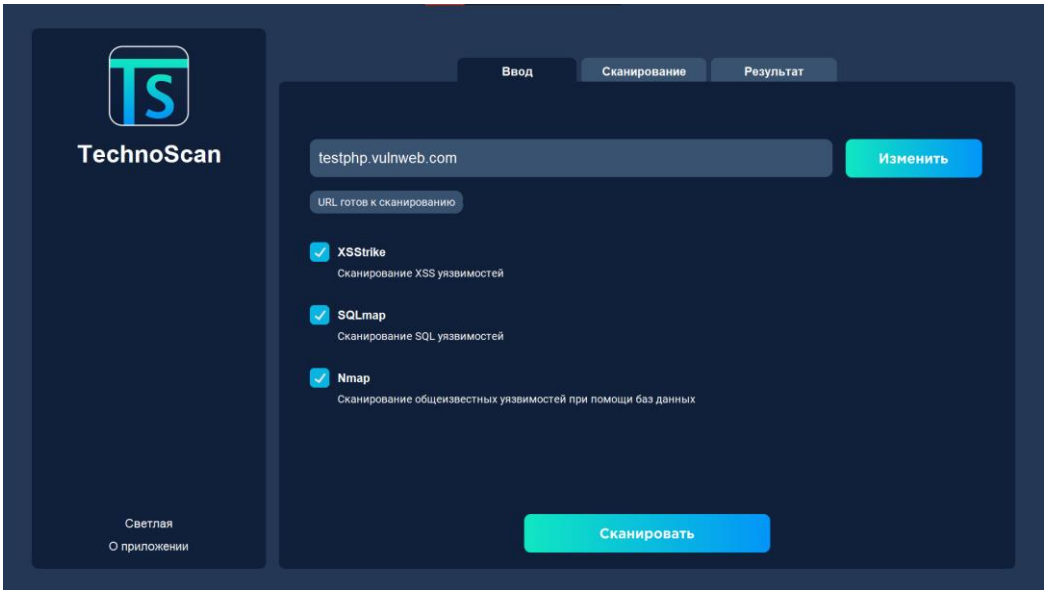

Рисунок 6. Выбор цели и инструментов сканера «TechnoScan»

Затем откроется вкладка «Сканирование». Требуется ожидать конца сканирования (рисунок 7).

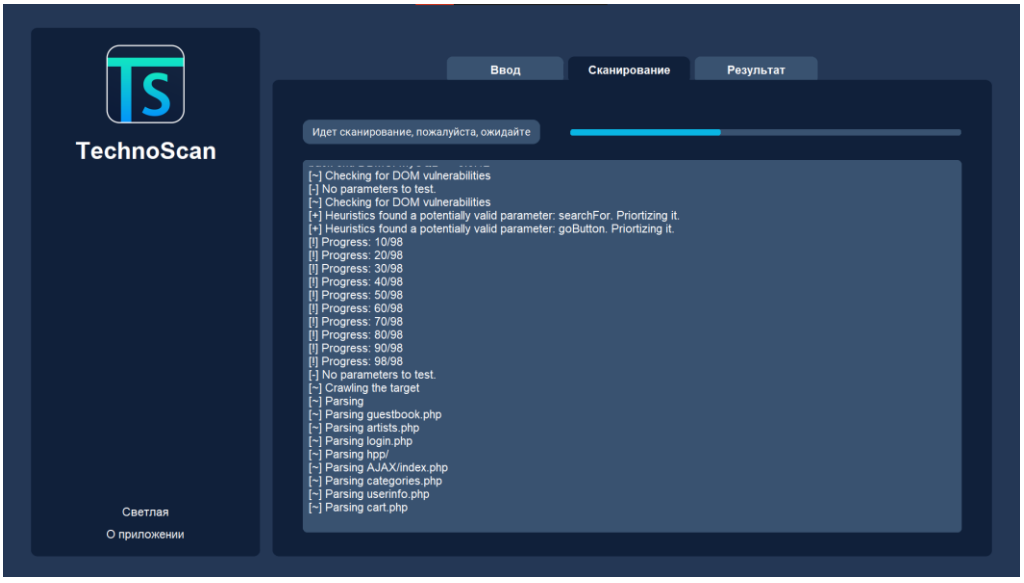

Рисунок 7. Процесс работы сканера «TechnoScan»

Затем откроется вкладка «Результат» (рисунок 8). В ней будут показаны все обнаруженные уязвимости, а также дополнительная информация.

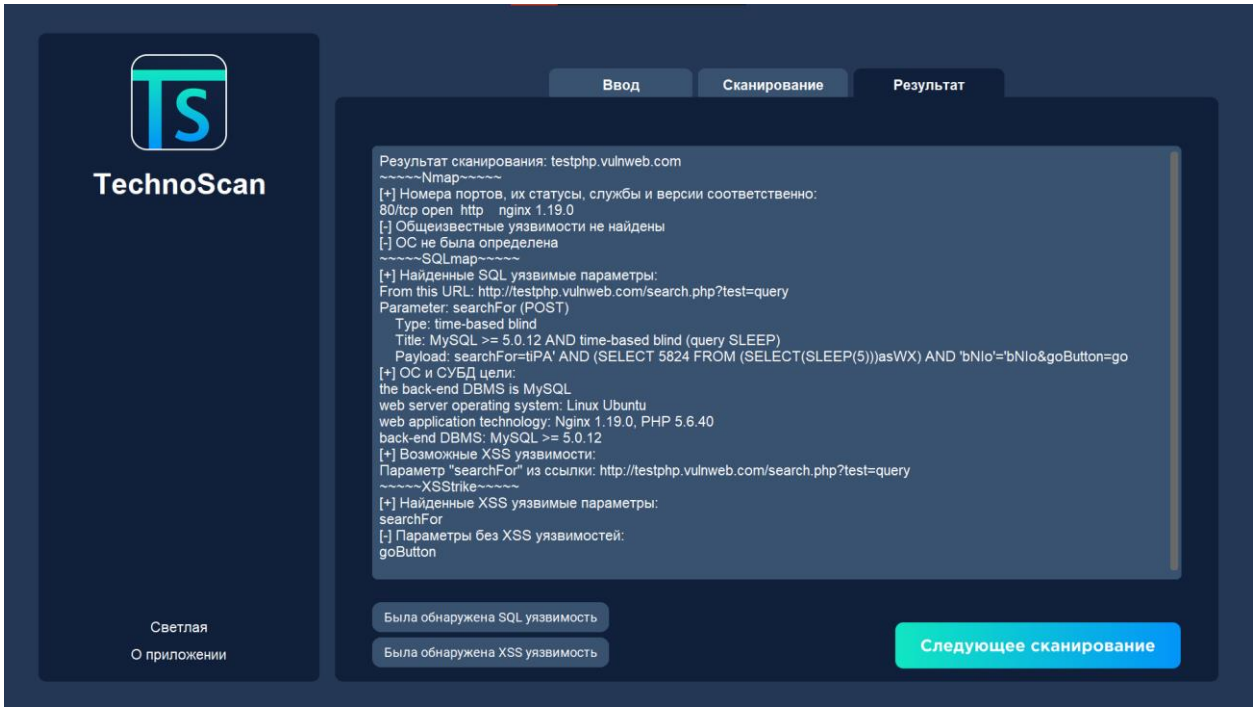

Рисунок 8. Результат сканирования сканера «TechnoScan»

Весь результат будет сохранен в log-файл формата .txt в папке «logs». Для повторного выбора цели и инструментов требуется нажать кнопку «Следующее сканирование».

Сравнительный анализ сканирования сканера веб-уязвимостей «TechnoScan»

Для оценки работоспособности и эффективности сканера «TechnoScan» было выполнено сравнение его результатов сканирования трех открытых для тестирования веб-сайтов: testphp.vulnweb.com, rest.vulnweb.com и altoromutual.com, с результатами сканирования двух других сканеров уязвимостей, таких как OWASP ZAP и W9scan.

Первым веб-сайтом для сканирования являлся testphp.vulnweb.com. При сканировании данной цели сканером уязвимостей «TechnoScan» были обнаружены, как было показано выше, XSS и SQL уязвимости. При сканировании сканером OWASP ZAP также были обнаружены XSS и SQL уязвимости. Сканер W9scan нашел 3 уязвимости и 12 угроз. Все 12 угроз были о возможном наличии XSS уязвимостей, а все 3 уязвимости о наличии SQLi.

Второй целью для сканирования являлся веб-сайт rest.vulnweb.com.

Благодаря инструменту Nmap, сканер “TechnoScan” нашел множество уязвимостей из списка CVE, CNVD и т.п. При сканировании сканером OWASP ZAP никаких серьезных уязвимостей не было найдено. Сканер W9scan также никаких уязвимостей и угроз не выявил.

Третьим и последним веб-сайтом для сканирования являлся altoromutual.com. Сканер TechnoScan нашел 2 уязвимости из CVE и 2 XSS уязвимости. Сканирование при использовании сканера OWASP ZAP показывает, что были найдены XSS и SQL уязвимости. В результате сканирования инструмента W9scan было выявлено 2 угрозы в виде XSS уязвимостей.

Все результаты сравнительного анализа эффективности функционирования разработанного сканера уязвимостей представлены в таблице 1.

Таблица 1 – Сравнительный анализ сканирования веб-уязвимостей приложением “TechnoScan”

Используемые сканеры уязвимостей	Цели и найденные в них уязвимости		
	testphp.vulnweb.com	rest.vulnweb.com	altoromutual.com
«TechnoScan»	SQLi, XSS	уязвимости из CVE, CNVB и т.п.	XSS, уязвимости из CVE
OWASP ZAP	SQLi, XSS	не было найдено	SQLi, XSS
W9scan	SQLi, XSS	не было найдено	XSS

После проведенного сравнительного анализа результатов сканирования сканера «TechnoScan» с результатами известных сканеров OWASP ZAP и W9scan, следует отметить, что «TechnoScan» продемонстрировал конкурентоспособное и сравнительно высокое качество сканирования. Таким образом, его можно считать полноценным и эффективным инструментом для нахождения веб-уязвимостей своей организации, включая специалистов с базовым уровнем знаний в данной области.

ЗАКЛЮЧЕНИЕ

В данной работе были рассмотрены уязвимости и угрозы веб-приложений и сетевой безопасности, а также принципы работы сканеров веб-уязвимостей. Были проанализированы инструменты Nmap, SQLmap и XSStrike и их возможности для обнаружения уязвимостей веб-приложений. Кроме того, были изучены основные концепции языка программирования Python и UML-диаграммы.

В рамках работы был разработан сканер веб-уязвимостей "TechnoScan" на языке программирования Python и продемонстрирована его работа. Основная цель работы заключалась в разработке сканера уязвимостей веб-сайтов и исследованию эффективности его работы. В результате был проведен сравнительный анализ результативности поиска уязвимостей разработанным сканером с другими популярными аналогами.

Исследование уязвимостей и угроз сетевой безопасности является актуальной темой в настоящее время, когда все больше информации хранится и передается через интернет. Использование сканеров веб-уязвимостей, таких как TechnoScan, позволяет обнаруживать уязвимости веб-приложений и принимать меры по их устранению.

В заключение можно отметить, что проведенная работа позволила углубить знания о уязвимостях и угрозах сетевой безопасности, а также о принципах работы сканеров веб-уязвимостей. Эти знания и инструменты помогут в создании более безопасной сетевой среды и защите от кибератак в настоящее время, когда цифровые угрозы становятся все более распространенными и серьезными.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

[1] Уязвимости и угрозы веб-приложений в 2020–2021 гг. [Электронный ресурс] – Режим доступа: https://www.ptsecurity.com/ru-ru/research/analytics/web-vulnerabilities-2020-2021/ - 19.07.2023

[2] Уязвимости веб-приложений Александр Обухов [Электронный ресурс]. – Режим доступа: https://trust-space.ru/blog/uyazvimost-veb-prilozhenij/ -19.07.2023

[3] Сканер уязвимостей – что это и зачем он нужен | Макхост [Электронный ресурс]. – Режим доступа: https://mchost.ru/articles/chto-takoe-skaner-uyazvimostej/ – 25.07.2023

[4] Сканер уязвимостей (Vulnerability scanner): что это, функции, принципы работы [Электронный ресурс]. – Режим доступа: https://itglobal.com/ru-ru/company/glossary/vulnerability-scanner/ – 25.07.2023

[5] Vulnerability Scanner System Diagram [Электронный ресурс]. – Режим доступа: https://www.researchgate.net/figure/Vulnerability-Scanner-System-Diagram_fig1_329958789/ – 25.07.2023

[6] Российские сканеры уязвимостей: сравниваем и выбираем [Электронный ресурс]. – Режим доступа:

[7] Как использовать Nmap для сканирования портов [Электронный ресурс]. – Режим доступа: https://setiwik.ru/kak-ispolzovat-nmap-dlya-skanirovaniya-portov/ – 04.08.2023

[8] NMap : Rebrain | Блог [Электронный ресурс]. – Режим доступа: https://rebrainme.com/blog/linux/nmap/ – 04.08.2023

[9] Как за 30 минут бесплатно проверить свой сайт на наличие уязвимостей [Электронный ресурс]. – Режим доступа: https://vc.ru/dev/158495-kak-za-30-minut-besplatno-proverit-svoy-sayt-na-nalichie-uyazvimostey – 04.08.2023

[10] Что такое SQL-инъекция? Определение и описание [Электронный ресурс]. – Режим доступа: https://www.kaspersky.ru/resource-center/definitions/sql-injection – 08.08.2023

[11] Sqlmap - Инструменты Kali Linux [Электронный ресурс]. – Режим доступа: https://kali.tools/?p=816 – 08.08.2023

[12] Sqlmap: SQL-инъекции — это просто [Электронный ресурс]. – Режим доступа: https://xakep.ru/2011/12/06/57950/ – 08.08.2023

[13] Как за 30 минут бесплатно проверить свой сайт на наличие уязвимостей [Электронный ресурс]. – Режим доступа: https://vc.ru/dev/158495-kak-za-30-minut-besplatno-proverit-svoy-sayt-na-nalichie-uyazvimostey – 09.08.2023

[14] Эксплуатация XSS уязвимостей с использованием XSStrike [Электронный ресурс]. – Режим доступа: https://defcon.ru/web-security/12387/ – 09.08.2023

[15] Држевецкий, Н. А. Проектирование программного обеспечения сканера веб-уязвимостей TechnoScan с использованием нотаций диаграммы UML / Н. А. Држевецкий, М. Ю. Ромейко, Т. А. Пулко. – Текст : непосредственный // Юный ученый. – 2023. – № 9 (72). – URL: https://moluch.ru/young/archive/72/3920/ (дата обращения: 04.10.2023).

[16] Пулко Т.А., Држевецкий Н. А., Ромейко М. Ю. Программа "TechnoScan". Свидетельство о добровольной регистрации и депонировании объекта авторского права № 1634-КП от 26 сентября 2023 г. выданное Национальным центром интеллектуальной собственности Республики Беларусь.