|  |  |  |
| --- | --- | --- |
|  |  | УТВЕРЖДАЮ |
|  |  | Руководитель образовательной программы  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сергеева Е. Г.  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.  М.П. |

**РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ**

«Система автоматизированного тестирования безопасности веб-приложениц»

|  |  |  |
| --- | --- | --- |
| СОГЛАСОВАНО |  | СОГЛАСОВАНО |
| Колледж ВятГУ  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Вершинина Н.Д.  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.  М.П. |  | Руководитель УП  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Долженкова М. Л.  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

2025

Содержание

[1 Введение 1](#_Toc212572179)

[1.1 Область применения 1](#_Toc212572180)

[1.2 Краткое описание возможностей 1](#_Toc212572181)

[1.3 Уровень подготовки пользователя 1](#_Toc212572182)

[1.4 Перечень эксплуатационной документации, с которыми необходимо ознакомиться пользователю 2](#_Toc212572183)

[2 Назначения и условия применения 3](#_Toc212572184)

[2.1 Виды деятельности, функции, для автоматизации которых предназначено данное средство автоматизации 3](#_Toc212572185)

[2.2 Условия, при соблюдении (выполнении, наступлении) которых обеспечивается применение средства автоматизации в соответствии с назначением 3](#_Toc212572186)

[3 Подготовка к работе 4](#_Toc212572187)

[3.1 Состав и содержание дистрибутивного носителя данных 4](#_Toc212572188)

[3.2 Порядок загрузки данных и программ 4](#_Toc212572189)

[3.3 Порядок проверки работоспособности 4](#_Toc212572190)

[4 Описание операций 5](#_Toc212572191)

[4.1 Описание всех выполняемых функций задач, комплексов задач, процедур 5](#_Toc212572192)

[4.2 Пользовательский интерфейс 5](#_Toc212572193)

[5 Аварийные ситуации 7](#_Toc212572194)

[5.1 Действия в случае несоблюдения условий выполнения технологического процесса, в том числе при длительных отказах технических средств 7](#_Toc212572195)

[5.2 Действия по восстановлению программ и/или данных при отказе магнитных носителей или обнаружении ошибок в данных 7](#_Toc212572196)

[5.3 Действия в случаях обнаружении несанкционированного вмешательства в данные 7](#_Toc212572197)

[6 Рекомендации к освоению 9](#_Toc212572198)

# 

# 1 Введение

Данный документ является руководством пользователя системы автоматизированного тестирования для проведения эмуляций атак, проверки уязвимостей и просмотр отчетов.

## Область применения

Система предназначена для специалистов в области информационной безопасности, разработчиков и DevOps-инженеров, заинтересованных в оперативном выявлении, анализе, и подтверждении уязвимостей в веб-ресурсах и API-сервисах.

## Краткое описание возможностей

* автоматизированный сбор и анализ структуры веб-приложений;
* моделирование и выполнение атак различных типов (XSS, SQL Injection, CSRF, Command Injections, и др.);
* верификация найденных уязвимостей с помощью Proof-of-Work;
* формирование отчетов и аналитических данных для последующего устранения уязвимостей;
* интеграция с системами CI/CD для автоматического запуска тестирования при обновлении кода.

## 1.3 Уровень подготовки пользователя

Уровень подготовки пользователя для работы с системой автоматизированного тестирования безопасности веб-приложений – средний.

Пользователь должен обладать базовыми знаниями в области информационных технологий и понимать принципы функционирования веб-приложений. Для эффективного использования системы рекомендуется:

* знание основных терминов и принципов информационной безопасности;
* базовые навыки работы с веб-интерфейсом, API и файловыми структурами;
* понимание принципов клиент-серверного взаимодействия;
* умение работать с браузером, консолью и файловой системой.

## Перечень эксплуатационной документации, с которыми необходимо ознакомиться пользователю

Данная система имеет перечень следующих документов:

* аналитическая записка;
* техническое задания;
* технический проект;
* руководство пользователя (данный документ);
* руководство программиста;
* проверка методики испытаний.

# Назначения и условия применения

## Виды деятельности, функции, для автоматизации которых предназначено данное средство автоматизации

Данная система предназначена для автоматизации процессов поиска, анализа и подтверждения уязвимостей в веб-приложениях и API-сервисах.

Система позволяет:

* выполнять автоматизированное сканирование веб-приложений с использованием DAST- и SAST-подходов;
* моделировать и эмулировать различные типы атак;
* формировать отчет об уязвимостях с указанием уровня критичности и предложениями по устранению;
* сохранять результаты сканирования в базе данных и экспортировать их в PDF/JSON;
* интегрироваться с CI/CD для автоматического запуска проверок при обновлении кода;
* отслеживать статистику и динамику безопасности через веб-интерфейс и систему миониторинга.

## Условия, при соблюдении (выполнении, наступлении) которых обеспечивается применение средства автоматизации в соответствии с назначением

Для стабильной работы персональный компьютер должен соответствовать следующим минимальным системным требованиям:

* операционная система – Windows 10 / Linux / macOS;
* монитор;
* компьютерная мышь;
* установленный Docker Desktop или Docker Engine версии не ниже 20.10;
* оперативная память не менее 8 ГБ.

Для стабильной работы веб-интерфейса, клиент должен соответствовать следующим минимальным требованиям:

* современный браузер;
* стабильное интернет-соединение.

Дополнительные условия эксплуатации:

* доступ к внешним сетям для загрузки зависимостей и шаблонов атак.

# Подготовка к работе

## Состав и содержание дистрибутивного носителя данных

Система автоматизированного тестирования безопасности веб-приложений распространяется в виде набора исходных файлов и конфигураций, размещённых в репозитории проекта.

Программное обеспечение не требует установки на пользовательский компьютер и разворачивается в контейнеризированной среде с использованием Docker Compose.

Состав дистрибутива:

* исходный код системы (каталоги services/, configs/, infra/, tests/, webui/);
* файлы конфигурации окружения (configs/env.example, docker-compose.yml, Makefile);
* документация (docs/ — техническое задание, технический проект, методика испытаний, руководство пользователя);
* вспомогательные скрипты (infra/scripts/ — для генерации шаблонов, инициализации БД и загрузки тестовых данных).

## Порядок загрузки данных и программ

1. скачайте или клонируйте проект из репозитория;
2. перейдите в корневую директорию проекта;
3. скопируйте пример конфигурации окружения;
4. откройте файл configs/.env и укажите свои параметры (данные подключения к PostgreSQL и Redis, ключи доступа MinIO, секретный ключ JWT);
5. выполните сборку контейнеров;
6. запустите систему.

## Порядок проверки работоспособности

* 1. после запуска контейнеров убедиться, что все сервисы активны;
  2. открыть адрес в браузере <http://localhost:8000/docs> - это страница Swagger-документации REST API;
  3. на странице API выполнить тестовый запрос, указав целевой URL для проверки. В ответ система должна вернуть Task ID и статус задачи «в очереди»;
  4. после завершения задачи должен быть доступен отчет с результатами анализа;
  5. для проверки веб-интерфейса нужно открыть панель управления и убедиться, что в списке задач отображаются результаты сканирования.

# Описание операций

## Описание всех выполняемых функций задач, комплексов задач, процедур

Система предоставляет следующие основные функции:

* запуск сканирования веб-приложения позволяет пользователю задать URL-адрес, область анализа, уровень глубины и тип проверок. Система выполняет краулинг и анализ структуры сайта;
* анализ уязвимостей;
* после обнаружения потенциальных уязвимостей система выполняет их верификацию с помощью безопасных РоС-запросов, сохраняя результаты (логи, ответы сервера, скриншоты);
* система формирует детализированные отчеты в форматах PDF и JSON, которые содержат описание найденных уязвимостей, уровень критичности, доказательство, рекомендации по устранению;
* пользователь может просматривать активные и завершенные проверки, фильтровать результаты по дате, уровню угрозы и статусу, повторно запускать задачи;
* система поддерживает автоматический запуск сканирования через webhook при каждом коммите или деплое в репозиторий;
* все результаты сохраняются в базе данных и файловом хранилище;
* все действия пользователей и системные события фиксируются в логах, а состояние сервисов и воркеров контролируется через Grafan и Prometheus.
  1. **Пользовательский интерфейс**

Пользовательский интерфейс системы реализован в виде веб-панели и включает несколько разделов:

1. Экран авторизации

Предназначен для выхода в систему с использованием учетных данных. После успешной аутентификации пользователю выдается токен доступа, который используется для последующих операций.

1. Главная панель

Отображает общую статистику: количество выполненных и активных сканирований, количество найденных уязвимостей по уровням критичности, последние события системы.

В верхней части расположено меню навигации по разделам.

1. Раздел «Новая проверка»

Предназначен для создания и запуска задач на сканирование веб-приложений. В форме указываются: URL цели, типы проверки, глубина и параметры краулинга, дополнительные опции (авторизация, токены).

После запуска отображается уникальный идентификатор задачи и ее статус.

1. Раздел «Мониторинг задач»

Позволяет просматривать список всех активных и завершенных задач.

Для каждой задачи отображаются: текущий статус, процент выполнения, дата запуска и завершения.

При нажатии на задачу открывается подробная информация с логами и ссылками на отчеты.

1. Раздел «Отчеты»

Содержит список всех завершенных сканирований.

Пользователь может открыть отчет в интерфейсе или скачать его, посмотреть доказательства и скриншоты атак, ознакомиться с рекомендациями по устранению уязвимостей.

1. Раздел «Настройки»

Данный раздел предназначен для управления личными данными, API-ключами и токенами доступа, параметрами подключения к внешним сервисам, настройками CI/CD.

1. Раздел «Мониторинг системы»

Данный раздел отображает текущее состояние всех компонентом: активность ворекров Celery, очередь заданий Redis, использование ресурсов (CPU, RAM), статус соединения БД и MinIO.

Информация визуализируется через интеграцию с Grafana и Prometheus.

# Аварийные ситуации

## Действия в случае несоблюдения условий выполнения технологического процесса, в том числе при длительных отказах технических средств

При сбоях в работе системы или нарушении условий эксплуатации необходимо выполнить следующие действия:

* проверить корректность работы всех сервисов, если один из контейнеров остановлен, то перезапустить его;
* убедиться, что соединение с БД и брокером очередей активно;
* проверить доступность веб-интерфейса и API;
* проверить логи системы.

В случае длительного отказа оборудования выполнить резервный запуск на резервном сервере или виртуальной машине.

## Действия по восстановлению программ и/или данных при отказе магнитных носителей или обнаружении ошибок в данных

В случае потери или подтверждения данных необходимо выполнить:

* восстановить БД из последней резервной копии;
* проверить целостность файлов артефактов в хранилище и восстановить их из резервной копии;
* пересобрать контейнеры приложения при повреждении зависимостей.

## Действия в случаях обнаружении несанкционированного вмешательства в данные

При подозрении на несанкционированное вмешательство или компрометацию системы необходимо:

* немедленно отключить сетевое подключение к серверу для предотвращения дальнейших действий злоумышленников;
* проверить логи доступа на предмет подозрительных запросов;
* изменить все учётные данные: пароли к БД, API-токены, ключи доступа к MinIO и Redis;
* провести проверку целостности исходного кода и зависимостей с помощью контрольных сумм;
* после анализа и устранения последствий восстановить систему из резервной копии и провести внеочередное обновление всех компонентов.
  1. **Действия в других аварийных ситуациях**

В случае возникновения других аварийных ситуаций при работе с программой необходимо выполнить следующее:

* определить характер и причины инцидента;
* принять меры для предотвращения распространения инцидента и минимизации последствий;
* зафиксировать событие в журнале происшествий;
* уведомить администратора системы и ответственных сотрудников службы информационной безопасности;
* при необходимости обратиться к технической документации или в службу поддержки для восстановления нормальной работы.

# Рекомендации к освоению

* ознакомьтесь с основными разделами системы через веб-интерфейс: «Новая проверка», «Мониторинг задач», «Отчёты» и «Настройки». Это поможет понять последовательность работы и доступные функции;
* для первого запуска используйте тестовый URL, чтобы ознакомиться с процессом сканирования и формированием отчёта;
* изучите структуру отчёта: уровень критичности уязвимостей, описание, доказательства (PoC) и рекомендации по устранению;
* ознакомьтесь с документацией API, для интеграции системы с внешними сервисами или CI/CD-пайплайнами;
* настройте параметры окружения в файле .env (подключения к БД, Redis, MinIO, CI/CD-токены) для корректной работы всех сервисов;
* регулярно проверяйте обновления зависимостей и шаблонов атак (Nuclei, Semgrep) для обеспечения актуальности проверок;
* используйте систему мониторинга (Grafana, Prometheus) для отслеживания состояния сервисов и своевременного реагирования на сбои;
* перед эксплуатацией в рабочей среде рекомендуется провести тестовую установку и пробное сканирование, чтобы убедиться в корректности конфигурации.