Федеральное государственное автономное образовательное учреждение высшего образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий  
Кафедра «Информационная безопасность»

Направление подготовки/ специальность: Безопасность компьютерных систем

ОТЧЕТ

по проектной практике

Студент: Ильин Кирилл Александрович Группа: 241-353

Место прохождения практики: Московский Политех, кафедра «Информационная безопасность»

Отчет принят с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дата \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель практики: Кесель Сергей Александрович

Москва 2025

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Общая информация о проекте

Название проекта

Цели и задачи проекта

Описание задания по проектной практике

Описание достигнутых результатов по проектной практике

ЗАКЛЮЧЕНИЕ

ПРИЛОЖЕНИЯ

**ЗАДАНИЕ ПО ПРОЕКТНОЙ ПРАКТИКЕ**

Задание на проектную практику разделялось на базовую и вариативную части. Трудоёмкость практики составляла 72 академических часа. Задание выполнялось в составе группы из 2 человек (Ильин К.А(241-353), Кондраков В.И(241-371)).

Для управления версиями использовался Git, для написания документации — Markdown, а для создания статического веб-сайта — языки разметки HTML и CSS. В качестве платформы для размещения репозиториев использовался [GitHub](https://github.com/).

Задание состоит из двух частей. Первая часть является общей и обязательной для всех студентов. Вторая часть вариативная. Задание на вторую (вариативную) часть было получено от ответственного за проектную практику на выпускающей кафедре.

**1. Базовая часть задания**

1. **Настройка Git и репозитория**:

* Создать групповой репозиторий на [GitHub](https://github.com/)  на основе предоставленного [шаблона](https://github.com/mospol/practice-2025-1).
* Освоить базовые команды Git: клонирование, коммит, пуш и создание веток.
* Регулярно фиксировать изменения с осмысленными сообщениями к коммитам.
* **Примерное время:** 5 часов.

1. **Написание документов в Markdown:**

* Все материалы проекта (описание, журнал прогресса и др.) оформить в формате Markdown.
* Изучить синтаксис Markdown и подготовить необходимые документы.
* **Примерное время:** 5 часов.

1. **Создание статического веб-сайта**:

* Для создания сайта необходимо использовать только HTML и CSS.
* Создать новый сайт об основном проекте по дисциплине «Проектная деятельность» (INVADE). Оформление и наполнение сайта должны быть уникальны.
* Сайт должен включать:
* Домашнюю страницу с аннотацией проекта.
* Страницу «О проекте» с описанием проекта.
* Страницу «Участники» с описанием личного вклада каждого участника группы в проект по «Проектной деятельности».
* Страницу «Журнал» с минимум тремя постами (новостями, блоками) о прогрессе работы.
* Страницу «Ресурсы» со ссылками на полезные материалы.
* Оформить страницы сайта графическими материалами (фотографиями, схемами, диаграммами, иллюстрациями)
* **Примерное время:** изучение и настройка — 14 часов, дизайн и наполнение — 8 часов.

**2. Вариативная часть задания:**

В качестве вариативной части нашей группе было дано следующее задание:

**Тема задания:**

"Настроить систему логирования и базового анализа событий безопасности веб-сервера."

**Задачи задания:**

* Установить и настроить веб-сервер (Apache/Nginx).
* - Включить и настроить ведение логов доступа и ошибок.
* - Обработать логи: фильтрация атак, brute-force, SQL-инъекции, DoS-атаки (и другие на усмотрение студентов).
* - Оформить шаблон отчета ИБ-инцидента (карточки инцидента) на основе логов.

*Примечание: рекомендуется проводить все манипуляции в изолированном сегменте*.

* **Примерное время:** 32-40 часов

**ДОСТИГНУТЫЕ РЕЗУЛЬТАТЫ ПО ПРОЕКТНОЙ ПРАКТИКЕ**

* Изучен язык разметки HTML для формирования базовой структуры веб-сайта (Затраченное время: 22 часа)
* Настроен групповой репозиторий на платформе GitHub, выполнено добавление и отслеживание изменений проекта с использованием системы контроля версий Git;( Затраченное время: 5 часов)
* Установлен и сконфигурирован веб-сервер Apache в изолированной среде (локальный сегмент), обеспечена доступность по адресу 127.0.0.1; (Затраченное время: 8 часов)
* Активированы и проверены механизмы логирования событий на веб-сервере, включая ведение файлов access.log и error.log; (Затраченное время: 3 часа)
* Проведены и зафиксированы три практических теста по моделированию атак:
  + **Brute-force атака** на форму входа с использованием инструмента Hydra, приведшая к успешному подбору пароля;
  + **SQL-инъекция** с применением утилиты sqlmap, позволившая извлечь данные из уязвимой базы SQLite;
  + **DoS-атака типа Slowloris**, направленная на исчерпание доступных соединений веб-сервера, результатом которой стало достижение лимита MaxRequestWorkers и нарушение доступности;
  + (Затраченное время: 14 часов)

В результате работы получен практический опыт по развёртыванию веб-сервера, организации логирования, реализации тестовых атак. Все действия проводились в рамках изолированной среды и сопровождались регистрацией действий для последующего документирования.

Кроме того, я являлся тимлидом нашей команды и проверял все файлы сокомандника. (Изучение требований работы заняло 3 часа, редакция текста на сайте заняла 4 часа, создание GitHub-репозитория, проверка его наполнения - 9 часов, посещение всех организационных онлайн-собраний - 8 часов).

Вариативная часть:

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 1. Установка Apache*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 2. Проверка работоспособности*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 3. Проверяем локальную доступность*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 4. Настройка логирования, запрос страницы*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 5. Анализ логов*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 6. Установка инструментов для тестирования*

Brute-force атака

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 7. Создаём простую форму входа на сайтие*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 8. Создаём простую форму входа на сайтие*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 9. Запуск атаки*

Изображение выглядит как снимок экрана, текст, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 10*

SQL-инъекция

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 11. Создаём уязвимый скрипт*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 12. Создаём уязвимый скрипт*

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 13. Создаём уязвимый скрипт*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 14. Эмулируем атаку*

*--batch — не задавать вопросы*

*--level=3 --risk=3 — расширенный анализ*

*--random-agent — имитирует разные браузеры*

*SQLmap начнёт проверку уязвимостей. Даже если реальной БД нет — он всё равно будет писать в лог* ***попытки SQL-инъекции*** *(через URL-параметры)*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 15*

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 16*

Изображение выглядит как снимок экрана, текст

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 17*

DoS-Атака

Изображение выглядит как текст, снимок экрана, Графическое программное обеспечение, дизайн

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 18. Запускаем атаку*

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, Значок на компьютере

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 19. Страница бесконечно пытается обновиться, значит атака работает*

Изображение выглядит как текст, снимок экрана, Шрифт, информация

Контент, сгенерированный ИИ, может содержать ошибки.

*Рисунок 20. Логи*

**Заключение**

В ходе проектной практики были достигнуты следующие результаты:

Настройка GitHub репозитория

Изучение HTML

Установлен и настроен веб-сервер Apache в изолированной среде (локальный сегмент 127.0.0.1);

Включено ведение логов доступа и ошибок (access.log и error.log);

Проведены три успешные имитации атак:

**Brute-force** с использованием утилиты hydra;

**SQL-инъекция** через sqlmap с реальной базой данных SQLite;

**DoS-атака (Slowloris)** с перегрузкой Apache;

Таким образом, поставленная цель — была успешно достигнута, задачи выполнены в полном объёме, результаты задокументированы и подтверждены логами и практическими тестами и коммитами на GitHub. Общее затраченное время: 76 часов

Приложения:  
[GitHub команды](https://github.com/swelock/PRACTIC_1_KURS/tree/main)