**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение высшего образования**

**«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**



**МОСКОВСКИЙ ПОЛИТЕХ**

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ  
КАФЕДРА «ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ»

**ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ**

**Выполнил**: ст. гр. 241-353 Рогинская А.Е.

**Руководитель**: Кесель С. А., к.т.н., доцент кафедры «Информационная безопасность»

**Место проведения**: Московский Политех, лаборатория «Программно-аппаратных средств обеспечения информационной безопасности»

**Москва – 2025**

**ОГЛАВЛЕНИЕ**

[**ВВЕДЕНИЕ 3**](#_heading=h.hxbmdhhmuqnm)

[**ГЛАВА 1. Изучение MITRE ATT&CK и OWASP 5**](#_heading=h.b0jkdyhvcxze)

[1.1 MITRE ATT&CK 5](#_heading=h.jb5ejle9vrj9)

[1.2 OWASP 7](#_heading=h.ksrvylhibha7)

[1.3 Практическая часть, анализ инцидента MGM Resorts 8](#_heading=h.pmbmx9p5m7qv)

[**ГЛАВА 2. Работа в рамках проектной деятельности. 11**](#_heading=h.ilcis3gmw0en)

[2.1. Безопасность веб-приложений (PortSwigger Labs) 11](#_heading=h.5dy8eomfgjay)

[2.2 Разработка приложений на Flask и шаблонизатор Jinja2 13](#_heading=h.3rswq6bj53nh)

[**ГЛАВА 3. Работа в рамках учебной практики. 15**](#_heading=h.uactxj69mopm)

[3.1 Базовая часть, работа с Git и GitHub 15](#_heading=h.z179vw8xfo8r)

[3.2 Вариативная часть 16](#_heading=h.g98p85mrsucf)

[3.3 Взаимодействие с партнерами. 17](#_heading=h.ltaqk0jqkt71)

[**ЗАКЛЮЧЕНИЕ 18**](#_heading=h.7p0hjm9as70v)

[**СПИСОК ЛИТЕРАТУРЫ 19**](#_heading=h.h29sl5e584kc)

# 

# **ВВЕДЕНИЕ**

Учебная практика, выполненная Рогинской Александрой Евгеньевной, ст. гр. 241-353, включает в себя комплексное изучение теоретических и практических аспектов в области информационной безопасности, веб-разработки и анализа защищённости. Основное внимание было уделено таким направлениям, как выявление уязвимостей веб-приложений, разработка безопасных приложений с использованием фреймворка Flask, анализ сетевого трафика, а также реализация механизмов автоматического реагирования на атаки.

Цель учебной практики — получить углублённые знания и практические навыки в области информационной безопасности, освоив современные подходы к защите веб-приложений, мониторингу активности и выявлению угроз.

Для достижения поставленной цели были решены следующие задачи:

1. изучение актуальных угроз и уязвимостей веб-приложений на основе практики в специализированных лабораториях (PortSwigger);
2. освоение инструментов анализа трафика и эксплуатации уязвимостей (Burp Suite, Scapy);
3. изучение теоретической базы и методических стандартов в области ИБ (OWASP Top 10, MITRE ATT&CK);
4. освоение средств управления версиями и командной разработки (Git и GitHub);
5. разработка безопасных веб-приложений с использованием Flask и шаблонизатора Jinja2;
6. реализация системы мониторинга HTTP-трафика и автоматического выявления вредоносных запросов с последующей блокировкой источников атак.

Объектом исследования являются методы и средства обеспечения безопасности веб-приложений, а также инструменты анализа и фильтрации сетевого трафика.

В результате практики был сформирован целостный взгляд на современные угрозы информационной безопасности и эффективные способы противодействия им. Полученные знания и навыки стали важным шагом на пути профессионального становления в сфере ИБ и могут быть использованы в дальнейших учебных и практических проектах.

# ГЛАВА 1. Изучение MITRE ATT&CK и OWASP

### 1.1 MITRE ATT&CK

MITRE ATT&CK представляет собой структуру, описывающую поведение противников на основе наблюдений за реальными кибератаками. Структура предназначена для помощи организациям в понимании того, как действуют злоумышленники, и как можно построить защиту.

Основные компоненты матрицы MITRE ATT&CK:

1. **Тактики (Tactics)** — это цели, которых стремится достичь злоумышленник. Примеры тактик: «Получение начального доступа», «Закрепление», «Обход защиты», «Извлечение данных» и др.

2. **Техники (Techniques)** — конкретные способы реализации тактик. Например, в рамках тактики «Получение начального доступа» может использоваться техника «Фишинг» (T1566), «Эксплуатация уязвимости» и др.

3. **Подтехники (Sub-techniques)** — более детализированные способы реализации техник. Например, «Фишинг с вложением» или «Фишинг со ссылкой».

4. **Процедуры (Procedures)** — реальные примеры атак, описывающие, как конкретные группы APT используют те или иные техники. Например, группа APT29 может использовать фишинг с целью установки вредоносного ПО.

5. **Платформы** — классификация техник в зависимости от ОС или среды: Windows, Linux, macOS, мобильные платформы, облачные и сетевые среды.

6. **Mitigations и Detections** — меры по противодействию (смягчению) атакам и индикаторы, по которым можно распознать применение техник.

Матрица MITRE ATT&CK широко применяется при построении моделей угроз (Threat Modeling), организации красных и синих команд (Red/Blue Teaming), тестировании на проникновение, формировании рекомендаций по защите и расследовании инцидентов.

### 1.2 OWASP

OWASP (Open Worldwide Application Security Project) — это глобальное некоммерческое сообщество, нацеленное на улучшение безопасности программного обеспечения. Оно предоставляет бесплатные и открытые материалы, стандарты, инструменты и рекомендации, направленные на обеспечение безопасности приложений.

Ключевые инициативы OWASP:

1. **OWASP Top 10** — перечень десяти наиболее критичных уязвимостей веб-приложений. Последняя версия (2021 года) включает:  
 - A01:2021 – Нарушение контроля доступа (Broken Access Control)  
 - A02:2021 – Криптографические сбои  
 - A03:2021 – Инъекции (SQL, OS и др.)  
 - A04:2021 – Небезопасный дизайн  
 - A05:2021 – Ошибки конфигурации  
 - A06:2021 – Уязвимые и устаревшие компоненты  
 - A07:2021 – Идентификация и аутентификация  
 - A08:2021 – Сбой программного обеспечения и данных  
 - A09:2021 – Недостаточный журнал событий и мониторинг  
 - A10:2021 – Серверные запросы с открытым перенаправлением

2. **OWASP ASVS (Application Security Verification Standard)** — стандарт оценки безопасности приложений, включающий уровни верификации от базовой проверки до полноценного анализа архитектуры.

3. **OWASP SAMM (Software Assurance Maturity Model)** — модель зрелости процессов разработки безопасного ПО. Предназначена для оценки текущего уровня безопасности SDLC и планирования его улучшения.

4. **OWASP ZAP (Zed Attack Proxy)** — бесплатный инструмент для автоматизированного тестирования безопасности веб-приложений. Один из самых популярных среди тестировщиков и исследователей.

5. **Cheat Sheet Series** — серия кратких и практических рекомендаций по реализации безопасных функций (например, безопасная аутентификация, обработка ошибок, защита от XSS).

Другие важные проекты OWASP включают Dependency-Check (анализ уязвимых библиотек), Security Knowledge Framework, Threat Dragon (инструмент моделирования угроз) и другие.

OWASP обеспечивает открытый доступ к лучшим практикам и инструментам, поддерживает сообщество разработчиков и специалистов по ИБ, способствует внедрению DevSecOps подходов.

### 1.3 Практическая часть, анализ инцидента MGM Resorts

Описание инцидента: MGM Resorts (сентябрь 2023)

В сентябре 2023 года корпорация MGM Resorts подверглась масштабной кибератаке, которая нарушила работу ИТ-инфраструктуры: игровые автоматы, системы бронирования отелей, мобильные приложения и веб-сайты оказались частично недоступны. Атака приписывается группировке Scattered Spider (возможно, аффилированной с BlackCat/ALPHV).

Этапы атаки и примененные TTPs (тактики, техники, процедуры):

**1. Получение начального доступа (Initial Access):**  
 - Техника: Phishing: Voice Phishing (Vishing) (T1566.002)\*  
 Злоумышленники использовали социальную инженерию, позвонив в техническую поддержку MGM и убедив её сбросить учётные данные.

**2. Использование действительных учётных записей (Valid Accounts):**  
 - Техника: Valid Accounts (T1078)  
 Полученные логины и пароли использовались для доступа к внутренним системам с реальными привилегиями.

**3. Закрепление (Persistence):**  
 - Техника: Account Manipulation (T1098)   
 Создание новых аккаунтов и изменение политик безопасности для сохранения доступа.

**4. Повышение привилегий (Privilege Escalation):**  
 - Техника: Exploitation for Privilege Escalation (T1068)   
 Использование уязвимостей в ПО или незащищенных конфигураций для получения администраторских прав.

**5. Обход защиты (Defense Evasion):** - Техника: Indicator Removal on Host (T1070)  
 Удаление логов, очистка следов активности в системах журналирования.

**6. Воздействие (Impact):**  
 - Техника: Data Encrypted for Impact (T1486)   
 Предположительно, была активирована шифровка данных, что парализовало бизнес-процессы.

Последствия и уроки:  
 Инцидент MGM демонстрирует значимость человеческого фактора и важность Zero Trust подхода. Уязвимость оказалась не столько в технологиях, сколько в процессах аутентификации и реагирования на запросы.

Меры предотвращения, основанные на рекомендациях OWASP:  
 - Внедрение многофакторной аутентификации (MFA)  
 - Ограничение прав доступа по принципу наименьших привилегий (Least Privilege)  
 - Регулярное обучение персонала  
 - Мониторинг аномальной активности (SIEM)  
 - Защита журналов и контроль за их целостностью

# ГЛАВА 2. Работа в рамках проектной деятельности.

### 2.1. Безопасность веб-приложений (PortSwigger Labs)

В рамках лабораторных заданий на платформе PortSwigger Web Security Academy я получила фундаментальные и практические знания по обеспечению безопасности веб-приложений. Эти задания были направлены на изучение и освоение типовых уязвимостей, с которыми сталкиваются современные веб-сервисы, а также методов их поиска и эксплуатации.

**Полученные знания и навыки:**

1. Изучение и эксплуатация основных уязвимостей:
   1. Path Traversal (обход каталогов): освоила технику, позволяющую получать доступ к файлам на сервере, к которым пользователь не должен иметь доступ. На практике научилась использовать специальные символы (../) для выхода за пределы разрешённого каталога и доступа, например, к файлам конфигурации и системным логам.
   2. Broken Access Control (нарушение контроля доступа): изучила примеры отсутствующего или некорректно реализованного механизма контроля доступа, что позволяет злоумышленнику выполнять действия, предназначенные только для других пользователей или администраторов.
   3. Horizontal Privilege Escalation (горизонтальное повышение привилегий): научилась идентифицировать ситуации, когда пользователь может получить доступ к данным других пользователей, подменяя, например, ID в URL или параметрах формы.
   4. Vertical Privilege Escalation (вертикальное повышение привилегий): освоила приёмы, при которых обычный пользователь получает доступ к функциям администратора или другим повышенным привилегиям за счёт ошибок в проверке прав доступа.
2. Работа с Burp Suite:
   1. Научилась перехватывать HTTP-запросы между браузером и сервером, исследовать их структуру и содержимое.
   2. Освоила модификацию параметров запроса, включая GET/POST-параметры, cookie-файлы, заголовки (headers), что позволяет тестировать веб-приложение на уязвимость к различным видам атак.
   3. Использовала инструменты Repeater и Intruder для повторной отправки изменённых запросов и автоматизации атак.
   4. Научилась распознавать ответы сервера на вредоносные запросы, интерпретировать HTTP-статусы, сообщения об ошибках и сигналы, указывающие на уязвимости.
3. Понимание атакующих техник:
   1. Получила представление о том, какими способами злоумышленники анализируют приложение, находят слабые места в логике и пытаются их использовать в обход механизмов защиты.
   2. Научилась мыслить как атакующий, чтобы эффективнее защищать приложения — понимать, где и как могут быть реализованы попытки обхода защиты, и какие меры предосторожности следует принять.

### 2.2 Разработка приложений на Flask и шаблонизатор Jinja2

В ходе учебной практики я освоила фреймворк Flask для создания веб-приложений на языке Python, а также научилась использовать шаблонизатор Jinja2 для генерации HTML-страниц. Flask — лёгкий, но мощный инструмент, подходящий как для обучения, так и для создания реальных приложений.

**Полученные знания и умения:**

1. Создала полноценные веб-приложения с использованием Flask:
   1. Реализовала маршруты (routes), обработку запросов, динамическую генерацию контента;
   2. Использовала декораторы для определения логики обработки URL-адресов;
2. Работала с HTML-шаблонами с использованием Jinja2:
   1. Освоила синтаксис шаблонов: наследование шаблонов, циклы, условия, вывод переменных;
   2. Реализовала динамическое отображение данных и реакцию интерфейса на действия пользователя;
3. Реализовала функции работы с формами:
   1. Обработка пользовательского ввода через методы POST/GET;
   2. Валидация данных, отображение ошибок;
4. Интеграция базы данных:
   1. Использовала SQLite для хранения информации о пользователях и публикациях;
   2. Научилась работать с ORM (SQLAlchemy): создание таблиц, добавление и извлечение данных;
5. Реализовала базовые функции системы аутентификации:
   1. Регистрация пользователей;
   2. Авторизация и выход из системы;
   3. Защита маршрутов, доступных только авторизованным пользователям.

# ГЛАВА 3. Работа в рамках учебной практики.

### 3.1 Базовая часть, работа с Git и GitHub

В процессе выполнения практики я освоила основы работы с распределённой системой контроля версий Git и научилась использовать платформу GitHub для хранения и управления проектами. Работа с системой контроля версий является важной частью современной командной разработки и обеспечивает прозрачность, отслеживаемость и восстановление изменений в исходном коде.

**Полученные навыки:**

1. Освоила базовые команды Git через терминал:
   1. git clone — для клонирования удалённых репозиториев на локальную машину;
   2. git status, git add, git commit — для отслеживания, подготовки и фиксации изменений;
   3. git branch, git checkout — для управления ветками разработки;
   4. git pull, git push — для синхронизации локальных и удалённых репозиториев.
2. Научилась работать с GitHub:
   1. Поняла структуру проектов, размещённых в публичных и приватных репозиториях;
   2. Ознакомилась с интерфейсом GitHub и его возможностями: issues, pull requests, документация (README), управление участниками;
   3. Научилась вносить изменения в проекты и отправлять их в удалённый репозиторий;
3. Получила понимание принципов командной разработки:
   1. Использование ветвления для параллельной работы над функционалом;
   2. Ревью кода и слияние веток;
   3. Разрешение конфликтов и контроль изменений.

### 3.2 Вариативная часть

В рамках практической части проекта был разработан и успешно протестирован прототип системы мониторинга веб-трафика с возможностью автоматического обнаружения вредоносной активности.

**Конкретные достигнутые результаты:**

1. Разработан Python-скрипт, отслеживающий входящий HTTP-трафик на порт 80 и анализирующий каждый запрос;
2. Реализованы функции анализа количества запросов от одного источника в единицу времени для выявления DoS-атак;
3. Внедрена простейшая система сигнатурного анализа запросов на наличие признаков SQL-инъекций (например, ' OR 1=1 --, UNION SELECT, --);
4. Автоматическая блокировка IP-адресов реализована с использованием команд ipset и iptables;
5. Проведено тестирование системы с использованием curl для эмуляции массовых и вредоносных запросов — в каждом случае срабатывало автоматическое добавление источника в список блокировки;

Таким образом, была создана работоспособная система базового уровня, способная обнаруживать и устранять некоторые виды атак в реальном времени.

### 3.3 Взаимодействие с партнерами.

В рамках учебной практики я приняла участие в карьерном марафоне 22 апреля 2025 года. Это мероприятие стало ценным дополнением к практической части, которое позволило расширить представление о профессиональной среде в сфере информационной безопасности и других направлений IT, я также узнала про возможности и преимущества стажировок.

**Что было получено в результате участия:**

1. Общение с представителями компаний — спонсоров вуза и проекта:
   1. Получила возможность задать вопросы специалистам и представителям HR-служб;
   2. Узнала, какие компетенции и навыки особенно востребованы у начинающих специалистов в сфере ИБ;
   3. Изучила предложения от компаний, представленных на мероприятии.
2. Информация о стажировках и трудоустройстве:
   1. Узнала о текущих и предстоящих программах стажировок, доступных для студентов и выпускников;
   2. Ознакомилась с требованиями к кандидатам, условиями участия и перспективами трудоустройства;
   3. Получила представление о том, как формируется карьерный путь в сфере.
3. Анализ современного рынка информационной безопасности:
   1. Услышала мнения экспертов о ключевых трендах в отрасли: рост числа инцидентов, развитие облачных технологий, важность реагирования на инциденты и автоматизации защиты;
   2. Поняла, что рынок ИБ динамичен, требует постоянного профессионального развития и адаптации к новым угрозам.

# ЗАКЛЮЧЕНИЕ

В результате прохождения учебной практики я получила не только теоретические знания, но и практические навыки, охватывающие ключевые аспекты информационной безопасности. Выполнение лабораторных работ позволило мне понять принципы работы с веб-уязвимостями, освоить инструменты анализа и эксплуатации, научиться безопасной разработке веб-приложений, а также применить знания для создания собственной системы защиты от сетевых атак.

Наиболее ценным результатом стал практический опыт создания и тестирования системы мониторинга HTTP-трафика с автоматическим реагированием. Я научилась выявлять потенциально опасные запросы и принимать меры по их блокировке на уровне сетевого фильтра.

# СПИСОК ЛИТЕРАТУРЫ

1. Введение в CSS верстку: <https://developer.mozilla.org/ru/docs/Learn_web_development/Core/CSS_layout/Introduction>
2. DevTools для «чайников»: <https://habr.com/ru/articles/548898/>
3. Элементы HTML: <https://developer.mozilla.org/ru/docs/Web/HTML/Element>
4. Основы HTML: <https://developer.mozilla.org/ru/docs/Learn_web_development/Getting_started/Your_first_website/Creating_the_content>
5. Основы CSS: <https://developer.mozilla.org/ru/docs/Web/CSS>
6. <https://doka.guide/>
7. Официальная документация Git: <https://git-scm.com/book/ru/v2>
8. <https://skillbox.ru/media/code/chto_takoe_git_obyasnyaem_na_skhemakh/>
9. Бесплатный курс на Hexlet по Git: <https://ru.hexlet.io/courses/intro_to_git>
10. Уроки по Markdown: <https://ru.hexlet.io/lesson_filters/markdown>
11. Лабораторная работа по FLASK: <https://romansimakov-reddatabaselab.readthedocs.io/ru/latest/flaskr.html>
12. Руководство для начинающих по шаблонам Jinja в Flask: <https://proglib.io/p/rukovodstvo-dlya-nachinayushchih-po-shablonam-jinja-v-flask-2022-09-05>
13. Portswigger: <https://portswigger.net/>