Федеральное государственное автономное образовательное учреждение высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра «Информационная безопасность»

Направление подготовки/ специальность: 10.03.01 Информационная безопасность

ОТЧЕТ

по проектной практике

Студент: Куприянова Юлия Андреевна 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

В данном проекте была поставлена задача изучить безопасность веб-сервера с использованием контейнеризации. Для этого мной был создан статический сайт, размещенный в Docker-контейнере, и проведен анализ безопасности образа с помощью специализированного инструмента. Работа позволила освоить основные навыки работы с Git, Docker и средствами проверки уязвимостей.

Общая информация о проекте

Название проекта: Лаборатория успеха

Цели проекта: Разработать варианты применения геймификации в сфере дополнительного образования кадров и создать прототип.

Задачи:

1. Проанализировать рынок, текущие предложения, аудиторию.
2. Создать игровую концепцию, направленную на мотивацию развития профессиональных навыков и личностных качеств, обучающую основам менеджмента и планирования.
3. Разработать прототип игры.
4. Презентовать полученные данные и вариант реализации.

Суть проекта: Разработка и реализация различных методов применения геймификации в формате различных настольных и компьютерных игр, позволяющих игроку применить и повысить свои управленческие, административные и профессиональные навыки и увеличить его стремление к дальнейшему развитию.

Описание полученных результатов выполненных задач: Проведен анализ возможностей команды, исследование рынка, аудитории и возможных вариантов реализации проекта. Распределена структура работы проекта, принято решение по текущей разработке. Сформулированы основные механики игр, их концепция. Проделана работа над визуальным наполнением игр. Сделаны первые прототипы настольной игры, начаты наработки видеоигры. Проведены первые тестирования настольной игры, внесены изменения.

Промежуточный продуктовый результат: В ходе работы над проектом была создана настольная экономическая игра основанная на механике расстановки рабочих и зданий, направленная на стратегическое мышление, управление ресурсами, планирование развития и взаимодействие с другими игроками. Основной целью игры является построение успешной компании, конкурируя с другими игроками.

Конкуренция за локации и ресурсы, бонусы за сотрудничество и необходимость взвешивания рисков и наград, заставляют игроков принимать сложные решения и стратегически планировать свои действия.

Кроме этого, была начата разработка видеоигры о развитии промышленных цепочек, планировании и менеджменте собственного бизнеса с возможностью многопользовательской игры и конкуренции. Получены первые наработки.

Заключение о проекте: За время работы в семестре были расставлены задачи, цели проекта, построено представление желаемого результата. В ходе работы были получены прототипы продуктовых результатов, проведены тестирования, внесены изменения. Участники показали хороший уровень подготовки и вовлеченность в проект. В дальнейшем планируется продолжение развития настольной игры и разработка видеоигры.

Общая характеристика деятельности организации (заказчика проекта)

Наименование заказчика: Федеральное государственное автономное образовательное учреждение высшего образования «Московский Политехнический университет»

Организационная структура:

1. Ректор (несет ответственность за полное руководство университетом)
2. Проректоры (непосредственно подчиняются ректору и отвечают за отдельные направления деятельности университета)
3. Ректорат (коллегиальный орган управления, возглавляемый ректором, в который входят проректоры и другие ключевые руководители)

Описание деятельности:

Московский политехнический университет — современный вуз, который готовит специалистов как технической, так и гуманитарной направленности для крупнейших компаний России.

Описание задания по проектной практике

Обязательная часть: студент осваивает работу с системой контроля версий Git с использованием репозитория на платформе GitHub или GitVerse. Необходимо научиться применять основные команды Git: клонирование репозитория, фиксация изменений, отправка изменений в удаленный репозиторий, создание и работа с ведением версий. Требуется регулярно сохранять изменения с подробными и информативными комментариями к каждому коммиту.

Все материалы оформляются в формате Markdown. Обязательно создать статический веб-сайт с использованием HTML/CSS или генератора сайтов, например Hugo. На сайте должны быть размещены следующие разделы: главная страница с аннотацией проекта, описание проекта, информация об участниках с указанием вклада каждого, новостной журнал с не менее чем тремя публикациями и раздел с полезными ссылками. На сайт необходимо добавить графические материалы.

В рамках практики предусмотрено взаимодействие с партнёром. Это может быть встреча, стажировка или участие в мероприятиях, таких как конференции или хакатоны. Результаты взаимодействия оформляются в виде отчёта в формате Markdown.

Вариативная часть: моим заданием вариативной части было исследование безопасности веб-сервера с использованием контейнеризации. Для выполнения этого задания я выбрал развертывание статического сайта, который был создан мной ранее в ходе проектной практики. В качестве веб-сервера использовался

Nginx, запущенный внутри Docker-контейнера, что позволило обеспечить изоляцию приложения и упростить управление средой выполнения.

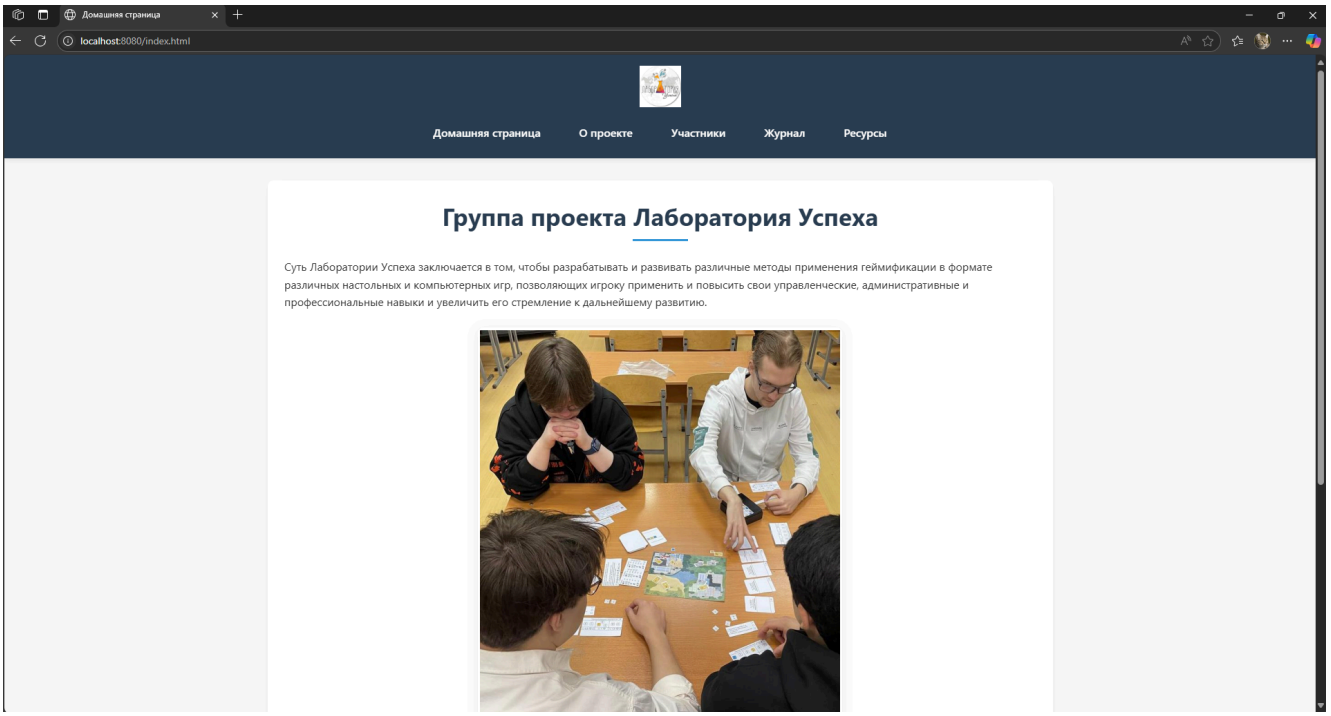
Для создания контейнерного образа был написан простой Dockerfile, а также настроен конфигурационный файл nginx.conf, обеспечивающий корректную обработку и выдачу локальных HTML и CSS файлов. Такой подход позволил не только развернуть веб-сервер, но и подготовить удобную и повторяемую среду для дальнейшего анализа безопасности.

Для анализа безопасности образа и выявления уязвимостей применялась специализированная утилита Trivy, предназначенная для сканирования Docker-образов на предмет известных проблем в компонентах и настройках, а также для оценки безопасности встроенного кода.

Описание достигнутых результатов по проектной практике

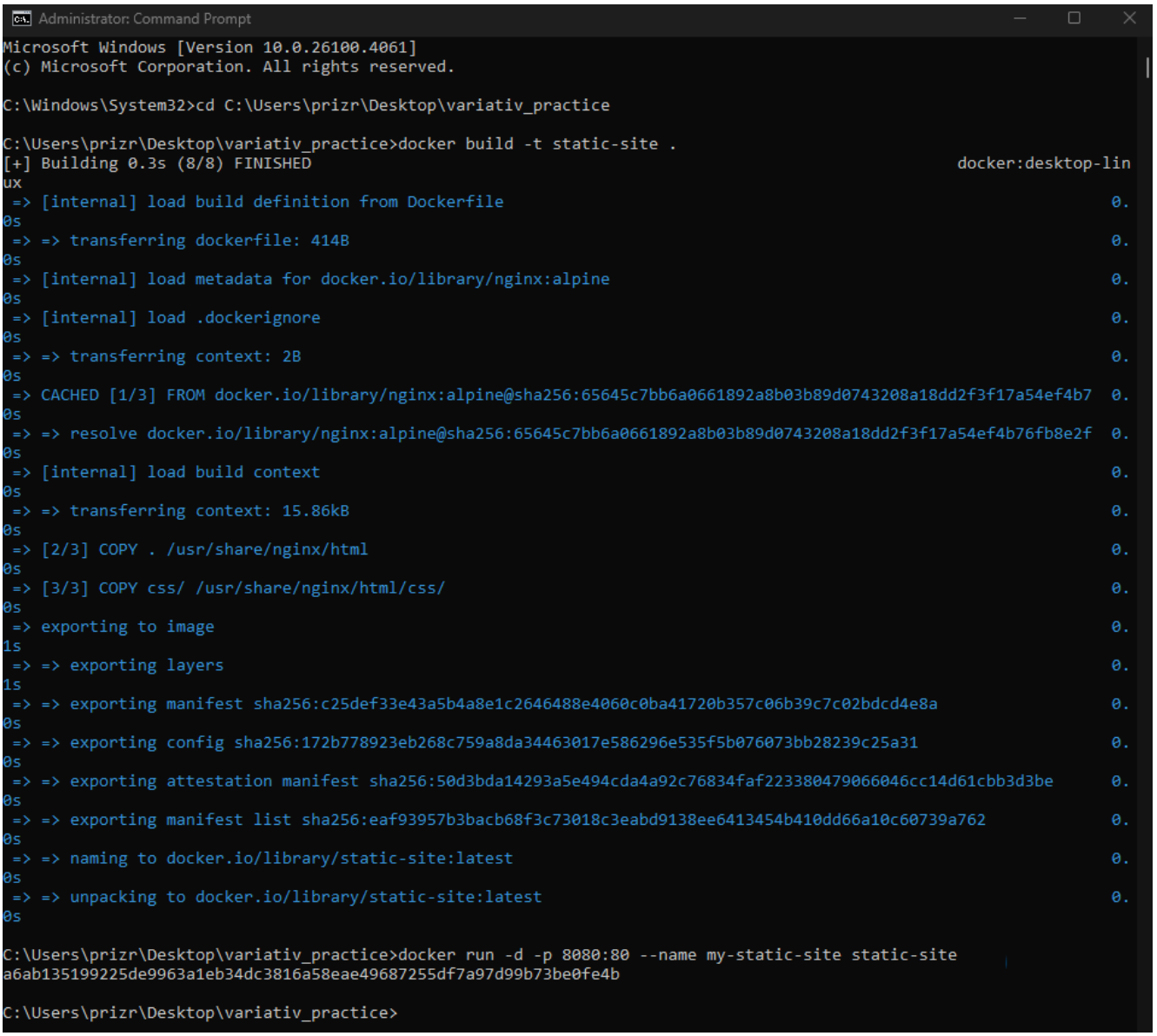
В ходе проектной практики мной были выполнены все поставленные задачи по освоению системы контроля версий Git и созданию статического веб-сайта. Я научился работать с основными командами Git, включая клонирование репозитория с платформы GitHub, фиксацию изменений с подробными комментариями к каждому коммиту, отправку изменений в удаленный репозиторий, а также создание и управление ветками для ведения версий проекта.

Все материалы проекта были оформлены в соответствии с требованиями в формате Markdown. Мной был создан статический веб-сайт *(иллюстрация 1)* с использованием HTML и CSS, который включает все необходимые разделы: главную страницу с аннотацией проекта, подробное описание проекта, информацию об участниках с указанием вклада каждого, новостной журнал с тремя публикациями и раздел с полезными ссылками. Для улучшения визуального восприятия на сайт были добавлены соответствующие графические материалы.

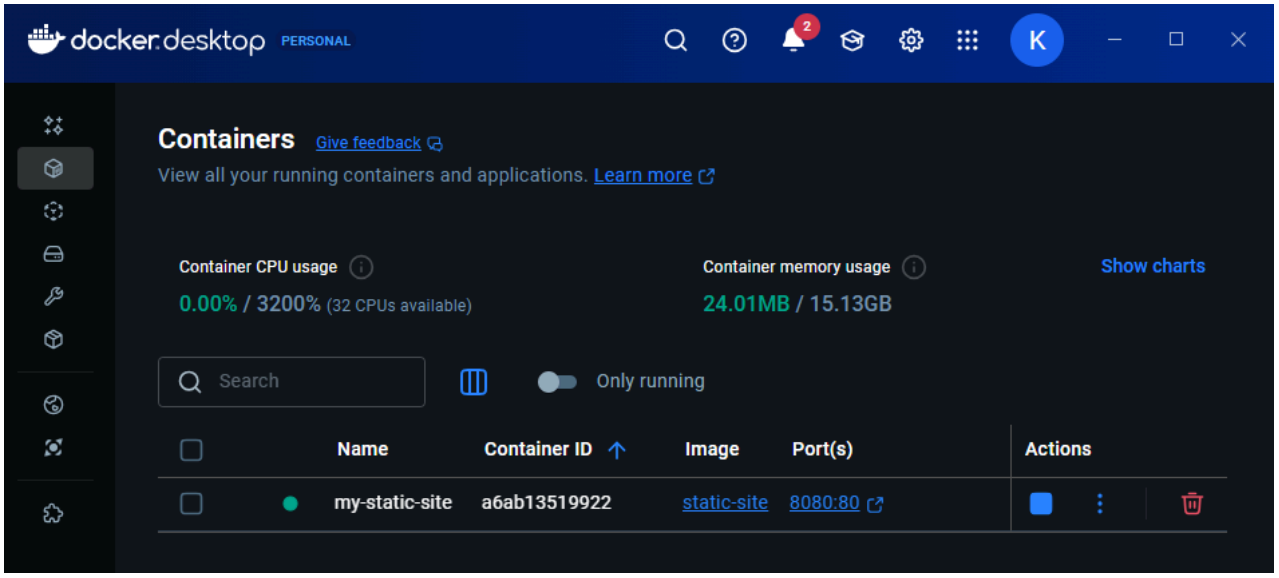


(иллюстрация 1)

Для проведения исследования был создан Dockerfile *(иллюстрация 2 и 3)*, в котором на базе легкого образа Alpine Linux развернут веб-сервер Nginx, настроенный на обслуживание локальных статических файлов HTML и CSS. После сборки образа был запущен контейнер с этим образом, что позволило протестировать работу веб-сервера в изолированной и управляемой среде. Контейнер успешно обслуживал статический сайт, обеспечивая стабильную работу сервера.



(иллюстрация 2)



(иллюстрация 3)

По результатам сканирования Docker-образа my-static-site с помощью Trivy *(иллюстрация 4)* были обнаружены две уязвимости высокого уровня (High) в системной библиотеке libxml2, которая входит в состав базового образа Alpine Linux версии 3.21.3. Обнаруженные уязвимости имеют идентификаторы CVE-2025-32414 и CVE-2025-32415 и связаны с выходом за пределы допустимой области памяти при разборе XML-структур. Это может привести как к сбою процесса веб-сервера, так и, в случае эксплуатации, к выполнению произвольного кода злоумышленником.

```
C:\Windows\System32>cd C:\Users\prizr\Desktop\trivy_0.62.1_windows-64bit

C:\Users\prizr\Desktop\trivy_0.62.1_windows-64bit>trivy --version
Version: 0.62.1

C:\Users\prizr\Desktop\trivy_0.62.1_windows-64bit>trivy image my-static-site
2025-05-22T22:45:50+03:00       INFO    [vulndb] Need to update DB
2025-05-22T22:45:50+03:00       INFO    [vulndb] Downloading vulnerability DB...
2025-05-22T22:45:50+03:00       INFO    [vulndb] Downloading artifact...        repo="mirror.gcr.io/aquasec/trivy-db:2"
64.23 MiB / 64.23 MiB [-------------------------------------------------------------------------] 100.00% 4.07 MiB p/s 16s
2025-05-22T22:46:07+03:00       INFO    [vulndb] Artifact successfully downloaded       repo="mirror.gcr.io/aquasec/trivy-db:2"
2025-05-22T22:46:07+03:00       INFO    [vuln] Vulnerability scanning is enabled
2025-05-22T22:46:07+03:00       INFO    [secret] Secret scanning is enabled
2025-05-22T22:46:07+03:00       INFO    [secret] If your scanning is slow, please try '--scanners vuln' to disable secret scanning
2025-05-22T22:46:07+03:00       INFO    [secret] Please see also https://trivy.dev/v0.62/docs/scanner/secret#recommendation for faster secret detection
2025-05-22T22:46:08+03:00       INFO    Detected OS     family="alpine" version="3.21.3"
2025-05-22T22:46:08+03:00       INFO    [alpine] Detecting vulnerabilities...   os_version="3.21" repository="3.21" pkg_num=68
2025-05-22T22:46:08+03:00       INFO    Number of language-specific files       num=0

Report Summary

┌──────────────────────────────┬────────┬─────────────────┬─────────┐
│            Target            │  Type  │ Vulnerabilities │ Secrets │
├──────────────────────────────┼────────┼─────────────────┼─────────┤
│ my-static-site (alpine 3.21.3) │ alpine │        2        │    -    │
└──────────────────────────────┴────────┴─────────────────┴─────────┘

Legend:
- '-': Not scanned
- '0': Clean (no security findings detected)


my-static-site (alpine 3.21.3)
==============================
Total: 2 (UNKNOWN: 0, LOW: 0, MEDIUM: 0, HIGH: 2, CRITICAL: 0)

┌─────────┬────────────────┬──────────┬────────┬───────────────────┬───────────────┬──────────────────────────────────────────────────────────────┐
│ Library │ Vulnerability  │ Severity │ Status │ Installed Version │ Fixed Version │                            Title                             │
├─────────┼────────────────┼──────────┼────────┼───────────────────┼───────────────┼──────────────────────────────────────────────────────────────┤
│ libxml2 │ CVE-2025-32414 │ HIGH     │ fixed  │ 2.13.4-r5         │ 2.13.4-r6     │ libxml2: Out-of-Bounds Read in libxml2                       │
│         │                │          │        │                   │               │ https://avd.aquasec.com/nvd/cve-2025-32414                   │
│         ├────────────────┤          │        │                   │               ├──────────────────────────────────────────────────────────────┤
│         │ CVE-2025-32415 │          │        │                   │               │ libxml2: Out-of-bounds Read in xmlSchemaIDCFillNodeTables    │
│         │                │          │        │                   │               │ https://avd.aquasec.com/nvd/cve-2025-32415                   │
└─────────┴────────────────┴──────────┴────────┴───────────────────┴───────────────┴──────────────────────────────────────────────────────────────┘
```

(иллюстрация 4)

Данные уязвимости устранены в версии libxml2 2.13.4-r6. В связи с этим рекомендуется обновить базовый образ Alpine Linux до версии 3.21.4 или выше либо вручную обновить библиотеку libxml2 при сборке Docker-образа.

Таким образом, проведенное исследование выявило конкретные риски безопасности, а также предложило пути их устранения, что существенно повышает надёжность и защищённость развернутого веб-сервера.

Заключение

В ходе проектной практики я освоил работу с системой контроля версий Git и платформой GitHub. Создал репозиторий, научился выполнять базовые команды, такие как клонирование, коммит, отправка изменений и создание веток. Все материалы проекта оформил в формате Markdown и разместил в репозитории.

Далее разработал статический веб-сайт на HTML и CSS. Сайт включает главную страницу с аннотацией, разделы с информацией о проекте и участниках, журнал новостей с не менее чем тремя публикациями, а также раздел с полезными ссылками. Для улучшения визуального восприятия добавил графические материалы.

Затем создал Dockerfile для контейнеризации сайта, используя базовый образ Alpine Linux. Собрал и запустил контейнер, который корректно отображает сайт.

Для оценки безопасности контейнера установил и настроил инструмент Trivy. Провел сканирование созданного образа и выявил две уязвимости высокой степени серьезности в библиотеке libxml2 версии 2.13.4-r5. Эти уязвимости связаны с выходом за пределы допустимой области памяти и уже исправлены в более новой версии 2.13.4-r6. В связи с этим рекомендую обновить базовый образ или выполнить обновление библиотек внутри контейнера для устранения обнаруженных проблем.

Таким образом, в ходе практики я получил опыт работы с системой контроля версий, создал и развернул статический сайт в Docker, а также провел анализ безопасности контейнерного образа. Результаты подчеркнули важность регулярного контроля и обновления образов для обеспечения безопасности веб-приложений.

Список использованной литературы

1. Введение в CSS верстку: https://developer.mozilla.org/ru/docs/Learn_web_development/Core/CSS_layout/Introduction
2. Элементы HTML: https://developer.mozilla.org/ru/docs/Web/HTML/Element
3. Основы HTML: https://developer.mozilla.org/ru/docs/Learn_web_development/Getting_started/Your_first_website/Creating_the_content
4. Основы CSS: https://developer.mozilla.org/ru/docs/Web/CSS
5. Официальная документация Git: https://git-scm.com/book/ru/v2