Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Фізико-технічний інститут
«Блокчейн та децентралізовані системи»
Лабораторна робота №3

Тема: „Дослідження безпечної реалізації та експлуатації децентралізованих додатків".
Мета роботи: «Отримання навичок роботи із децентралізованими додатками та оцінка безпеки інформації при їх функціонуванні».
Виконав:
студент групи ФІ-41мн
Должко Назарій

Для першого типу лабораторних робіт:
дослідження вимог OWASP (безпека web-додатків) та складання аналогічних вимог для обраної системи децентралізованих додатків.
Список вимог OWASP Top 10 до безпеки Web-додатків:

Список вимог OWASP Top 10 до безпеки Web-додатків:

1. Broken Access Control — неправильна перевірка доступу
2. Cryptographic Failures — слабке або відсутнє шифруванн
3. Injection — SQL, OS або інші ін'єкції
4. Insecure Design — поганий проект з точки зору безпеки
5. Security Misconfiguration — неправильні налаштування безпеки

6. Vulnerable and Outdated Components — небезпечні сторонні бібліотеки
7. Identification and Authentication Failures — проблеми з аутентифікацією
8. Software and Data Integrity Failures — незахищене оновлення або відробка коду
9. Security Logging and Monitoring Failures — відсутність логування та моніторингу
10. Server-Side Request Forgery (SSRF) — сервер виконує шкідливі запити

Аналогічні вимоги до децентралізованого додатку (на прикладі Uniswap)

1. Несправна перевірка доступу (Broken Access Control)

Uniswap використовує смарт-контракти, де критично важливо контролювати, хто має доступ до адміністрування функцій, наприклад, оновлення пулів чи додавання нових токенів.

Рішення:

- Використання `onlyOwner` / `AccessControl` від OpenZeppelin
- Відмова від централізованого адміністрування там, де це можливо (DAO governance)

---

2. Слабке або відсутнє шифрування (Cryptographic Failures)

Хоч Uniswap працює у публічному блокчейні Ethereum, приватність користувача залишається чутливим питанням.

Рішення:

- Додавання шифрування даних на фронтенді (наприклад, історії транзакцій)
- Захист конфіденційних операцій через інтеграцію з приватними рішеннями (наприклад, zk-SNARKs)

---

3. Ін'єкції (Injection)

Смарт-контракти Uniswap можуть бути ціллю для маніпуляцій через некоректно оброблені запити.

Рішення:

- Статичний аналіз коду смарт-контрактів
- Валідація всіх параметрів у функціях контракту

---

4. Поганий проект з точки зору безпеки (Insecure Design)

Недосконалий дизайн функцій (наприклад, swap) може призвести до фінансових втрат (наприклад, sandwich-атаки).

Рішення:

- Врахування типових DeFi атак при архітектурі
- Проведення незалежного аудиту безпеки дизайну

---

5. Невірні налаштування безпеки (Security Misconfiguration)

Фронтенд Uniswap (інтерфейс) також може бути вразливим через неправильні заголовки або політики доступу.

Рішення:

- Налаштування політик безпеки HTTP (CSP, CORS)
- Використання HTTPS та сертифікатів

---

6. Небезпечні сторонні компоненти (Vulnerable and Outdated Components)

Фронтенд Uniswap побудований на React з використанням багатьох npm-бібліотек.

Рішення:

- Регулярне оновлення залежностей
- Використання GitHub Dependabot або Snyk для виявлення вразливостей

---

7. Проблеми з аутентифікацією (Identification and Authentication Failures)

Uniswap не використовує класичну аутентифікацію, але залежить від підписів через гаманці.

Рішення:

- Перевірка валідності підпису користувача

- Впровадження багатофакторної валідації через гаманці з мультипідписом

---

8. Незахищене оновлення або відробка коду (Software and Data Integrity Failures)

Критичне значення має захист смарт-контрактів від небажаних змін.

Рішення:

- Використання immutable-смартконтрактів
- Перевірка підписів у DAO-рішеннях перед апгрейдом

---

9. Відсутність логування та моніторингу (Security Logging and Monitoring Failures)

Uniswap має обмежене логування подій, але потрібно відстежувати аномальні транзакції.

Рішення:

- Використання публічного моніторингу блокчейну (наприклад, Dune Analytics)
- Інтеграція з аналітичними системами для відстеження атак

---

10. SSRF (Server-Side Request Forgery)

У Uniswap SSRF є менш імовірним, але можливі загрози через API-запити на сторонні сервіси.

Рішення:

- Обмеження URL-адрес, на які дозволено надсилати запити
- Валідація всіх запитів API на фронтенді