НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. Ігоря СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

Звіт з виконання комп'ютерного практикуму

ОТРИМАННЯ НАВИЧОК РОБОТИ ІЗ ДЕЦЕНТРАЛІЗОВАНИМИ ДОДАТКАМИ ТА ОЦІНКА БЕЗПЕКИ ІНФОРМАЦІЇ ПРИ ЇХ ФУНКЦІОНУВАННІ

Виконали студенти
групи ФБ-31мн
Швець Максим,
Чикрій Кирило

Перевірила:
Селюх П.В.

Київ — 2024

ЗВІТ

Тема: Дослідження безпечної реалізації та експлуатації децентралізованих додатків.

Мета роботи: отримання навичок роботи із децентралізованими додатками та оцінка безпеки інформації при їх функціонуванні

Завдання на лабораторну роботу: дослідження вимог OWASP (безпека web-додатків) та складання аналогічних вимог для обраної системи децентралізованих додатків.

Вступ

В епоху цифрових технологій, безпека програмного забезпечення стає критично важливою для захисту даних та забезпечення надійності систем. Веб-додатки, які щодня використовують мільйони людей, часто стають мішенями для хакерських атак. Відповідно, ініціатива Open Web Application Security Project (OWASP) пропонує світові стандарти для забезпечення безпеки веб-додатків, зокрема, відомий список OWASP Top 10, що включає найпоширеніші загрози безпеки.

З іншого боку, децентралізовані додатки, що функціонують на технології блокчейн, здобувають популярність завдяки своїй прозорості, безпеці та відсутності одного контрольного центру. Однак, не дивлячись на вбудовані механізми безпеки, такі системи також схильні до унікальних викликів та загроз, які потребують особливої уваги та адаптації існуючих методик безпеки.

Цей реферат має на меті дослідити вимоги безпеки OWASP і розробити аналогічні вимоги для системи децентралізованих додатків, вибравши для детального аналізу платформу Ethereum. Завдяки цьому, ми спробуємо зрозуміти, як можна адаптувати визнані практики безпеки веб-додатків до світу децентралізованих технологій, враховуючи їх специфіку та потенційні ризики.

Огляд OWASP

Open Web Application Security Project (OWASP) є відкритим співтовариством, яке працює над покращенням безпеки програмного забезпечення. З моменту свого створення, ця ініціатива спрямована на те, щоб зробити інформацію про безпечні практики доступною для всіх зацікавлених сторін у процесі

розробки програмного забезпечення.

Історія та місія OWASP:
Заснована в 2001 році, місія OWASP полягає у створенні безкоштовних та відкритих ресурсів для веб-спільноти, що допомагає в ідентифікації та усуненні проблем безпеки веб-додатків. OWASP надає інструменти, документацію, форуми та навчальні курси для підвищення обізнаності про веб-безпеку серед розробників і технічних фахівців.

OWASP Top 10:
Серцем ініціативи OWASP є "OWASP Top 10", список десяти найбільш критичних ризиків безпеки веб-додатків. Цей список регулярно оновлюється та включає такі загрози:

Injection: Включає SQL, NoSQL, OS та LDAP ін'єкції. Такі уразливості дозволяють зловмисникам вводити власний шкідливий код, який система виконає.
Broken Authentication: Неправильно імплементовані механізми аутентифікації дозволяють зловмисникам вгадувати або перехоплювати користувацькі дані.
Sensitive Data Exposure: Незахищене зберігання чи передача чутливих даних може призвести до їх витоку.
XML External Entities (XXE): Ця уразливість дозволяє атакувати внутрішні системи, проводити DoS атаки, та здобувати доступ до файлів.
Broken Access Control: Недостатні обмеження доступу дозволяють зловмисникам отримувати несанкціонований доступ до функцій або даних.
Security Misconfiguration: Найбільш поширена уразливість, що включає неправильну конфігурацію безпеки.
Cross-Site Scripting (XSS): Дозволяє зловмисникам вставляти клієнтські скрипти в веб-сторінки, які переглядають інші користувачі.
Insecure Deserialization: Ця уразливість може призвести до віддалених атак на код.
Using Components with Known Vulnerabilities: Використання компонентів з відомими уразливостями.
Insufficient Logging & Monitoring: Недостатня реєстрація подій та моніторинг, що ускладнює виявлення або запобігання атакам.
Кожен пункт цього списку містить рекомендації щодо запобігання та виправлення виявлених уразливостей, а також методи їх виявлення.

Важливість OWASP для розробників:
Знання та розуміння OWASP Top 10 є необхідними для створення безпечних веб-додатків. Ці рекомендації допомагають розробникам визначити

пріоритети у виправленні уразливостей, а також формувати стратегії захисту від поширених атак.

Аналіз системи децентралізованих додатків: Ethereum

Ethereum є однією з провідних платформ для створення децентралізованих додатків (DApps), яка використовує технологію блокчейн не тільки для ведення транзакцій, але й для виконання смарт-контрактів. Ця платформа значно розширила можливості блокчейну, додавши до неї можливість програмування додатків, що виконуються в розподіленому середовищі.

Основні характеристики Ethereum:

Смарт-контракти: Це самовиконувальні контракти з умовами угоди прямо в коді контракту. Вони виконуються автоматично, коли задовольняються вказані умови, забезпечуючи високий рівень безпеки та надійності.
Ethereum Virtual Machine (EVM): Всі смарт-контракти виконуються на EVM, що дозволяє їм працювати на будь-якому обчислювальному обладнанні без потреби в специфічних адаптаціях.
Консенсус через Proof of Work (PoW): Забезпечує безпеку мережі та надійність доданих до блокчейну транзакцій. Ethereum планує перехід на Proof of Stake (PoS) для зниження енергетичних витрат та підвищення ефективності мережі.
Виклики та загрози безпеки:

Вразливість смарт-контрактів: Через високу складність та особливості мови програмування Solidity, смарт-контракти можуть містити помилки або уразливості, які можуть бути використані для атак.
Масштабованість: Так як кожен вузол у мережі обробляє кожну транзакцію, це може призвести до затримок і підвищення вартості транзакцій при збільшенні обсягу мережі.
Збереження конфіденційності: Стандартні смарт-контракти в Ethereum є повністю прозорими, що може бути проблемою для додатків, яким необхідна конфіденційність.
Рекомендації за OWASP:
Для застосування принципів безпеки OWASP до Ethereum необхідно розробити спеціальні рекомендації, які б враховували особливості децентралізованих додатків. Це включає:

Перевірка і тестування коду смарт-контрактів: Використання автоматизованих інструментів для аналізу та перевірки смарт-контрактів на

наявність вразливостей.
Обмеження прав доступу: Розробка механізмів контролю доступу, що забезпечують обмеження використання смарт-контрактів лише авторизованими користувачами.
Моніторинг та журналювання: Встановлення систем моніторингу для виявлення незвичайних транзакцій та можливих атак в реальному часі.

Розробка вимог безпеки для обраної системи децентралізованих додатків: Ethereum

Щоб забезпечити безпеку децентралізованих додатків (DApps), що працюють на платформі Ethereum, необхідно адаптувати і застосувати принципи безпеки, сформульовані OWASP, до особливостей цієї системи. Ось декілька ключових вимог безпеки, які можна використовувати як основу для розробки більш детальних рекомендацій:

Безпечне програмування смарт-контрактів:

Валідація входів: Всі входи в смарт-контракти мають бути строго валідовані для запобігання ін'єкціям та іншим видам атак.
Обмеження прав доступу: Смарт-контракти мають включати механізми контролю доступу, що обмежують виклики функцій виключно авторизованими користувачами.
Перевірка поведінки контрактів: Регулярне тестування і аудит коду смарт-контрактів на наявність вразливостей і логічних помилок.
Управління ідентифікацією та аутентифікацією:

Багаторівнева аутентифікація: Застосування багаторівневої аутентифікації для критичних транзакцій і адміністративних функцій у DApps.
Захист ключів: Забезпечення безпечного зберігання приватних ключів користувачів, використовуючи сучасні методи шифрування.
Захист конфіденційності даних:

Шифрування: Використання сильного шифрування для зберігання чутливих даних у блокчейні.
Конфіденційні транзакції: Розробка та впровадження механізмів для забезпечення конфіденційності транзакцій.
Масштабування та обробка помилок:

Шардінг і бічні ланцюги: Розгляд використання шардінгу та бічних ланцюгів для покращення масштабованості та ефективності обробки транзакцій.

Обробка помилок: Розробка механізмів для безпечного та елегантного відновлення після помилок або збоїв у DApps.
Моніторинг та відповідь на інциденти:

Журналювання та моніторинг: Впровадження розширеного журналювання та моніторингу діяльності для виявлення та реагування на можливі інциденти безпеки.
План відповіді на інциденти: Розробка детального плану дій на випадок безпекових порушень.
Ці вимоги слід розглядати як вихідну точку для глибшої адаптації стандартів OWASP до децентралізованих платформ. Розробка конкретних технічних та організаційних рішень дозволить підвищити рівень безпеки Ethereum та інших систем децентралізованих додатків.