МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки

Технологія блокчейн та розподілені системи
Лабораторна робота 3
Дослідження безпечної реалізації та експлуатації децентралізованих додатків.

Перевірив:
Селюх П.В

_____

Виконали:
Студенти групи ФБ-31мп
Снігур Антон
Тислицький Данііл
Чорний Анатолій

Київ 2024

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

OWASP (Open Web Application Security Project) - це спільнота, що займається розробкою і поширенням інформації про забезпечення безпеки веб-додатків. Вона складається з добровольців з усього світу, які працюють над створенням безкоштовних ресурсів, які допомагають організаціям і фахівцям у галузі інформаційної безпеки.

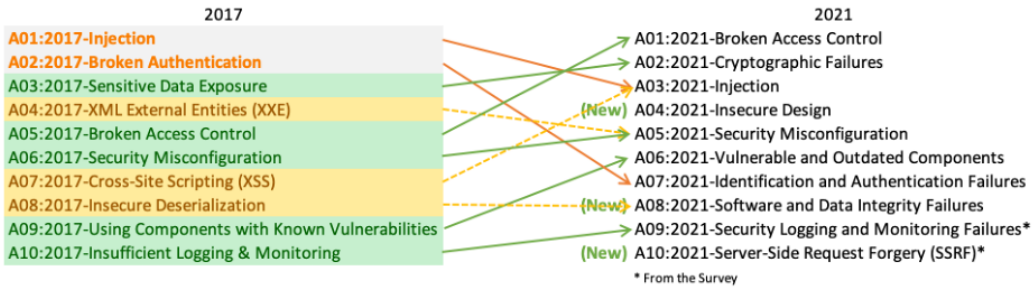
Основна мета OWASP - підвищення рівня безпеки веб-додатків шляхом надання розуміння потенційних загроз та вразливостей, які можуть бути використані для атак, а також надання рекомендацій і кращих практик для їх уникнення. OWASP випускає різноманітні матеріали, такі як списки вразливостей (наприклад, OWASP Top 10), рекомендації з безпеки, інструменти тестування безпеки та навчальні ресурси.

Дослідження вимог OWASP є частиною широкого спектру документів і матеріалів, розроблених цією організацією для підвищення безпеки веб-додатків. Одним із найбільш відомих документів є OWASP Top 10, який представляє перелік найбільш поширених та критичних вразливостей у веб-додатках.

Щодо OWASP Top 10 – то це список найбільш поширених та критичних вразливостей у веб-додатках, який складається і оновлюється OWASP щорічно або раз у кілька років. Цей список створюється на основі аналізу реальних атак та інцидентів безпеки й експертної експертизи з безпеки програмного забезпечення. Він допомагає організаціям та розробникам краще розуміти потенційні загрози, які можуть вплинути на їх веб-додатки, і надає рекомендації щодо кращих практик з їх уникнення. Також цей список може служити основою для розробки стратегій безпеки, виконання аудитів безпеки та вдосконалення процесів розробки програмного забезпечення.

Top 10 Web Application Security Risks

There are three new categories, four categories with naming and scoping changes, and some consolidation in the Top 10 for 2021.



- **A01:2021-Broken Access Control** moves up from the fifth position; 94% of applications were tested for some form of broken access control. The 34 Common Weakness Enumerations (CWEs) mapped to Broken Access Control had more occurrences in applications than any other category.
- **A02:2021-Cryptographic Failures** shifts up one position to #2, previously known as Sensitive Data Exposure, which was broad symptom rather than a root cause. The renewed focus here is on failures related to cryptography which often leads to sensitive data exposure or system compromise.
- **A03:2021-Injection** slides down to the third position. 94% of the applications were tested for some form of injection, and the 33 CWEs mapped into this category have the second most occurrences in applications. Cross-site Scripting is now part of this category in this edition.
- **A04:2021-Insecure Design** is a new category for 2021, with a focus on risks related to design flaws. If we genuinely want to “move left” as an industry, it calls for more use of threat modeling, secure design patterns and principles, and reference architectures.

OWASP Top 10 на 2021 рік включає в себе:

- Broken Access Control (Несправний контроль доступу): Ця уразливість виникає, коли недостатньо або неправильно забезпечений контроль доступу, що може дозволити несанкціонованим користувачам отримати доступ до конфіденційних даних чи функціональностей.
- Cryptographic Failures (Криптографічні вразливості): Це стосується неправильного використання криптографічних алгоритмів та недостатньої захищеності криптографічних ключів, що може призвести до витоку конфіденційної інформації.
- Injection (Ін'єкції): Ця уразливість виникає, коли зловмисники вставляють зловмисний код у вхідні дані, такі як SQL-запити, що може призвести до виконання несанкціонованих команд у системі.
- Insecure Design (Ненадійний дизайн): Це включає у себе неправильне проектування додатка, яке створює можливості для різних видів атак.
- Security Misconfiguration (Неправильна конфігурація безпеки): Ця уразливість виникає через неправильну конфігурацію серверів, платформ та інших складових системи, що може призвести до витоку конфіденційної інформації.

Відповідно, OWASP відіграє важливу роль для розробників, надаючи їм доступ до навчальних ресурсів, інструментів та стандартів безпеки. Це сприяє навчанню та розвитку їх навичок у галузі кібербезпеки, забезпечуючи зростання свідомості про потенційні загрози.

Щодо системи децентралізованих додатків (DApps), то - це програмні додатки, які працюють на децентралізованих платформах, таких як блокчейн. Основна відмінність від централізованих додатків полягає у тому, що у децентралізованих системах відсутня централізована влада, що контролює додаток. Замість цього, управління, прийняття рішень і обробка транзакцій здійснюються за допомогою розподіленої мережі, яка складається з вузлів (комп'ютерів), які працюють разом.

Для систем децентралізованих додатків (DApps) також важливо створювати аналогічні вимоги безпеки, як для OWASP. Навіть якщо системи децентралізованих додатків відповідно до своєї архітектури мають певний рівень безпеки, вони не є неуразливими до загроз безпеки. Використання аналогічних вимог безпеки допоможе ідентифікувати та захистити систему від різних видів атак. Децентралізовані додатки можуть використовувати цифрові активи (токени або криптовалюти) або обробляти цінні дані.

Аналогічні вимоги безпеки для системи децентралізованих додатків на базі Ethereum:

- Недоліки авторизації на рівні об'єктів **(Object Level Authorization Vulnerabilities**). Система повинна активно перевіряти авторизацію для доступу до конкретних об'єктів (смарт-контрактів, ресурсів) та забезпечувати обмежений доступ лише для авторизованих користувачів.
- Порушення аутентифікації (**Authentication Failures**). Використання міцних методів аутентифікації та захисту ключів доступу, щоб уникнути компрометації аккаунтів і недозволених дій у мережі Ethereum.
- Недоліки авторизації на рівні властивостей об'єктів (**Object Property Level Authorization Issues**). Врахування обмежень доступу до конкретних властивостей об'єктів або даних в смарт-контрактах та інших компонентах системи.
- Недоліки управління ресурсами (**Resource Management Failures**). Захист від неправомірного використання ресурсів мережі Ethereum, зокрема обмеження обчислювальних та мережевих ресурсів, щоб запобігти атакам з вичерпанням ресурсів (DoS).
- Порушення авторизації на рівні функцій **(Function Level**

Authorization Issues). Забезпечення доступу до функцій та операцій в смарт-контрактах лише для користувачів з необхідними дозволами.

- Порушення нормативного керування ресурсами (**Insecure Business Logic**) - Захист від недопустимого використання функціональності смарт-контрактів та інших компонентів системи для несанкціонованих цілей.
- Порушення аутентичності **(Security Misconfiguration).** Налаштування системи децентралізованих додатків на базі Ethereum з урахуванням найкращих практик безпеки та уникнення конфігураційних помилок.
- Виявлення та експлуатація вразливостей **(Vulnerability Detection and Exploitation)**. Постійний аудит безпеки смарт-контрактів та інших компонентів системи для виявлення та усунення вразливостей перед їх експлуатацією зловмисниками.
- Вивчення та обробка запитів зі сторони клієнта (**Client-Side Request Handling**). Захист від атак типу SSRF та інших атак, які використовують некоректно оброблені клієнтські запити для зламу системи.
- Недостатня захищеність від атак на консенсус **(Insufficient Consensus Protection).** Захист від атак на протокол консенсусу, таких як "51% атака" або атаки типу "selfish mining", забезпечуючи надійність і стабільність мережі Ethereum.