УДК 004.056
Пономаренко Є.О.[1], Неласа Г.В. [2]
[1] асп. НУ «Запорізька політехніка»
[2] канд. техн. наук, доц. НУ «Запорізька політехніка»

ТЕСТУВАННЯ БЕЗПЕКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Тестування безпеки програмного забезпечення – це проактивний підхід до захисту цифрових активів. Це передбачає систематичну оцінку програмного забезпечення для виявлення та усунення слабких місць безпеки. Проводячи всебічне тестування безпеки, ви можете мінімізувати ризик витоку даних, захистити конфіденційну інформацію, підтримувати цілісність систем.

Хоча окремі тести та інструменти можуть бути досить ефективними, коли справа доходить до пошуку дефектів безпеки, жоден із них не є настільки ефективним окремо, як усі разом [1]. Тому немає сенсу покладатися на один інструмент для тестування.

Статичний аналіз включає інструменти для перевірок на наявність вразливостей, порушень стандартів безпечного кодування та загальних проблем, які можуть мати певний вплив на безпеку чи конфіденційність. Динамічний аналіз коду може включати виконання модульних тестів для перевірки того, що компоненти поводяться належним чином. Наскільки це можливо, негативні випадки використання, виявлені в процесі проектування та моделювання загроз, повинні бути включені в модульні тести, щоб будь-які проблеми можна було виявити та вирішити на ранніх стадіях розробки [2]. Платформа Rangeforce (https://www.rangeforce.com/) дозволяє отримати навички за цим напрямом тестування, як і за багатьма іншими напрямами в сфері кібербезпеки.

Процес тестування безпеки програмного забезпечення включає кілька етапів, кожен з яких спрямований на виявлення та пом'якшення потенційних вразливостей.

Першим кроком у тестуванні безпеки програмного забезпечення є чітке визначення вимог до безпеки програми. Це передбачає розуміння потенційних загроз і ризиків, пов'язаних із вашою програмою, визначення конфіденційних даних, які потрібно захистити, а також визначення елементів керування безпекою та заходів, які необхідно застосувати.

Оцінка вразливості передбачає сканування програмного забезпечення на відомі вразливості за допомогою автоматизованих інструментів або ручних методів. Це може включати тестування на загальні вразливості, такі як впровадження SQL, міжсайтовий сценарій (XSS), підробка міжсайтового запиту (CSRF) та інші[3]. Мета полягає в тому, щоб визначити потенційні

слабкі місця, якими можуть скористатися зловмисники. Проведення тестування на проникнення передбачає імітацію реальних атак на програмний додаток для виявлення вразливостей, які можуть не бути виявлені автоматизованими сканерами вразливостей. Тестери проникнення використовують різні методи та інструменти для використання вразливостей і отримання несанкціонованого доступу до програми, допомагаючи виявити потенційні прогалини в безпеці, які необхідно усунути.

Перегляд вихідного коду та конфігурації програмного забезпечення є ще одним важливим кроком у тестуванні безпеки програмного забезпечення[4]. Це передбачає аналіз кодової бази та налаштувань конфігурації для виявлення потенційних недоліків безпеки, таких як незахищені методи кодування, неналежний контроль доступу та неправильні конфігурації. Тестування того, як програма обробляє конфіденційні дані, має вирішальне значення для забезпечення приватності та конфіденційності даних. Це включає тестування шифрування даних, безпечного зберігання даних і протоколів передачі даних. Це також передбачає тестування здатності програми обробляти перевірку вхідних даних, кодування вихідних даних і санітарну обробку даних, щоб запобігти витоку даних.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Smith, M., & Andrews, J. The Art of Software Security Assessment: Identifying and Preventing Software Vulnerabilities. (2016).

2. Testing software security[Електронний ресурс] – Режим доступу: https://portal.rangeforce.com/courses

3. Whittaker, J. A. How to Break Software Security: Effective Techniques for Security Testing. (2019).

4. Secure Coding Practices – Quick Reference Guide. (2021)[Електронний ресурс] – Режим доступу: https://wiki.sei.cmu.edu/confluence/display/seccode/Secure+Coding+Practices+-+Quick+Reference+Guide