SECTION: PEDAGOGICAL SCIENCES.

SEKCJA: NAUK PEDAGOGICZNYCH.

How to cite: Hunko, I., Vakarov, V., & Kosenyuk, H. (2024). Protection against cyber-attacks due to the vulnerability of web applications: methods and techniques of testing. *The Science of Tomorrow: Innovative Approaches and Forecasts.* (pp. 128-133). Futurity Research Publishing. https://futurity-publishing.com/the-science-of-tomorrow-innovative-approaches-and-forecasts-archive/

Protection against cyber-attacks due to the vulnerability of web applications: methods and techniques of testing

Ігор Гунько [1], Віктор Вакаров [2], Григорій Косенюк [3]

[1] Бакалавр «комп'ютерні науки» (НТУУ КПІ імені Ігоря Сікорського, м. Київ, Україна), спеціаліст «облік і аудит» (Національна академія статистики, обліку та аудиту, м. Київ, Україна), twinigor@gmail.com, http://orcid.org/0009-0001-7704-6352,

[2]студент 1 курсу, кафедра систем автоматизованого проектування, спеціальність 122 (проектування і програмування інтелектуальних систем та пристроїв), Інститут комп'ютерних наук та інформаційних технологій, Національний університет "Львівська політехніка", м. Львів, Україна, vik.vakarov.home@gmail.com

[3]кандидат технічних наук, доцент, Черкаський національний університет імені Богдана Хмельницького, м. Черкаси, Україна, grig@ukr.net, https://orcid.org/0000-0003-2103-3904

Accepted: April 6, 2024 | **Published**: April 30, 2024 | **Language:** Ukrainian

Abstract: The purpose of the study is to analyze the issue of protection against cyberattacks due to the vulnerability of web applications in the context of testing methods and techniques. In the course of the research, scientific literature was used in the thematic direction, methods of analysis and synthesis, methods of scientific abstraction, methods of monitoring. Web applications have long been used to provide services and sell goods to make them easier for customers to access.

Understanding the basics of testing web applications for cyber threats will help make them more secure, which will affect the protection of users' personal data.
Keywords: penetration testing, application of a web application firewall, code audit for vulnerabilities, ethical hacking techniques, automated vulnerability scanning.

Вступ

Цифровий простір ще ніколи не був таким вразливим, як зараз, через розквіт кібершахрайства і поширеність його інструментів для кожного, хто має доступ до глобальної мережі та технології роботи з нею. Оскільки кількість і складність кібератак зростає щороку, організаціям у державному та приватному секторах часто не вистачає ресурсів і досвіду, необхідних для створення комплексного захисту. Однією з найбільших проблем залишаються атаки на вебдодатки, які можуть скомпрометувати дані в системі, порушити транзакції та навіть повністю паралізувати компанію. За даними звіту Positive Technologies (Positive Technologies, 2022), у більшості випадків (98%) кіберзлочинці можуть так чи інакше атакувати користувачів вебдодатків. 91% вивчених вебдодатків піддавалися компрометації даних, а 84% платформ виявили вразливості неавторизованого доступу. Згідно з даними проекту OWASP (присвячений забезпеченню безпеки вебдодатків) (OWASP, 2021), у 2021 році в першу трійку критичних загроз входили: порушення контролю доступу – внаслідок такого порушення користувачі можуть отримати доступ до не призначених для них ресурсів та функцій системи; криптографічні збої – наслідком є те, що хакери можуть легко скомпрометувати дані або влаштувати атаку man-in-the-middle; ін’єкції та скриптинг – хакери можуть отримати доступ до конфіденційних даних або навіть маніпулювати роботою програми, втручатися в транзакції. Все це актуалізує питання дослідження методів та прийомів тестування вразливості вебдодатків до кібератак, оскільки кількість загроз для їх користувачів постійно зростає.

Результати дослідження

Під час використання незахищених вебдодатків необхідно пам’ятати про збільшення ризиків несанкціонованого доступу до особистих даних користувача та корпоративних ресурсів, що веде до репутаційних, а також фінансових втрат, тому важливо забезпечити надійний та ефективний захист цих додатків. Для цього під час роботи додатка використовується вебаплікаційний фаєрвол (WAF) і технологія RASP (Runtime Application Self-Protection) (Seth, 2022). WAF дозволяє запобігти багатьом типам атак, серед яких:

1. Ін’єкції SQL і PHP, які базуються на використанні коду, написаного без урахування питань безпеки.

2. Міжсайтовий скриптинг (XSS), що полягає у впровадженні у видану вебсистемою сторінку шкідливого коду (який буде виконано на пристрої користувача у разі відкриття ним цієї сторінки) та взаємодії цього коду з вебсервером зловмисника.

3. Підбір пароля (брутфорс): суть атаки полягає в послідовному автоматизованому переборі всіх можливих комбінацій символів з метою рано чи пізно знайти правильну.

4. Використання вразливостей «нульового дня» у вебдодатках, а саме вектор кібератак, який використовує невідому або невирішену помилку безпеки в програмному чи апаратному забезпеченні. «Нульовий день» означає, що постачальник програмного забезпечення або пристрою має нуль днів для усунення недоліку, оскільки зловмисники вже можуть використовувати його для доступу до вразливих систем.

5. DDoS-атаки на рівні додатку та атаки на ресурсомісткі сторінки: з їх допомогою хакери намагаються викликати недоступність протоколів інтернет-служб через їх перевантаження. Цілями такої атаки можуть бути як сервери загалом, так й інші мережеві компоненти.

Технологія RASP виявляє і блокує атаки на додатки в реальному часі за допомогою додавання функцій захисту в середовище виконання вебдодатків, що дає додаткову можливість «самозахисту» кінцевого сервісу компанії («self-protection»).

RASP можна інтегрувати як фреймворк або модуль, який працює разом із програмними кодами, бібліотеками та системними викликами. Технологія також може бути реалізована як віртуалізація. RASP схожа на інтерактивне тестування безпеки додатків (IAST), ключова її відмінність полягає в тому, що IAST зосереджено на виявленні вразливостей у програмах, а RASP – на захисті від атак на кібербезпеку, які можуть скористатися перевагами цих вразливостей або інших векторів атак.

Технологію RASP можна розгортати двома різними способами: у режимі моніторингу або захисту. У режимі моніторингу RASP повідомляє про атаки на вебдодатки, але не блокує жодні з них. У режимі захисту RASP повідомляє та блокує ці атаки.

Перевірка і тестування вебдодатків є невід'ємною частиною кібербезпеки. Сучасні вебдодатки використовують багаторівневу архітектуру, де реалізація розподіляється по різних шарах і запускається на різних машинах. З цієї причини, щоб перевірити загальну поведінку вебдодатків, потрібні наскрізні методи тестування (Ящук та ін., 2022). Основні методи та прийоми такого тестування наведено в Таблиці 1.

Таблиця 1

Зв'язок між вибраними для огляду прийомами й методами тестування вебдодатків та їх суттю

№	Назва методу	Суть методу
1	Методика етичного хакінгу	У рамках тесту на проникнення експерт може взяти на себе роль хакера та спробувати виявити вразливі місця в програмі. Такі перевірки можуть виконуватися методами білого ящика (хакер знає внутрішні параметри системи), сірого ящика (відомі лише деякі параметри) або чорного ящика (всі параметри невідомі)
2	Пенетраційне тестування	Практика імітації атак на систему задля отримання доступу до конфіденційних даних з метою визначення рівня безпеки системи. Ці атаки здійснюються як усередині, так і зовні системи, вони

		допомагають виявити вразливі місця в ній і розкрити експлойти, які можуть фактично скомпрометувати систему. Це важлива перевірка працездатності системи, яка інформує тестувальників про те, чи потрібні виправлення та заходи безпеки
3	Аудит коду на вразливості	Процес аналізу вихідного коду або перевірки додатку під час роботи з метою виявлення вразливостей безпеки, невідповідності ліцензій та інших програмних проблем. Головною перевагою аудиту безпеки коду є виявлення проблем до випуску програмного забезпечення, щоб жоден користувач не постраждав
4	Автоматизоване сканування вразливостей	Здійснюється за допомогою автоматизованих інструментів сканування вразливостей для виявлення потенційних ризиків і напрямів атак у мережах, апаратному забезпеченні, програмному забезпеченні та системах

Джерело: створено на основі (Synopsys, n. d.; Snyk, 2022)

Два найпоширеніші типи аудиту коду на вразливості – це статичний і динамічний аудити коду. Статичний аудит коду, також відомий як статичний аналіз коду або просто сканування – це перевірка вихідних файлів з метою виявлення вразливостей. Іншими словами, це автоматизована перевірка коду, котра може зосередитися на низці різних аспектів, як-от стандарти коду, форматування коду або вразливість коду. Найпопулярнішим типом аудиту безпеки коду є статичний аналіз, що зосереджений на вразливості коду.

Існує ряд доступних інструментів тестування, які можна використовувати для аудиту безпеки коду, зокрема:

– статичне тестування безпеки програми (SAST): аналізує вихідний код для виявлення проблем;

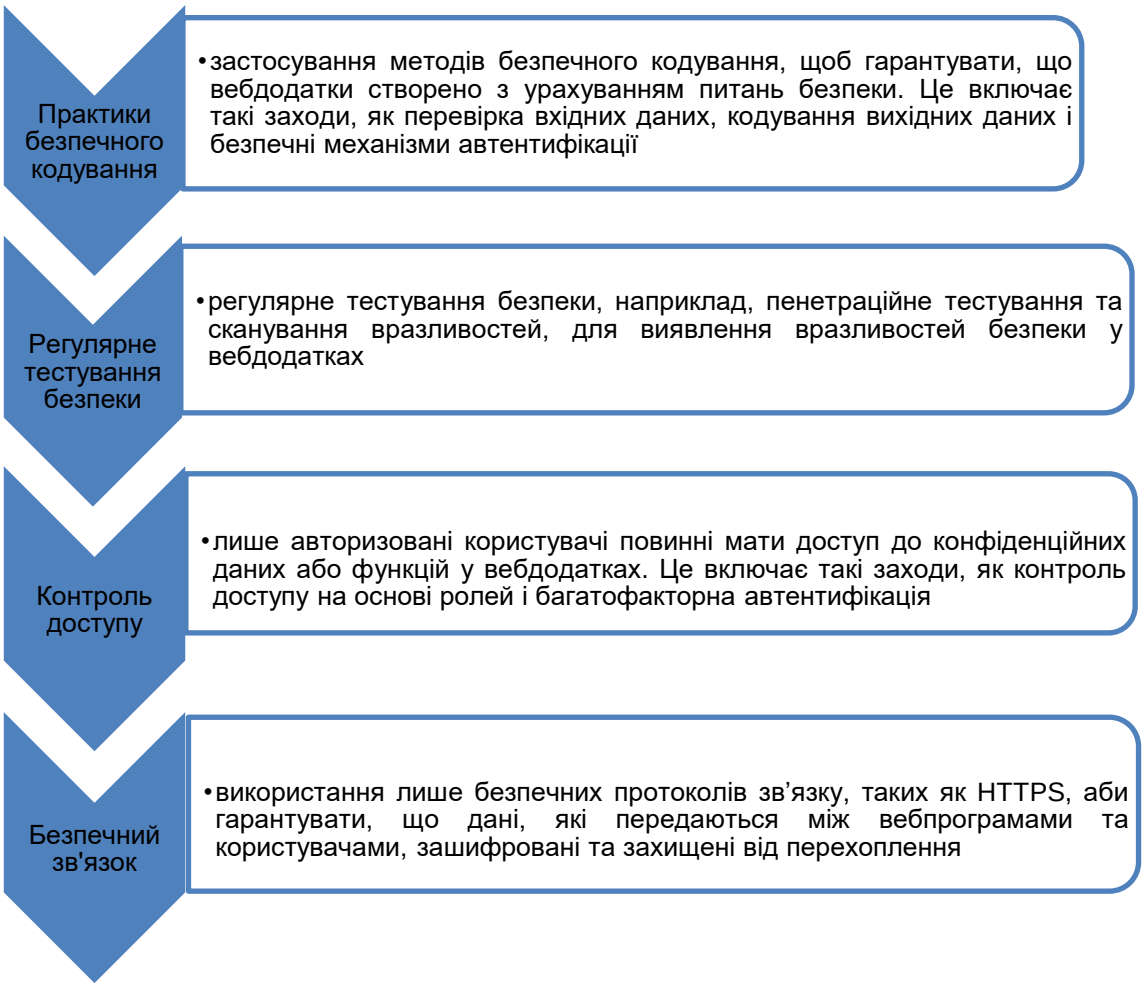
– динамічне тестування безпеки програми (DAST): запускає додаток та оцінює його безпеку за допомогою таких методів, як пенетраційне тестування;

– аналіз складу програмного забезпечення (SCA): шукає вразливості в прямих і непрямих залежностях відкритого коду.

Розроблення переконливих та реалістичних стратегій захисту вебдодатків від кібератак стало ефективним засобом протидії зловмисникам та «чорним капелюхам» (хакерам, що працюють у нелегальному полі) [Рисунок 1].

Рисунок 1

Стратегія захисту вебдодатків їх розробниками від кібератак



Джерело: власна розробка авторів

Користувачам вебдодатків потрібно постійно пам'ятати про небезпеку кібератак та дотримуватись простих правил: уникати переходу за підозрілими посиланнями або завантаження контенту з невідомих джерел.

Висновки

Отже, використання методів і прийомів тестування вебдодатків та їх коду задля захисту від кібератак має потенціал у створенні стратегії захисту цих додатків їх розробниками, що допомагає протидіяти зловмисникам та хакерам. Хоча натепер загальний рівень захищеності вебдодатків залишається вкрай низьким, але навіть існуючих засобів захисту, як правило, вистачає за умови, що додаток постійно отримує останні виправлення безпеки та оновлення для усунення відомих вразливостей. Важливим засобом збереження конфіденційності особистої інформації користувача є його навички «інформаційної гігієни» в Інтернеті, що вимагає уважності та критичного мислення під час користування додатком.

Література

Ящук, В. І., Репетило, Т. М., & Кушнірук, М. (2022). Методи тестування на проникнення веб-додатків. https://sci.ldubgd.edu.ua/jspui/handle/123456789/11410

OWASP. (2021). *OWASP Top Ten | OWASP Foundation*. OWASP Foundation, the Open Source Foundation for Application Security | OWASP Foundation. https://owasp.org/www-project-top-ten/

Positive Technologies. (2022). *Threats and vulnerabilities in web applications 2020–2021*. ptsecurity.com. https://www.ptsecurity.com/ww-en/analytics/web-vulnerabilities-2020-2021/

Seth, A. (2022). *Comparing effectiveness and efficiency of interactive application security testing (iast) and runtime application self-protection (rasp) tools*. North Carolina State University. https://www.proquest.com/openview/a39d6b444f572f9478135e22dd0b32f3/1?pq-origsite=gscholar&cbl=18750&diss=y

Snyk. (2022). *Code security auditing 101 | Snyk*. https://snyk.io/learn/code-security-audit/

Synopsys. (n. d.). *What Is Web Application Penetration Testing and How Does It Work? | Synopsys*. Synopsys | EDA Tools, Semiconductor IP and Application Security Solutions. https://www.synopsys.com/glossary/what-is-web-application-penetration-testing.html