УДК 004.056.5

Опанасенко Є.Р., магістрантка
Єфіменко А.А., к.т.н., доцент
Державний університет «Житомирська політехніка»

ОЦІНКА ВПЛИВУ АРХІТЕКТУРИ ВЕБ-ДОДАТКУ НА ЧАСТОТУ ХИБНОПОЗИТИВНИХ СПРАЦЮВАНЬ ПРИ СКАНУВАННІ ВРАЗЛИВОСТЕЙ

Сучасні веб-додатки дедалі частіше будуються за принципами розподілених систем: Single Page Applications (SPA), мікрофронтенди, мікросервіси та serverless-архітектури вже стали стандартом для забезпечення масштабованості, гнучкості та швидкої розробки. Водночас інструменти динамічного аналізу безпеки (DAST), такі як OWASP ZAP, Burp Suite, Invicti чи Detectify, залишаються ключовим елементом автоматизованого тестування безпеки. Однак у складних архітектурах традиційні DAST-сканери генерують значну кількість хибнопозитивних (false positive) спрацювань, що призводить до втоми від сповіщень, перевантаження команд безпеки та зниження довіри до автоматизованого сканування[2,3].

Аналіз сучасних джерел та практики використання DAST-інструментів показує, що найбільша кількість хибнопозитивів виникає в архітектурах з високим рівнем динаміки клієнтської частини та розподіленості бекенду.

Найвищий рівень хибнопозитивів фіксується в мікрофронтендах. Фрагментована структура, використання різних фреймворків у межах одного додатка, міждоменна взаємодія та динамічне завантаження компонентів призводять до того, що сканери не можуть правильно побудувати контекст застосування. Це спричиняє дублювання тестів, помилкове трактування безпечного JavaScript-коду як XSS чи IDOR, а також численні хибні спрацювання на контроль доступу[3].

Дуже близьким за проблематикою є SPA (React, Angular, Vue, Svelte тощо). Клієнтська маршрутизація, віртуальний DOM, умовне відображення компонентів та те, що URL не змінюється при переходах між сторінками, — усе це сильно ускладнює нормальне сканування додатка. Традиційні краулери (наприклад, у OWASP ZAP чи Burp) часто зациклюються на станах або пропускають значну частину функціоналу, що генерує повторні та хибні тести на ін'єкції, CSRF чи порушення контролю доступу[2,3].

Третє місце посідають мікросервіси + serverless (FaaS, Lambda, Cloudflare Workers). Тут проблеми пов'язані з великою кількістю

динамічних ендпоінтів, подієво-орієнтованою логікою, стартом функцій «з нуля» та складною міжсервісною автентифікацією (JWT, OAuth). Сканер не здатен підтримувати повний контекст сесії між сервісами, що призводить до помилкових спрацювань на помилки конфігурації, обхід авторизації та API-аномаліях[2].

Дещо менш проблематичними є GraphQL + API-heavy архітектури, де сканери погано моделюють складні схеми запитів, що також підвищує рівень хибних спрацювань. Натомість класичні моноліти та JAMstack/статичні сайти демонструють найнижчий рівень хибних спрацювань завдяки простішій структурі URL та меншій поверхні атаки. SSR/SSG-рішення (Next.js, Nuxt, Remix) займають проміжну позицію — вони кращі за чисті SPA, але все одно генерують більше шуму, ніж моноліти.

Основні причини високого рівня хибнопозитивів: відсутність повного розуміння JavaScript-логіки, проблеми з підтримкою стану сесії, вибухи станів у розподілених системах та недостатня адаптація традиційних DAST-інструментів до сучасних патернів розробки[1].

Перехід до сучасних архітектур суттєво ускладнює ефективність традиційного DAST-сканування через зростання хибнопозитивних результатів. Для зменшення шуму рекомендується використовувати AI-based DAST-інструменти (Invicti, Escape, Checkmarx One), надавати OpenAPI/GraphQL-специфікації, застосовувати аутентифіковане сканування та комбінувати DAST з IAST/RASP. Наразі ключовим напрямком розвитку є перехід до валідації на основі доказів що до вразливостей та інтеграція DAST у CI/CD-процеси з автоматичною фільтрацією false positives[1,2,3].

Список використаних джерел

1. O. Revniuk, N. Zagorodna, R. Kozak, B. Yavorskyy DEVELOPMENT OF AN INFORMATION SYSTEM FOR THE QUANTITATIVE ASSESSMENT OF WEB APPLICATION SECURITY BASED ON THE OWASP ASVS STANDARD 2025, № 2 (118), с. 57- 64.

2. SecureSlate Team. The 7 Best DAST Solutions for 2026 — Ranked by Speed and Accuracy [Електронний ресурс] – Режим доступу: https://getsecureslate.com/blog/the-7-best-dast-solutions-for-2026-ranked-by-speed-and-accuracy.

Invicti. False Positives in Web Application Security – Facing the Challenge [Електронний ресурс] – Режим доступу: https://www.invicti.com/white-papers/false-positives-in-application-security-whitepaper.