УДК 004.738.5

Пятаков О.О., здобувач
Колощук М.С., ст. викладач
Державний університет «Житомирська політехніка»

ПОРІВНЯЛЬНИЙ АНАЛІЗ МОЖЛИВОСТЕЙ GITHUB CODEQL ТА OWASP ZAP В УМОВАХ DEVSECOPS-КОНВЕЄРА

Сучасні веб-додатки постійно зіштовхуються зі зростаючою кількістю кіберзагроз. Ускладнення архітектури програмного забезпечення вимагає переходу до концепції DevSecOps та інтеграції перевірок безпеки на ранніх етапах життєвого циклу розробки (Shift-Left). Існуючі методи тестування, такі як статичний аналіз (SAST) та динамічне сканування (DAST), мають свої переваги та суттєві обмеження при ізольованому використанні: SAST часто генерує хибно-позитивні спрацювання і не аналізує конфігурацію сервера, тоді як DAST не має доступу до вихідного коду і пропускає приховані логічні вразливості.

Метою дослідження є практичний порівняльний аналіз можливостей інструментів GitHub CodeQL (SAST) та OWASP ZAP (DAST), а також дослідження ефективності їх об’єднання в єдиний аналітичний гібридний конвеєр.

В рамках дослідження було використано еталонний вразливий веб-додаток OWASP Juice Shop. У хмарному середовищі GitHub Actions було спроєктовано та налаштовано автоматизований CI/CD конвеєр, який включав послідовний запуск інструментів безпеки під час збірки та розгортання коду у Docker-контейнері.

Після успішного налаштування пайплайну (рис. 1), було проведено сканування додатка. Для аналізу та консолідації результатів було розроблено авторський програмний модуль мовою Python. Даний скрипт виконує роль агрегатора: він отримує звіти від SAST та DAST, порівнює їх та усуває дублікати, формуючи зведений звіт.

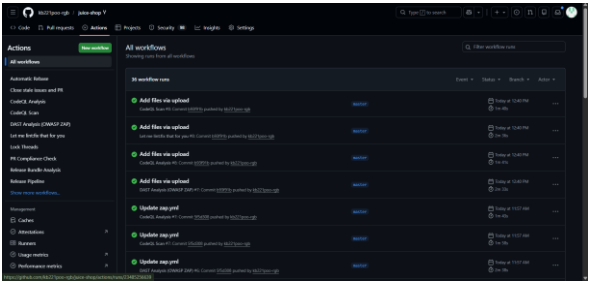

Рисунок 1 – Візуалізація успішного виконання гібридного конвеєра CI/CD

Результати агрегації (рис. 2) наочно демонструють спеціалізацію кожного з інструментів. Інструмент GitHub CodeQL успішно виявив вразливості на рівні коду, такі як Server-Side Request Forgery (SSRF) та Cross-Site Scripting (XSS). Водночас OWASP ZAP ідентифікував проблеми конфігурації веб-сервера: відсутність Content Security Policy (CSP) та Cross-Domain Misconfiguration, які фізично не могли бути виявлені статичним аналізатором.

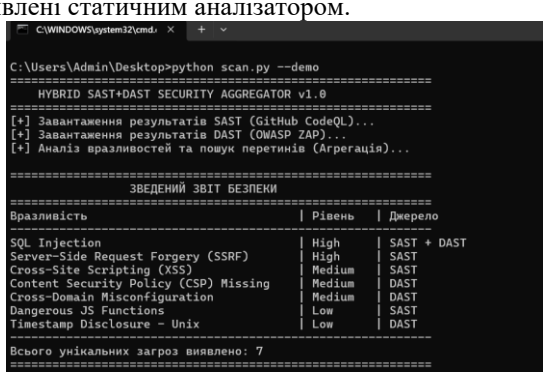

Рисунок 2 – Результат роботи програмного модуля агрегації вразливостей (SAST+DAST)

Найбільш критична уразливість — SQL Injection — була успішно виявлена обома системами, що підтверджується зведеним звітом програмного модуля (рис. 2).

Отже, проведене практичне дослідження підтвердило, що використання лише одного класу сканерів є недостатнім для комплексного захисту. Побудова гібридного CI/CD конвеєра з інтеграцією GitHub CodeQL та OWASP ZAP дозволяє компенсувати недоліки окремих методів та забезпечити максимальне покриття вразливостей згідно з класифікацією OWASP Top 10.

Список використаних джерел

1. OWASP Top 10:2021. URL: https://owasp.org/Top10/
2. About CodeQL code scanning in your CI/CD system. URL: https://docs.github.com/en/code-security/code-scanning
3. OWASP ZAP - Baseline Scan. URL: https://www.zaproxy.org/docs/docker/baseline-scan/