Вісник Тернопільського національного технічного університету
https://doi.org/10.33108/visnyk_tntu
Scientific Journal of the Ternopil National Technical University
2025, № 2 (118) https://doi.org/10.33108/visnyk_tntu2025.02
ISSN 2522-4433. Web: ***visnyk.tntu.edu.ua***

UDC 004.056.5

DEVELOPMENT OF AN INFORMATION SYSTEM FOR THE QUANTITATIVE ASSESSMENT OF WEB APPLICATION SECURITY BASED ON THE OWASP ASVS STANDARD

Oleksandr Revniuk; Nataliya Zagorodna; Ruslan Kozak; Bohdan Yavorskyy

Ternopil Ivan Puluj National Technical University,
Ternopil, Ukraine

Abstract. *The design of an information system for assessing the security of web applications based on an original methodology developed by the authors is presented in this paper. The proposed security assessment methodology is based on the requirements of the OWASP Application Security Verification Standard (ASVS) and adapted to various application architectures and functionalities by selecting a set of relevant requirements and determining their impact on the overall evaluation. The quantitative assessment of requirements is calculated using a system of developed criteria and an evaluation algorithm that incorporates weight coefficients of importance assigned by experts. The assessment is carried out by multiple experts to minimize subjectivity in judgments. The aggregation of expert judgments is performed within a fuzzy logic subsystem. The article describes all stages of the assessment process automation – from collecting input data to calculating the integrated security score, taking into account the weight coefficients. The information system supports a modular architecture, personalized project workflows, and result visualization, enabling its application in information security audits.*

Key words: *OWASP ASVS, web application, expert evaluation, information system, security, information security audit.*

https://doi.org/10.33108/visnyk_tntu2025.02.056 *Received 18.02.2025*

1. INTRODUCTION

In the context of the rapid development of information technologies, web applications serve as key components of the digital infrastructure, supporting commercial, administrative, and social processes [1]. Their widespread implementation underscores the importance of information security issues, particularly in terms of protecting sensitive data and ensuring resilience to external threats.

The complexity of architectural solutions, intensive client-server interactions, and the dynamic nature of business logic contribute to an increasing number of potential attack vectors. This, in turn, necessitates a systematic approach to evaluating the security level of web applications. A variety of technical tools and industry standards exist to enable the identification of vulnerabilities in web applications and to assess their criticality [2]. Despite the availability of professional guidelines and testing tools, several challenges remain related to the formalization of results, the unification of approaches, and the objectivity of conclusions. The authors of [3] proposed an assessment methodology capable to take into account the specific characteristics of the analyzed object and providing a quantitative interpretation of security requirements. This study presents a practical approach to the implementation of an information system for the formalized assessment of web application security using the OWASP ASVS standard. This approach enhances the accuracy, structure, and transparency of the analysis, while also enabling the automation of technical decision-making processes.

 Corresponding author: Nataliya Zagorodna; e-mail: kaf_kb@tntu.edu.ua

2. LITERATURE REVIEW

Since the beginning of the 21st century, web applications have undergone a radical evolution. Implementation of cloud-oriented infrastructures (AWS, GCP, Azure), the deployment of microservices and serverless architectures, and the widespread adoption of containerization technologies (Docker, Kubernetes) and CI/CD practices have transformed web systems into highly dynamic, distributed complexes. This has significantly increased their structural complexity and introduced new, more sophisticated challenges in the field of information security.

The response to these challenges has been the active implementation of automated scanning tools, which provide basic security tests at the initial stages. However, researches show that none of these tools guarantees full coverage of possible vulnerabilities. For example, as noted by Dixitkumar V. and Prajapati [4], automated scanners may produce false positives or miss complex vulnerabilities related to business logic [5–7]. In this regard, the combined use of multiple scanners and additional testing methods is recommended [8].

At the same time, manual penetration testing demonstrates higher effectiveness, particularly in identifying atypical or context-dependent vulnerabilities. However, its results largely depend on the performer's competence. To reduce the influence of the subjective factor, industry standards like OWASP, NIST and SANS have been increasingly implemented. The OWASP Application Security Verification Standard (ASVS) [9] has gained significant popularity, offering a multi-level framework for verifying requirements related to confidentiality, integrity, and availability, and is closely linked to the CWE vulnerability classification [10].

Despite its structured and practical nature, ASVS does not provide a formal basis for quantitatively assessing the security level of web resources [11]. The lack of clear criteria for verifying requirements does not allow objective comparison of analysis results and complicates the process of making decisions. This shortcoming has been addressed by developing an approach to formalizing the scope of verification, which reduces the impact of subjective factors during security assessment [11].

The market offers various tools that can partially assist in verifying specific ASVS requirements, but none of them ensures a comprehensive approach or provides a quantitative assessment of web application security. For example, OWASP ZAP – a free DAST scanner – allows scripting to automate certain ASVS checks; its advantages include free access and easy integration into CI/CD, but it covers only a small portion of ASVS. Commercial scanning tools have also expanded ASVS support: Invicti (Netsparker) detects hundreds of vulnerabilities, Checkmarx offers a dedicated OWASP ASVS preset for static analysis, and SonarQube generates an ASVS report based on the identified CWEs. However, these products focus on code analysis and only cover those ASVS requirements that can be detected automatically, without providing any quantitative assessment [12–14].

There are also projects that automate partial ASVS verification: for instance, OWASP ASVS Security Evaluation Templates (Nuclei) offer templates for scanning specific ASVS items, but they require refinement and are not a full substitute for manual auditing. Thus, existing solutions provide only partial ASVS automation, which underscores the relevance of developing an information system, as described in the next section.

3. DESING AND IMPLEMENTATION OF INFORMATION SYSTEM

3.1. Development of the Information System

To formalize approaches to evaluating the security level of web applications, a methodology was developed [3], based on the standardized requirements of the OWASP

Application Security Verification Standard (ASVS). The proposed methodology can be adapted to the functionality, architecture and data criticality level of a specific web application project. ASVS is widely used in the information security industry to assess the reliability of the architecture, implementation, and configuration of web applications. However, despite its universal nature, full coverage of all ASVS requirements (more than 260) is often excessive: some requirements are duplicated, and others can be applied only to certain types of architectures and web application functionalities. Such an approach is impractical for implementation in typical web development environments, particularly in small and medium-sized businesses or under resource constraints.

In this context, the importance of a selective approach was substantiated, whereby only those ASVS requirements that are essential for ensuring basic and advanced levels of security were selected from the overall list. The selection was carried out by the authors based on an analysis of each section of the standard, consultations with cybersecurity experts, and consideration of typical web application usage scenarios, architectural features, access to information about the software development life cycle, and the most likely attack vectors. As a result, the final model includes only those requirements that represent critical security aspects relevant to most web platforms.

To improve the objectivity of evaluating each selected requirement, a system of criteria and methods for their assessment were developed. The quantitative assessment of each requirement is calculated as a weighted sum of the scores for the criteria, allowing the degree of compliance of the software product with the corresponding requirement to be determined. The weight coefficient of each criterion was determined through a evaluation by three independent cybersecurity experts. To avoid subjectivity, take into account expert uncertainty in assessments, and align expert opinions, fuzzy logic methods were applied.

The primary goal of the developed information system is to automate the process of quantitatively assessing the security of a web application in order to increase the efficiency of the tester's work. The architecture of the information system consists of two parts: the adaptive evaluation methodology and the algorithm for assessing the security of web applications (Figure 1).

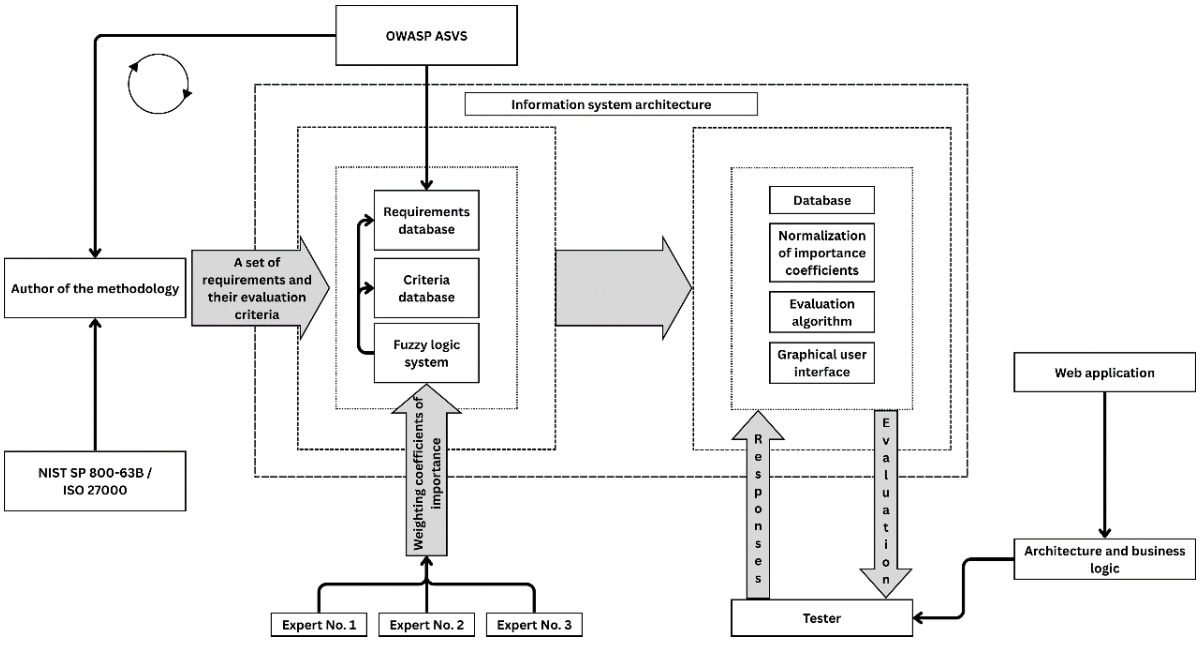

Figure 1. Architecture of the Information System

The input data for the system include the OWASP ASVS standard requirements selected by the authors of the methodology, as well as a system of developed criteria for verifying the fulfillment of each requirement and evaluating them based on cybersecurity industry standards. The importance of each criterion and requirement was assessed by three independent experts. Expert score aggregation and consideration of uncertainty are handled in the fuzzy logic subsystem. The aggregated weight coefficients of the criteria and requirements are stored in a default database, enabling reuse across various testing projects and allowing modifications when necessary. Additionally, all weight coefficients undergo a normalization procedure. The tester interacts with a web application that includes the web interface and evaluation algorithm. Considering the context of the application being evaluated, the tester responds to a predefined set of questions, and the algorithm transforms these responses into quantitative scores, which are stored in the information system's database.

As shown in Figure 1, one of the key stages in the information system design is organizing and creating a data structure capable to effectively store, process and utilize assessment results. In this context, the database plays a crucial role, providing interaction between the analytical model, user interface, and functional modules of the web application. The authors developed a relational database model that reflects the data structure and operational processes of user interaction. The model is based on the concept of structured, hierarchical storage of security requirements, as well as mechanisms for recording evaluation results and adapting the set of requirements according to the functional characteristics of the analyzed object.

The logical structure of the database forms three main functional subsystems: user and project management, storage of the regulatory base of security requirements, and recording of evaluation results at various levels of detail. At the user level, it is possible to create one or more projects for assessing different web applications. User and project tables, which contain authentication data, are used to personalize system operation.

The database structure is a three-level hierarchy, where sections are grouped semantically related requirements, each of which is further detailed through a set of evaluation criteria. Both requirements and criteria have attributes for base and normalized weights, which makes it possible to apply a formal weighted computation during evaluation. This is especially important in the context of adaptive assessment, where not all elements of the methodology are equally relevant to every evaluated project.

To ensure system adaptability, a preliminary questionnaire mechanism has been implemented, with its results stored in question and answer tables. User responses automatically determine the set of sections, requirements, and criteria relevant to the project. This enables the dynamic adjustment of the selection according to the architectural, functional, and business characteristics of a specific web application, thereby increasing the accuracy and objectivity of the assessment.

During the evaluation process, all results are recorded at four levels: criteria, requirements, sections, and the project as a whole. At each level, results from the lower level are aggregated using the corresponding weight coefficients. The logical structure of the database is illustrated in the diagram in Figure 2; all tables are interconnected via foreign keys, which ensures data consistency

during modification operations. Unique indexes in the result tables prevent duplication of evaluations for the same criterion or requirement within a given project, and cascading constraints ensure proper deletion of related entities.

Figure 2. Database Schema

The effective use of the developed database is ensured through a set of user interaction scenarios with the system. One of the key components of the developed information system is the implementation of user interaction scenarios with the software throughout the full life cycle of web application security assessment. The interaction between the user, the web interface, business logic services, and computational modules is implemented in accordance with the principles of service-oriented architecture (SOA), using patterns typical of the Laravel framework. The corresponding sequence of actions is presented as a sequence diagram shown in Figure 3, which formalizes the message exchange between the actor (user) and the internal components of the system.

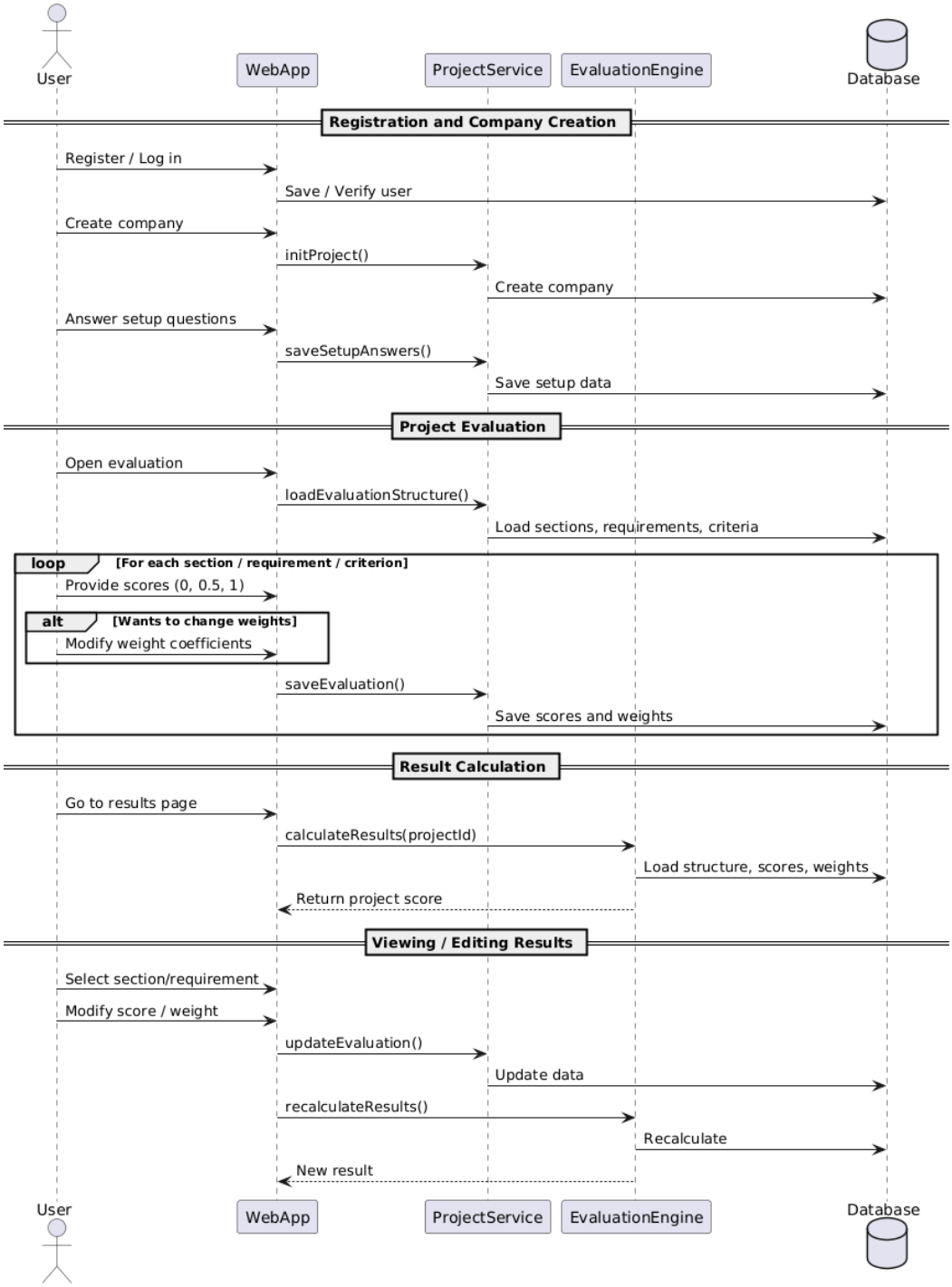

Figure 3. Sequence Diagram of Web Application Assessment

3.2. Technological Stack

For the development of the information system, a technological stack was selected to ensure compliance with current information security requirements and the OWASP ASVS standard. The server side is implemented using the Laravel (PHP) framework, which follows the MVC architectural model and includes integrated tools to counter common types of attacks (CSRF, SQL injections, XSS). It also supports secure authentication with credential hashing using the bcrypt algorithm.

The client side is built with HTML5, CSS3, and Bootstrap 5, ensuring interface responsiveness, while dynamic interaction is implemented using JavaScript in combination with jQuery. Data exchange protection is achieved through CSRF token validation and server response integrity control.

The data storage is based on the MySQL 8.0 DBMS, which provides transaction support, data-at-rest encryption, the use of secure connections (SSL/TLS), and the implementation of a role-based access model in accordance with the principle of least privilege.

The application operates in a cross-platform environment without the need for local deployment and supports the visualization of assessment results using the Chart.js library.

3.3. Application of the Security Assessment Methodology

To apply the information system in practice, the tester used the well-known educational web application «OWASP Juice Shop» [15]. This e-commerce web application is specifically designed for security testing and contains numerous vulnerabilities.

At the initial stage, the user interacts with the system interface for authentication, which is implemented through registration or login operations. Upon successful authentication, the user proceeds to the stage of creating a new assessment project.

During project creation, the tester is prompted to fill out a corresponding form that records the basic information about the project, as shown in Figure 4.

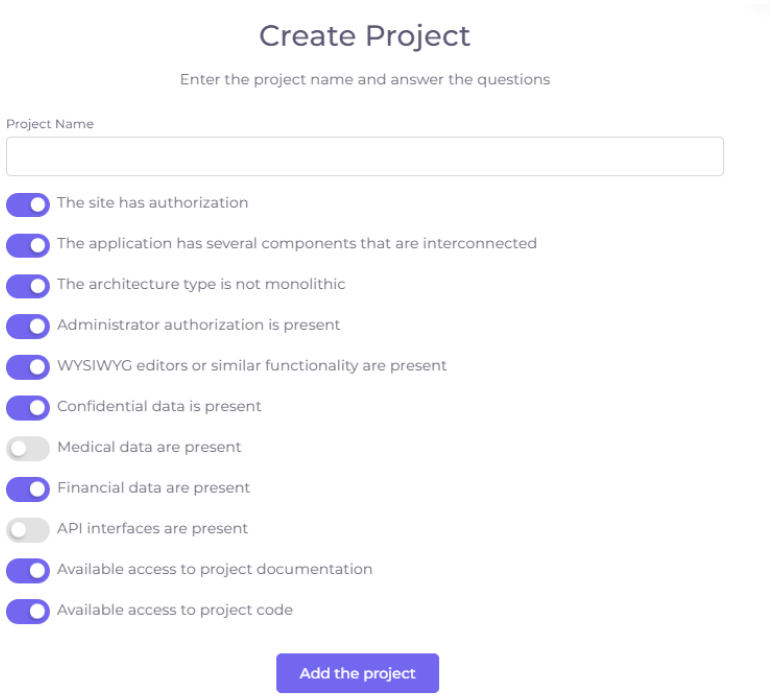

Figure 4. Project Creation Form

To ensure accurate identification of the system type and its operational specifics, a set of control questions is applied. These allow for the identification of the architectural model (monolithic, service-oriented, or microservice), determination of the presence of authentication and authorization mechanisms, user types, API usage, handling of confidential data, and support for dynamic content. For the verification of certain requirement categories, access to the source code and technical documentation is necessary.

This stage serves as the starting point for generating a relevant list of OWASP ASVS standard requirements tailored to the specifics of the evaluated project. The created project is automatically displayed on the control panel, where the user can initiate the assessment process. Initiating the assessment triggers the loading of the methodological structure, which includes relevant sections, requirements, and evaluation criteria.

The assessment process involves step-by-step assignment of scores to each criterion within the corresponding requirements. For each requirement, it is possible to adjust the importance coefficient, which can be modified by the tester if needed. The normalized coefficient value is recalculated immediately after the user makes changes. Each requirement includes its own set of detailed embedded criteria that use a similar coefficient system. Evaluation is carried out using the following scale: «No», «Partially», and «Yes».

Upon completion of the assessment, the system automatically calculates the project's security score, and the result is displayed in the user's web interface. In addition to the initial calculation, the system supports the ability to review and edit the assessment results. Figure 5 presents a graphical representation of the assessment results for each section.

Figure 5. Graphical Assessment Results of the Project by Section

Below, the user can select any section or requirement and change the corresponding score or weighting coefficient. According to the color scheme, sections and requirements are highlighted with colors to facilitate easier navigation of the project. Criteria are highlighted based on the response: red – «No», orange – «Partially», and green – «Yes», respectively. This functionality is illustrated in Figure 6.

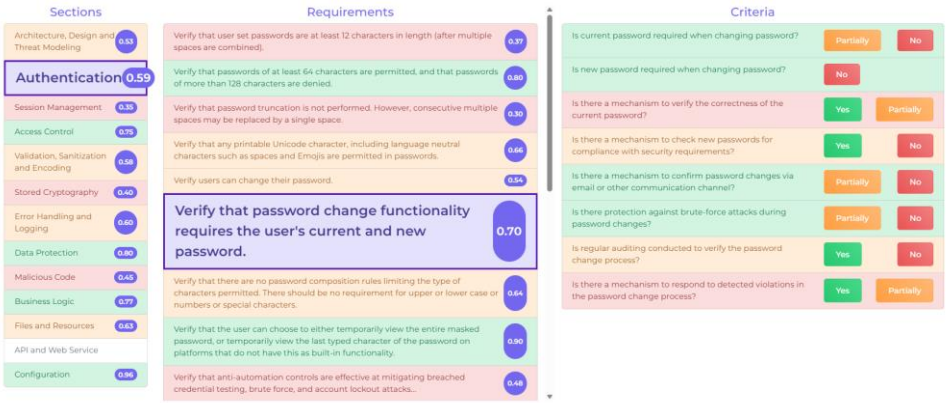

Figure 6. Detailed Project Report

The expert also has the ability to modify the responses for the criteria, which will dynamically update the chart and change the coloring of the sections, requirements, and criteria.

3. CONCLUSIONS

As part of the conducted research, an information system was designed for the quantitative assessment of web application security, implementing an adaptive assessment methodology based on the OWASP ASVS standard requirements. The proposed approach takes into account the technological complexity of web resources, their functionality, and usage context. It gives possibility to eliminate several shortcomings in traditional assessment tools. The implementation of a flexible weighting system and the development of a multi-level structure of criteria and their evaluation procedure help reduce subjectivity, ensuring a high level of accuracy, structure, and transparency in the security analysis process.

The information system, built upon this methodology, supports the full assessment cycle – from collecting input data and forming a relevant set of requirements to computing an integral score with consideration of weight coefficients and visualizing the results. This not only formalizes information security audit processes but also ensures their scalability, transparency, and flexibility. Thanks to its modular architecture, adaptability, and personalized project management features, the system is suitable for integration into security processes at various stages of the software development lifecycle.

The practical value of the developed system lies in its potential use both within independent expert evaluations and as an integral part of security assurance processes during the lifecycle of web application development. The assessment results can be incorporated into decision-making processes concerning technical and organizational security measures.

Future research will focus on validating the accuracy of the methodology under testing conditions, improving the mechanisms for adapting requirements to different types of web application architectures, and expanding the system's functionality, particularly through automating result analysis using machine learning algorithms.

References

1. Shahid J., Hameed M. K., Javed I. T., Qureshi K. N., Ali M. & Crespi N. (2022). A Comparative Study of Web Application Security Parameters: Current Trends and Future Directions. Applied Sciences, 12 (8), 4077. https://doi.org/10.3390/app12084077
2. Derkach, M. V., Khomyshyn, V. G., Gudzenko, V. O. (2023). Web resource security testing based on tools for scanning and detecting vulnerabilities. Scientific News of Dal University. Electronic edition, 25, pp. 1–8. [in Ukrainian].
3. Revniuk O., Zagorodna N. & Ulichev O. (2024) Adaptive Methodology for Computing the Quantitative Security Status Indicator of Web Applications. *Central Ukrainian Scientific Bulletin. Technical Sciences*, *2*(10(41)), 3–10. https://doi.org/10.32515/2664-262X.2024.10(41).2.3-10
4. Yaqoob I., Hussain A. S., Mamoon S., Naseer N., Akram J. & Rehman A. U. R. (2017) Penetration Testing and Vulnerability Assessment. Journal of Network Communications and Emerging Technologies (JNCET), 7 (8).
5. Tadhani J. R., Vekariya V., Sorathiya V., Alshathri S. & El-Shafai W. (2024) Securing web applications against XSS and SQLi attacks using a novel deep learning approach. *Scientific Reports*, 14 (1). https://doi.org/10.1038/s41598-023-48845-4
6. Wen S.-F. & Katt B. (2023) A quantitative security evaluation and analysis model for web applications based on OWASP application security verification standard. *Computers & Security*, *135*, 103532. https://doi.org/10.1016/j.cose.2023.103532
7. Kaźmierak I. (2025) Comparison of the effectiveness of tools for testing the security of web applications. *Journal of Computer Sciences Institute*, 34, 36–43. https://doi.org/10.35784/jcsi.6613
8. Tryhubets B., Tryhubets M. & Zagorodna N. (2024) Analysis of the efficiency of open source and commercial vulnerability scanners for e-commerce web applications. *Scientific Journal of the Ternopil National Technical University*, 116 (4), pp. 23–30. https://doi.org/10.33108/visnyk_tntu2024.04.023
9. *OWASP Application Security Verification Standard (ASVS). OWASP Foundation*. OWASP Foundation, the Open Source Foundation for Application Security | OWASP Foundation. https://owasp.org/www-project-application-security-verification-standard.
10. *CWE – Common Weakness Enumeration*. CWE – Common Weakness Enumeration. https://cwe.mitre.org/

11. Revnyuk, O. A., Zagorodna, N. V. (2024). Methodology of quantitative assessment of security of electronic commerce web application at the operation stage. Scientific Bulletin of Ivano-Frankivsk National Technical University of Oil and Gas. 2(57), pp. 107–119. https://doi.org/10.31471/1993-9965-2024-2(57)-107-119 [in Ukrainian].
12. Putra F. P ., Ubaidi U., Hamzah A., Pramadi W. A. & Nuraini A. (2024). Systematic Literature Review: Security Gap Detection on Websites Using OWASP ZAP. *Brilliance Research of Artificial Intelligence*, 4 (1), pp. 348–355. https://doi.org/10.47709/brilliance.v4i1.4227
13. Seth A., Bhattacharya S., Elder S., Zahan N. & Williams L. (2025) Comparing effectiveness and efficiency of Interactive Application Security Testing (IAST) and Runtime Application Self-Protection (RASP) tools in a large java-based system. *Empirical Software Engineering*, 30 (3). https://doi.org/10.1007/s10664-025-10621-5
14. Mangaoang N. E. F. (2024) Common Vulnerabilities and Exposures Assessment of private higher educational institutions using web application security. Deleted Journal, 20 (5s), pp. 668–676. https://doi.org/10.52783/jes.2288
15. *OWASP Juice Shop. OWASP Foundation*. OWASP Foundation, the Open Source Foundation for Application Security. OWASP Foundation. https://owasp.org/www-project-juice-shop/

УДК 004.056.5

РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ КІЛЬКІСНОГО ОЦІНЮВАННЯ ЗАХИЩЕНОСТІ ВЕБЗАСТОСУНКІВ НА ОСНОВІ СТАНДАРТУ OWASP ASVS

Олександр Ревнюк; Наталія Загородна;
Руслан Козак; Богдан Яворський

Тернопільський національний технічний університет імені Івана Пулюя, Тернопіль, Україна

Резюме. *Спроєктовано інформаційну систему для оцінювання захищеності вебзастосунків на основі авторської методики. Розроблена авторами методика оцінювання безпеки вебдодатків базується на вимогах стандарту OWASP Application Security Verification Standard (ASVS) та адаптується під різні архітектури та функціонал застосунків шляхом відбору множини релевантних вимог і визначення їх впливу на загальну оцінку. Кількісна оцінка вимог обчислюється за допомогою системи розроблених критеріїв та алгоритму їх оцінки з урахуванням експертних оцінок вагових коефіцієнтів важливості. Оцінювання проводиться кількома експертами для уникнення суб'єктивності суджень. Узгодження експертних оцінок проводиться в підсистемі нечіткої логіки. Описано всі етапи автоматизації процесу оцінювання – від збирання вихідних даних до обчислення інтегрального показника захищеності з урахуванням вагових коефіцієнтів. Інформаційна система підтримує модульну архітектуру, персоналізовану роботу з проєктами та візуалізацію результатів, що уможливлює її використання для аудитів інформаційної безпеки. Описано логічну структуру бази даних, яка забезпечує ієрархічне зберігання вимог, критеріїв оцінювання та результатів експертної перевірки. Структура включає підсистеми управління користувачами, проєктами, нормативною базою та результатами, що дозволяє ефективно масштабувати рішення під потреби різних організацій. Розроблено інтерфейс користувача, який забезпечує повноцінний цикл взаємодії з системою – від ініціалізації проєкту до отримання графічного звіту про рівень захищеності. Система реалізована за принципами сервіс-орієнтованої архітектури із застосуванням сучасного стеку технологій, зокрема Laravel, MySQL, Bootstrap і Chart.js. Проведено тестування інформаційної системи на прикладі тестового вебзастосунку OWASP Juice Shop, що підтверджує її ефективність у виявленні критичних проблем безпеки та формалізованому поданні результатів.*

Ключові слова: *OWASP ASVS, вебзастосунок, експертна оцінка, інформаційна система, безпека, аудит інформаційної безпеки.*

https://doi.org/10.33108/visnyk_tntu2025.02.056 *Отримано 18.02.2025*