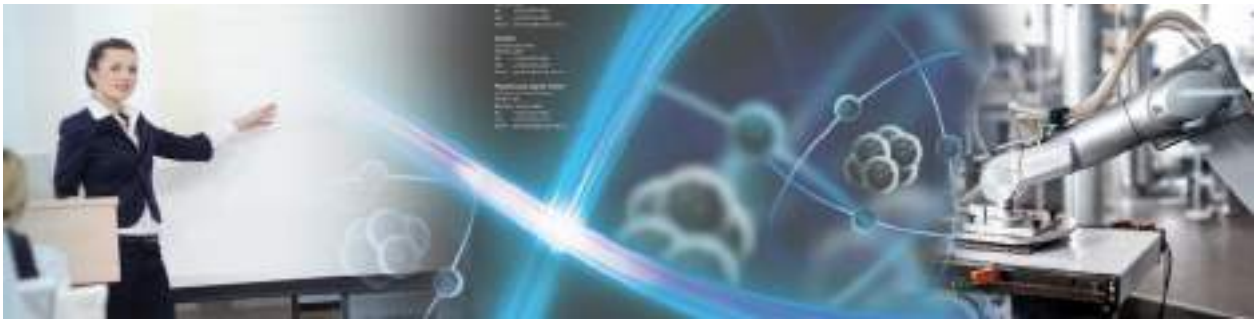
Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Кафедра комп'ютерно-інтегрованих технологій, автоматизації та робототехніки

IX Міжнародна Конференція
ВИРОБНИЦТВО
&
МЕХАТРОННІ СИСТЕМИ 2025



IX International Conference
MANUFACTURING
&
MECHATRONIC SYSTEMS 2025

M&MS
2025
IX International Conference
25-26 October
Kharkiv

УДК: 005:004.896:62-65:338.3

Виробництво & Мехатронні Системи 2025: матеріали IX-ої Міжнародної конференції, Харків, 25-26 жовтня 2025 р.: тези доповідей / [редкол. І.Ш. Невлюдов (відповідальний редактор)].-Харків: [електронний друк], 2025. – 115 с.

У збірник включені тези доповідей, які присвячені сучасним тенденціям розвитку технологій та засобів виробництва та мехатронних систем, передовому досвіду та впровадженню їх в галузях систем промислової автоматизації та керування виробництвом; системній інженерії; CAD/CAM/CAE системах; мехатроніці (електро-механічних системах, електронних інструментах систем керування, механічних CAD системах); робототехніці та засобахм інтелектуалізації; MEMS (сучасних матеріалів та технологіях виготовлення MEMS) та компонентах і технологіях автоматизації видобутку, переробки та транспортування нафти та газу.

Редакційна колегія: І.Ш. Невлюдов, В.В. Євсєєв.

Manufacturing & Mechatronic Systems 2025: Proceedings of IX st International Conference, Kharkiv, October 25-26, 2025: Thesises of Reports / [Ed. I.Sh. Nevlyudov (chief editor).] .- Kharkiv .: [electronic version], 2025. - 115 p.

The collection includes the thesises of reports on modern trends in the development of technologies and means of production and mechatronic systems, top experience and implementation of them in fields of: industrial automation and production management systems; systems engineering; CAD/CAM/CAE systems; mechatronics (electrical and mechanical systems, electronic control tools, mechanical CAD systems); robotics and intellectual toolls; MEMS (modern materials and manufacturing technologies MEMS) and components and technologies for the automation of oil, gas and oil extraction, processing and transportation.

Editorial board: Igor.Sh. Nevludov, Vladyslav.V. Yevsieiev

ЗМІСТ

“Web application security: protection against modern cyber threats” Overview of key vulnerabilities (XSS, CSRF, SQL injections), protection methods, use of HTTPS, authentication, and authorization

Anton Andreiev, Svetlana Sotnik

Department CITAR, Kharkiv National University of Radio Electronics, Ukraine,
Kharkiv, av. Nauki. 14., email: anton.andreiev@nure.ua

Anotation: The research highlights critical aspects of web application security in the context of growing cyber threats. The main focus is on the analysis of common web application vulnerabilities, including XSS, CSRF, and SQL injection, as well as modern protection methods against them. The work discusses the importance of using the HTTPS protocol and robust authentication and authorization systems as fundamental components of a secure web architecture. Particular attention is paid to a comprehensive protection approach that combines technical solutions with organizational measures. The research is not limited to theoretical aspects but provides practical recommendations for implementing protection systems in real-world web applications. A critical analysis of modern threats and countermeasures provides a balanced view of the state of web application security. Visual elements in the work contribute to a better understanding of complex security concepts. Overall, the research emphasizes the key role of a comprehensive security approach in ensuring the reliable operation of web applications.

***Key words*:** Web application security, cyber threats, XSS, CSRF, SQL injections, HTTPS, authentication, authorization.

I. INTRODUCTION

In today's digital world, web applications have become an integral part of business processes and everyday life [1, 2]. Their growing popularity and functionality are accompanied by increased cybersecurity risks. Web applications process vast amounts of confidential information, including user personal data, financial transactions, and corporate secrets, making them attractive targets for cybercriminals.

The processes of global informatization of society have radically changed the landscape of digital security [3-6].

Modern cyber threats are constantly evolving, becoming increasingly complex and sophisticated. Traditional protection methods are often insufficient to counter new types of attacks, such as complex XSS attacks, CSRF exploits, and SQL injections. In this context, the development and implementation of comprehensive web application security systems is becoming a critical task.

Ensuring the security of web applications is closely linked to the development of security technologies and secure development methodologies [7-10]. The use of HTTPS, robust authentication and authorization mechanisms, and the implementation of secure coding principles can significantly improve the security of web applications. The integration of these technologies contributes to the formation of new, more attack-resistant web systems that can not only withstand current threats but also adapt to future cybersecurity challenges [3-5].

Thus, in the context of the constant growth of cyber threats and the increasing complexity of web applications, the relevance of this research becomes obvious. A comprehensive approach to security, combining technical solutions with organizational measures, opens up new opportunities for improving the reliability, resilience, and security of web applications.

II. THE ROLE OF SECURITY MEASURES IN WEB APPLICATIONS

In the modern information environment, where the number of web applications is growing exponentially and cyber threats are becoming increasingly sophisticated, the role of security systems is becoming critical. Modern security technologies not only enable faster and less risky protection of confidential data but also adapt to new types of threats, increasing the overall level of web application security.

Table 1 below provides a detailed overview of how each of these measures impacts web application security and contributes to their protection.

Table 1. ROLE OF INTELLIGENT TOOLS IN INFORMATION RETRIEVAL SYSTEMS

Protective measure	The role in ensuring web application security
HTTPS protocol	Data encryption during transmission between the client and the server.
	Ensuring data integrity and protection against interception.
	Verification of website authenticity using SSL/TLS certificates.
Authentication systems	Verification of the user's identity before granting access to resources.
	Protection against unauthorized access using multi-factor authentication.
	Monitoring and logging access attempts for security analysis.

Continuation of Table 1

Protective measure	The role in ensuring web application security
Means of XSS attack prevention	Validation and filtering of input data to prevent the execution of malicious scripts.
	Using Content Security Policy (CSP) to control resources.
	Sanitizing output data for safe content display.

Analysis of the presented protective measures demonstrates a multi-layered approach to ensuring web application security. The Hypertext Transfer Protocol Secure (HTTPS) protocol forms the basic level of protection, ensuring the confidentiality and integrity of data during transmission. Authentication systems create a second layer of defense, controlling access to resources and identifying users.

Cross-site scripting (XSS) attack prevention tools implement proactive protection at the data processing level, blocking potentially malicious code before its execution.

The comprehensive application of all three categories of defense mechanisms is critically important for creating a reliable security system. Each layer of protection compensates for the potential weaknesses of others, forming an effective multi-layered security architecture (defense-in-depth). The absence or insufficient implementation of any of these components creates vulnerabilities that can be exploited by attackers to compromise the web application.

III. MAIN VULNERABILITIES OF WEB APPLICATIONS AND METHODS OF PROTECTION

The collection and protection of user data is a top priority in ensuring the security of web applications, as information compromise can lead to serious consequences – from financial losses to breaches of confidentiality and reputational risks for organizations [11-17]. Modern web applications face a wide range of security threats, requiring a comprehensive approach to their protection. The most common vulnerabilities include XSS, Cross-Site Request Forgery (CSRF), and SQL injections, each of which has its own characteristics and requires specific countermeasures. XSS attacks are among the most common threats to web applications. These attacks allow attackers to inject malicious code into web pages viewed by other users. The concept of XSS attack protection is shown in Fig. 1.

Fig. 1. Concept of protection against XSS attacks

At the «Validation and Sanitization» stage, all user-entered data is checked for potentially dangerous code. Special filters and regular expressions are used to detect and neutralize malicious scripts.

At the «Output Encoding» stage, all data displayed on the web page is processed to prevent the execution of JavaScript code. Special characters are converted into their HTML equivalents.

CSRF attacks exploit the web application's trust in an authenticated user, tricking them into performing unwanted actions without their knowledge or consent. An attacker creates a malicious request that is executed on behalf of the victim using their active session. This can lead to unauthorized financial transactions, changes to account credentials, modification of security settings, or the execution of other critical operations. The peculiarity of CSRF is that the attack exploits the browser's mechanism of automatically sending cookies, as a result of which the server cannot distinguish a legitimate request from a forged one.

The concept of protection against CSRF attacks is shown in Figure 2.

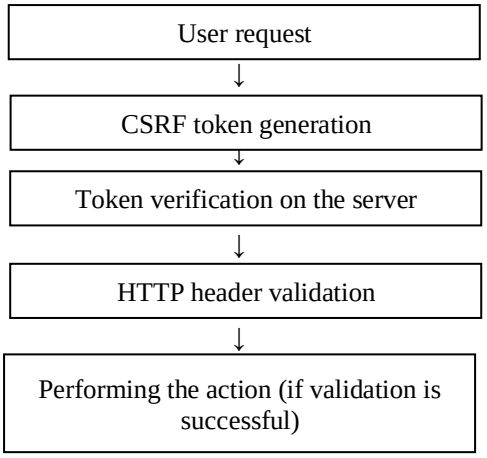


Fig. 2. The concept of protection against CSRF attacks

1. Generating unique tokens. For each user session, a unique CSRF token is generated and embedded into all forms and AJAX requests. This token serves as a cryptographic proof that the request originated from the legitimate application. The server validates the token upon receiving any state-changing request, comparing it against the token stored in the user's session. If the tokens do not match or the token is missing, the request is rejected. This mechanism effectively prevents attackers from forging requests, as they cannot predict or obtain the randomly generated token without access to the victim's session.

2. HTTP header verification. The server examines the Referer and Origin headers to confirm that requests originate from trusted sources. The Origin header indicates the domain that initiated the request, while the Referer header provides the full URL of the page from which the request was sent. By implementing strict validation of these headers, the server can identify and block requests coming from external or malicious domains. This approach adds an additional layer of defense, particularly effective against basic CSRF attacks where the attacker hosts the malicious content on a different domain.

3. Using SameSite cookies. Configuring cookies with the SameSite attribute prevents their transmission in cross-site

requests, significantly mitigating CSRF risks. The SameSite attribute can be set to "Strict" (cookies are never sent in cross-site requests), "Lax" (cookies are sent only with safe HTTP methods like GET from external sites), or "None" (cookies are sent with all requests, requiring Secure flag). By default, setting SameSite to "Strict" or "Lax" ensures that session cookies are only included in requests originating from the same site, making it nearly impossible for attackers to exploit the user's authenticated session from external domains.

SQL injection attacks allow malicious actors to execute arbitrary SQL commands within the application's database, potentially leading to data breaches, unauthorized data modification, or complete system compromise.

The concept of protection against SQL injection is illustrated in Figure 3.

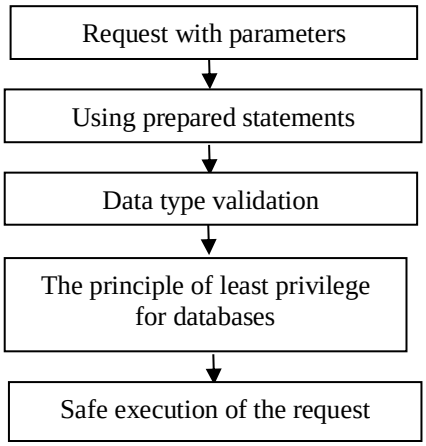


Fig. 3. Concept of NLP

1. Prepared statements. The use of parameterized queries ensures a clear separation between SQL code and data, preventing input data from being interpreted as SQL commands. In this approach, the SQL query structure is defined first with placeholders for user-supplied values, and then the actual data is passed separately as parameters. The database engine treats these parameters strictly as data values, not as executable code, regardless of their content. This technique eliminates the possibility of SQL injection even if an attacker attempts to inject malicious SQL syntax, as the database will never execute the injected code. Prepared statements also offer performance benefits through query plan caching and are supported by virtually all modern database systems and programming frameworks.

2. Input validation and sanitization. All data received from users undergoes rigorous validation to ensure compliance with expected types, formats, and acceptable value ranges. This includes implementing whitelist validation (accepting only known-good input patterns), checking data length constraints, verifying data types (integers, strings, dates), and rejecting inputs containing suspicious characters or SQL keywords. Additionally, input sanitization involves escaping special characters that have meaning in SQL context, such as quotes, semicolons, and comment markers. Multi-layer validation should occur both on the client side (for user experience) and critically on the server side (for security), as client-side validation can be easily bypassed by attackers.

3. Principle of least privilege. Database accounts used by the application are granted only the minimum permissions necessary to perform their intended functions, significantly limiting the potential damage from a successful SQL injection attack. Instead of using administrative or root database accounts, the application should operate with restricted accounts that have access only to specific tables and can execute only required operations (select, insert, update, delete). For example, if a particular application component only needs to read data, its database account should not have insert, update, or delete privileges. Additionally, sensitive operations like drop table, create user, or access to system tables should be completely prohibited for application accounts. This layered security approach ensures that even if an attacker successfully injects SQL code, the scope of possible malicious actions remains severely limited.

IV. HTTPS AND CRYPTOGRAPHIC PROTECTION

The HTTPS protocol is a fundamental component of web application security, ensuring the encryption of data transmitted between the client and the server. The use of HTTPS not only protects confidential information from interception but also verifies the authenticity of the website.

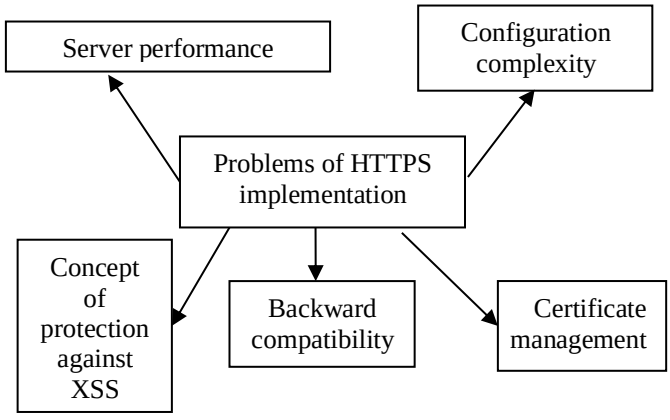
The principles of the HTTPS protocol are based on the use of SSL/TLS certificates and asymmetric encryption, which enable web applications to:

Ensure data confidentiality by encrypting all traffic between the client and the server;

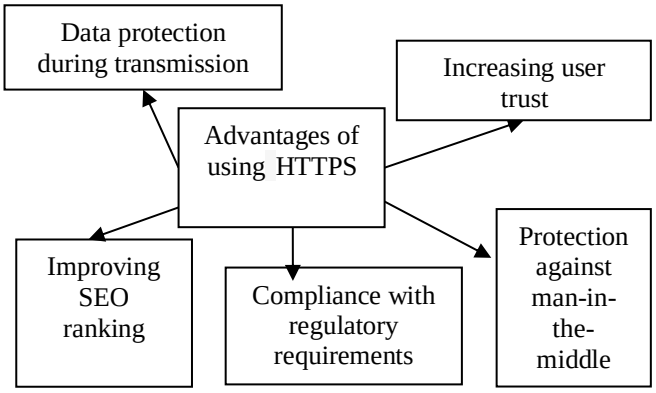
Guarantee data integrity through the use of hash functions and digital signatures;

Verify server authenticity via a system of trusted certificate authorities.

The challenges and advantages of problems HTTPS are shown in Fig. 4, a, b.



a)



b)

Fig. 4. Concept of deep learning and neural networks: a) Problems with HTTPS implementation; b) Advantages of using HTTPS

Problems of HTTPS implementation:

1. Server load. Additional computational overhead for encrypting and decrypting data, which can impact performance under high concurrent connections.

2. Certificate management. Requires regular renewal of SSL certificates and proper configuration of certificate trust chains.

3. Configuration complexity. Involves correct web server setup, selection of appropriate cipher suites, and configuration of security headers.

Advantages of Using HTTPS:

1. Data interception protection. Ensures the confidentiality of logins, passwords, financial data, and other sensitive information.

2. Increased user trust. Achieved through browser security indicators (green lock, «Secure» label).

3. Improved SEO Ranking. Search engines give preference to HTTPS sites in search results.

4. Regulatory Compliance. Meets standards like PCI DSS for payment card processing and GDPR for personal data protection.

Practical recommendations for HTTPS implementation:

1. Use HTTP Strict Transport Security (HSTS) to enforce HTTPS connections.

2. Configure Perfect Forward Secrecy for enhanced encryption key protection.

3. Implement Certificate Transparency for monitoring certificate issuance.

4. Utilize free certificates from Let's Encrypt to reduce implementation costs.

V. CONCLUSIONS

This research has conducted a comprehensive analysis of critical aspects of web application security in the context of constantly evolving cyber threats. The main achievements and contributions of this work include:

1. Systematic analysis of key vulnerabilities. The study has thoroughly examined the three most prevalent and dangerous web application vulnerabilities: XSS attacks, CSRF exploits, and SQL injections. For each vulnerability type, detailed attack mechanisms have been described along with their potential consequences, ranging from data theft to complete system compromise.

2. Development of comprehensive protection concepts. Visual concepts and multi-layered defense strategies have been developed for each identified vulnerability type. These concepts demonstrate the practical implementation of protection mechanisms, including input validation, token generation, prepared statements, and the principle of least privilege. The presented approaches provide actionable guidance for developers and security specialists.

3. Evaluation of cryptographic protection and HTTPS protocol. The research has analyzed the fundamental role of the HTTPS protocol in ensuring web application security, examining both its advantages (data confidentiality, integrity verification, regulatory compliance) and implementation challenges (server load, certificate management complexity). Practical recommendations for HTTPS deployment have been formulated, including the use of HSTS, Perfect Forward Secrecy, and Certificate Transparency.

4. Substantiation of the defense-in-depth approach. The work emphasizes that effective web application security can only be achieved through a comprehensive, multi-layered protection strategy. The integration of HTTPS protocol, robust authentication systems, and vulnerability-specific countermeasures creates a resilient security architecture where each layer compensates for potential weaknesses of others.

The developed concepts and recommendations can be directly applied in the design and implementation of secure web applications. The systematic approach to security presented in this work enables organizations to:

- identify and prioritize protection against the most critical vulnerabilities;
- implement layered defense mechanisms adapted to specific threat landscapes;
- ensure compliance with international security standards and regulations;
- minimize risks of data breaches and associated financial and reputational losses.

Further development of this work should focus on analyzing emerging threats such as advanced persistent threats, zero-day exploits, and AI-powered attacks. Additionally, research on automated security testing tools, integration of machine learning for threat detection, and development of secure-by-design development methodologies represents promising directions for enhancing web application security.

The growing informatization of society and the increasing sophistication of cyber threats make the continuous improvement of web application security systems not just a technical necessity but a fundamental requirement for sustainable digital transformation across all sectors of the economy and society.

LIST OF REFERENCES

[1] A. S. Andreiev, et al., “Computer games and Web design,” *Proceedings of the XVII International scientific and practical conference «Information technologies and automation – 2024»*, 2024, pp. 712-714

[2] M. Rudenko, et al., “Overview of approaches to scaling relational databases in development and adaptation of web applications,” *Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: Тези доповідей XII Міжнародної науково-практичної конференції (10-12 грудня 2024 р., м. Запоріжжя). [Електронний*

ресурс] /Електрон. дані. – Запоріжжя: НУ «Запорізька політехніка», 2024, pp. 398-402

[3] S. Sotnik, M. Rudenko, "Evaluating relational database scaling strategies in web engineering," *International Conference on Advanced Trends In Radioelectronics and Infocommunications (ATRIC-2025) (May 21–22, 2025), Lviv Polytechnic Publishing House, Lviv, Ukraine*, 2025, pp. 224-228

[4] S. Sotnik et al., "Development Features Web-Applications," *International Journal of Academic and Applied Research (IJAAR)*, 2023, Vol. 7, Issue 1, pp. 79-85

[5] S. V. Sotnik, "Analysis of Personal Information Security Issues in Peacetime and Wartime," *International Journal of Academic Engineering Research (IJAER)*, 2024, Vol. 8, Issue 10, pp. 108-113

[6] S. V. Sotnik, "Features of using REST architecture for development of ARS for information systems," *Міжнародна науково-практична конференція «Інформаційні системи в управлінні проєктами та програмами», Коблево, 9–13 вересня 2024 р. Збірник праць. – Харків: ХНУРЕ*, 2024, pp. 42 – 45

[7] A. Tverdokhlib, et al., "Intelligent tools for optimizing information and search engines," *Manufacturing & Mechatronic Systems 2024: Proceedings of VIII st International Conference, Kharkiv, October 25-26*, 2024, pp. 28-31

[8] R. V. Marunich, et al., "Features of IoT application in the security sector," *«Computer-integrated technologies, automation and robotics» CITAR-2025*, 2025, pp. 80-84

[9] С. Сотник, "Розробка автоматизованої інформаційно-пошукової системи вибору маніпулятора промислових роботів," *Електромеханічні і енергозберігаючі системи*, 2025, 1 (68), pp. 52-58

[10] Y. I. Khalimonov, et al., "Integration of IoT into security systems: opportunities and risks," *Комп'ютерно-інтегровані технології автоматизації технологічних процесів на транспорті та у виробництві : матеріали всеукр. наук.-практ. конф. здобувачів вищ. освіти і молодих учених, 20 листоп. 2024 р.*, 2024, pp. 117-121

[11] M. S. Achkan, et al., "Integration of cloud technologies into modern SCADA systems: prospects and challenges," *«Computer-integrated technologies, automation and robotics» CITAR-2025*, 2025, pp. 26-29

[12] R. V. Marunich, et al., "Modern IoT technologies for creating automated access systems," *Sustainable smart cities and communities: business and innovation solutions 2025: Proceedings of I st I International Conference, Kharkiv, April 21, 2025: Thesises of Reports*, 2025, pp. 38-39

[13] S. V. Sotnik, et al., "Analysis of searching methods for explosive objects using information technology and computer modeling," *Стан, досягнення та перспективи інформаційних систем і технологій / Матеріали XXIV Всеукраїнської науково-технічної конференції молодих вчених, аспірантів та студентів. Одеса, 18-19 квітня 2024 р. - Одеса, Видавництво ОНТУ*, 2024, pp. 20-22

[14] A. Konieva, et al., "Main trends in the development of automated image processing systems," *«Computer-integrated technologies, automation and robotics» CITAR-2025*, 2025, pp. 68-7

[15] S. Sotnik, "Integration of IoT into security systems: opportunities and risks," *International Journal of Academic Engineering Research (IJAER)*, 2024, 8 (11), pp. 56-61

[16] K. A. Polikanov, et al., "Overview of modern technologies for residential automation," *«Computer-integrated technologies, automation and robotics» CITAR-2025*, 2025, pp. 85-89

[17] O. R. Kolbasa, et al., "The significance and necessity of automating the selection of sensors and actuators," *«Computer-integrated technologies, automation and robotics» CITAR-2025*, 2025, pp. 63-67