SECURING DOCUMENT ACCESS IN WEB APPLICATIONS

Petar Milić*,
[0000-0003-0427-8379]
Dragiša Miljković,
[0000-0002-5350-7265]
Stefan Pitulić
[0000-0002-7976-0680]

University of Pristina – Kosovska Mitrovica, Faculty of Technical Sciences, Kosovska Mitrovica, Serbia

Abstract:

Ensuring document access protection within web applications presents significant challenges for developers utilizing modern web development frameworks. The task of facilitating secure access and mitigating vulnerabilities, while preventing exposure of sensitive information to attackers, requires the application of advanced protection methods and techniques. In this paper, we conduct a comparative analysis of various document protection methods and techniques, evaluating their strengths and weaknesses against the OWASP Top 10, a crucial benchmark for security awareness in the field. We then introduce a novel approach to document protection within web applications, detailing the benefits and potential drawbacks of this method. Moreover, this paper underscores the vital importance of document access protection in web applications, which goes beyond mere file storage.

Keywords:

Web Application Security, Document Protection, Access Control, Data Security, Encryption.

INTRODUCTION

Web applications offer a modern approach for accessing network resources conveniently and on-demand, serving various purposes. Consequently, these applications are susceptible to network-originated attacks, posing significant security challenges. Despite the advanced frameworks used in their development, which provide a higher level of data protection than traditional web applications, modern web applications remain vulnerable to such threats [1].

Vulnerabilities in web applications can stem from improper implementation, poorly configured web servers, or the lack of strong privacy policies. Therefore, the security of web applications is a paramount concern during their deployment. The Open Web Applications Security Project (OWASP) Top Ten, an essential benchmark for web application security, outlines the most common risks. The latest version, updated for 2021 as shown in Figure 1, highlights new areas of concern such as 'Insecure Design' and 'Software and Data Integrity Failures', while continuing to stress the importance of 'Security Logging and Monitoring'. These developments reflect the ever-changing threat environment and validate the necessity for ongoing vigilance and adaptation in security practices [2].

Correspondence:
Petar Milić

e-mail:
petar.milic@pr.ac.rs

DOI: 10.15308/Sinteza-2024-60-65

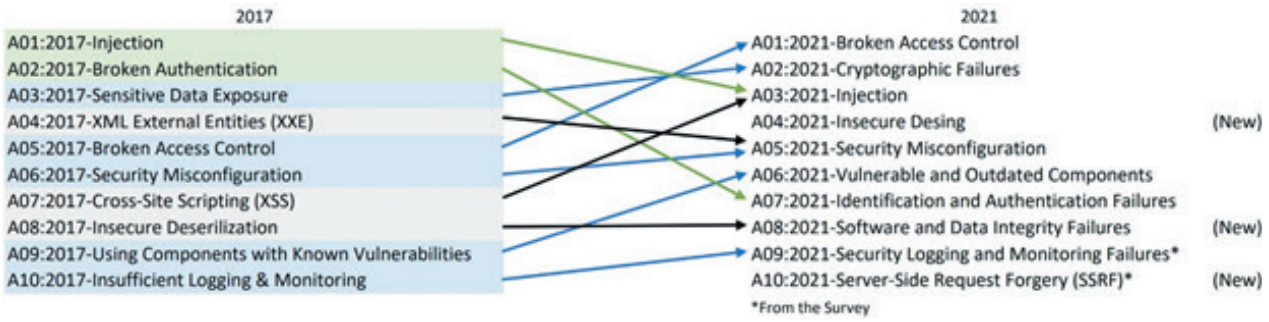

Figure 1. OWASP Top Ten Vulnerability 2021.

The securing of data via web frameworks and the safeguarding of documents are crucial in contemporary web applications [3]. Web frameworks play a pivotal role in developing secure and robust web applications, requiring developers to utilize the security features they offer to shield their applications from potential exploits. Furthermore, encryption is essential for protecting data in transit. Web frameworks often support encryption protocols like HTTPS to facilitate secure communication between clients and servers, thereby ensuring that sensitive information remains confidential and protected from eavesdropping or tampering by malicious entities.

The protection of documents in web applications extends beyond file storage. Proper access controls should be implemented to prevent unauthorized access to confidential documents. Role-based access control (RBAC) mechanisms may be employed to administer permissions effectively, ensuring that only authorized personnel can view or modify specific documents.

This paper aims to explore methodologies and strategies for securing document access in modern web applications. Our review of the existing literature indicates that achieving adequate document protection involves employing a multi-layered defence mechanism, enabling their secure use within web applications. However, our findings suggest that the current protection mechanisms do not fully provide the necessary level of security. Throughout this paper, we perform a comparative analysis of the various methods and techniques for securing document access within web applications, outlining their advantages and disadvantages with reference to the OWASP Top 10 as a benchmark awareness document in this field. Additionally, we present a particular approach for document protection and discuss the outcomes of its application. We conclude by summarizing our findings and suggesting directions for future research, including essential recommendations for improving document security in contemporary web applications.

2. BACKGROUND

As highlighted previously, protecting documents in web applications involves more than mere file storage, particularly because they may contain privacy-sensitive information [4]. Encrypting data and communication channels during user interactions with web applications is a crucial step in addressing this concern. Nevertheless, documents transmitted through web applications can sometimes become accessible to unintended recipients due to misconfigurations of the applications and servers [5]. Organizations like OWASP (Open Web Application Security Project), PCI-SSC (Payment Card Industry – Security Standards Council), IETF (Internet Engineering Task Force), and IEEE (Institute of Electrical and Electronics Engineers) offer essential recommendations for establishing secure mechanisms for both authentication and authorization. They advise on security testing of web applications to identify vulnerabilities in software code, session management, role-based access control, and encryption algorithms [6].

Given these security challenges, it is vital to enhance both authentication and authorization in web applications [7]. A promising approach to addressing these challenges involves the integration of machine learning into the authentication process. For example, a system may require second-factor authentication, such as a one-time code sent via SMS, especially when the classifier detects suspicious activity with low confidence. However, many web applications continue to face challenges in effectively managing authorization, often requiring advanced technological solutions at the backend to mitigate these issues [8]. This approach delineates user access boundaries within the web application. Moreover, by designing a secured architecture for preserving privacy and security inside applications, it is possible to safeguard against various risks, threats, and attacks [1].

Almin [9] argues that securing a web application requires developers to clearly define what security means in the specific context of their project. He recommends following the OWASP guidelines and using the OWASP Application Security Verification Standard (ASVS) as a guideline for establishing security requirements for the applications. Almin also asserts that integrating security considerations from the project's inception is more cost-effective than retroactively incorporating them. In support of this, [10] highlights the significant challenge in developing robust yet user-friendly security controls, noting that adopting standardized security controls can greatly ease the process of creating secure applications. Other scholars [11, 12, 13, 14] propose the formulation and implementation of customized software security strategies tailored to the specific risks an organization faces.

AAA [15] emphasizes the detailed nature of privacy disclosures under legal regulations, while Peukert et al. [16] note the GDPR's wider impact on sectors like antitrust, necessitating interdisciplinary research to comprehend its full implications on document security."

Legal regulation provides a crucial perspective on user interactions with information stored and accessed through web applications [14]. The General Data Protection Regulation (GDPR), enforces rigorous data protection and privacy standards for web applications within the European Union (EU) and the European Economic Area (EEA), directly affecting document security protocols. It compels applications to safeguard personal data from unauthorized access and obliges transparency in document access controls, aligning with principles of legality and fairness. Web applications must therefore strengthen their security measures, including encryption and breach response, to comply with GDPR mandates. In this regulatory context, AAA [15] underscores the precision with which legal regulation defines privacy disclosures, while Peukert et al. [16] highlight the broader implications of the GDPR, signifying its influence on policy areas such as antitrust and trade. They also indicate how closely intertwined these areas of the law have become, implicating that theoretical relationship between privacy and antitrust laws should be researched.

3. ANALYSIS OF METHODS AND TECHNIQUES FOR DOCUMENT PROTECTION

Protecting documents accessible through web applications presents a significant challenge for developers and administrators. To more thoroughly understand the advantages and disadvantages of existing tools, this section aims to establish criteria for evaluating methods, techniques, and mechanisms employed in document protection. Ensuring the security of documents within a web environment is essential for achieving a sufficient level of overall system security.

3.1. ENCRYPTION PROTOCOLS

The application of encryption protocols is a first step in protecting whole web applications and consequently documents available through them. These protocols ensure confidentiality and integrity during the exchange of information via web. TLS and SSL are widely adopted protocols for this purpose, encrypting communication between a user's browser and the web server to prevent eavesdroppers from intercepting and deciphering the transmitted data. Moreover, encryption techniques such as DES, RC4, and Blowfish have shown vulnerabilities due to the use of weak keys in the ciphering process. Sophisticated attacks that utilize GPUs for more potent brute force attempts increase security risks [17].

3.2. AUTHENTICATION AND ACCESS CONTROL

An unavoidable aspect of securing web applications is the verification of identity, which includes identifying the user while accessing data and services, as well as systems interacting with each other. These are mostly used in line with access control mechanisms to ensure what actions authenticated users are allowed to perform and what resources they can access. Various access control mechanisms exist, such as RBAC (Role Based Access Control), MAC (Mandatory Access Control), DAC (Discretionary Access Control), PBAC (Policy Based Access Control), TBAC (Task Based Access Control), ABAC (Attribute Based Access Control), FGAC (Fine Grained Access Control) and etc [18]. Implementing these mechanisms reduces the risk of unauthorized data exposure.

3.3. WEB SERVER SECURITY

Securing a web server primarily involves key concepts such as regular software updates, security configurations, secure file uploads, and security headers. By incorporating these measures into a web server security strategy, one can create a resilient and secure web environment, thereby enhancing the overall document protection. The heterogeneity of the web environment becomes a challenge in creating secure information

exchange [12]. This is confirmed by OWASP, which identifies security misconfiguration as one of the top categories in the entire environment [2].

3.4. SECURITY POLICIES

Implementing security policies can aid in preventing attacks on web applications and consequently in protecting them as a whole [19]. These policies, derived from best practices presented in industry-driven research, are intended to mitigate potential threats. Furthermore, [2] suggests the use of a policy called 'Principle of Least Privilege,' which checks all the resources used by the web application against established authorization rules. Moreover, addressing issues such as broken access control, identification and authentication failures through adequate security policies, according to OWASP, can contribute to increasing the level of security of a web application. Lala, Kumar and Subbulakshmi [20] showed that the implementation of OWASP guidelines and policies renders web applications more resilient to attacks and security breaches.

4. NOVEL APPROACH FOR DOCUMENT ACCESS PROTECTION

To achieve comprehensive document protection in web applications, we developed an approach illustrated in Figure 2. This methodology comprises four pivotal components: "User Authentication," "User Authorization," "File Access Permissions," and ".htaccess Permissions," each tasked with safeguarding a distinct aspect of web-based document access.

The initial two components, as Figure 2 illustrates, are foundational. They manage user authentication, ensuring access is granted solely to users with legitimate credentials—a widely adopted standard in contemporary practices. Furthermore, user authorization is indispensable for verifying whether a user holds the requisite permissions for accessing specific segments of the web application and, by extension, the documents therein. The importance of file access permissions cannot be overstated, as they fortify documents against unauthorized access and their inadvertent exposure online. The strategic configuration of .htaccess files constitutes the final element of our approach. These files are instrumental in shielding documents within the web application by preventing direct web access, thus reinforcing the efficacy of other security measures.

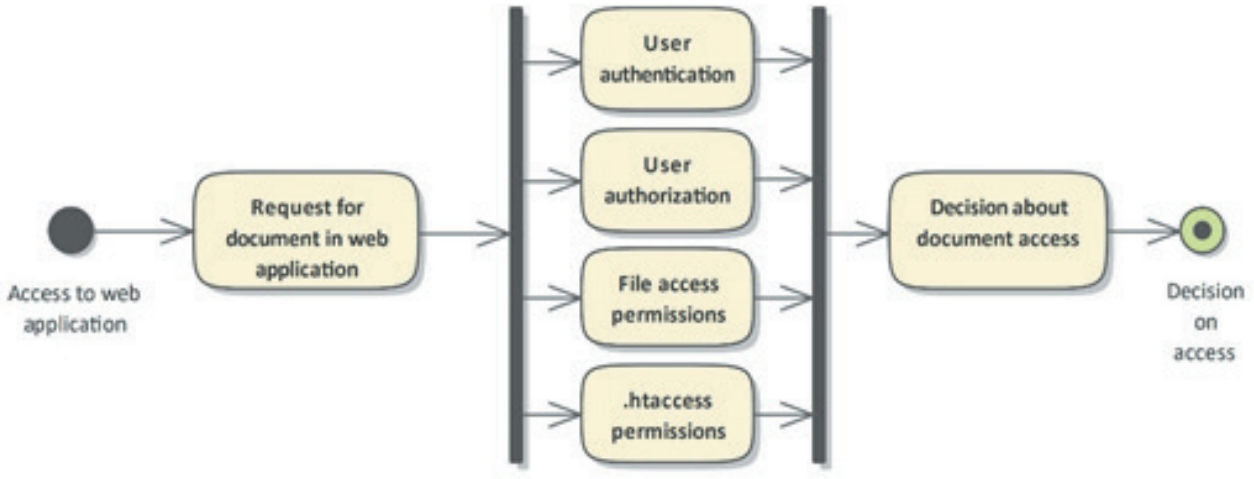

Figure 2. Steps in proposed approach.

```
RewriteEngine on
RewriteCond %{HTTP_REFERER} !^https://(www\.)?domain.example.com [NC]
RewriteCond %{HTTP_REFERER} !^https://(www\.)?domain.example.com.*$ [NC]
RewriteRule \.(PDF|pdf)$ - [F]
```

Listing 1. An example of .htaccess configuration.

For access to be granted, it is imperative that all four components yield a positive outcome; should any component produce a negative result, access to the document will be accordingly denied. Example of proper .htaccess configuration is provided below.

In above configuration commands, it can be seen that access requests for documents within the web application are approved only if originating from a recognized HTTP REFERER and seeking a specific document type, such as PDF in this example. Employing this approach enhances the security of document access in web applications by approving requests from verified users and sources, previously authorized to access certain web application areas. This strategy is particularly pertinent for web applications engineered using modern web frameworks, emphasizing user authentication and authorization without sufficiently addressing file access permissions and the protection against unauthorized HTTP REFERERs. The reason why we have exploited the '.htaccess' as it serve as powerful configuration tools that allow administrators to enhance the security of their web applications hosted by Apache2 web servers. This decentralized approach to configuration enables quick adjustments without the need for server-wide changes, making it particularly useful for shared hosting environments. Additionally, '.htaccess' provides a flexible and granular control over permissions, allowing you to restrict or grant access based on various criteria.

In the context of Nginx, the use of configuration files, such as nginx.conf and site-specific configuration files, can produce similar effect as in our example. Nginx allows administrators to employ server block configurations to control access and manage various security-related directives. When it comes to Tomcat, securing web applications is achieved through its configuration files, such as server.xml and web.xml. Tomcat's Manager application, combined with proper configurations, enables remote management and monitoring with security considerations. The flexibility of Tomcat's configuration options allows administrators to ensure a robust defense against potential vulnerabilities.

It is presupposed that the web application employs encrypted protocols, a fundamental prerequisite for safeguarding the web application and, consequently, the documents it houses. By adopting the proposed methodology, all vulnerabilities listed in the OWASP Top 10, as depicted in Figure 1, are effectively circumvented, ensuring a robust security level. This aligns with the findings of Lala, Kumar, and Subbulakshmi [20] who proved that adherence to OWASP guidelines and policies significantly enhances the resilience of web applications against attacks and security breaches. It's essential to prioritize the establishment of suitable security policies tailored to the intended use of the web application, the environment in which it operates, and the management of potential threats.

5. CONCLUSION

In this paper we have conducted an in-depth exploration of methodologies and strategies to secure document access in contemporary web applications. An extensive examination of the prevailing literature has revealed that the establishment of robust document protection is contingent upon the deployment of a comprehensive, multi-layered defence strategy. Such a strategy is essential to facilitate the secure management and use of documents within web applications.

Moreover, we introduced a new method for enhancing document security in web applications. This method not only augments the security landscape, but also plays a pivotal role in safeguarding against the inadvertent exposure of sensitive data. This contribution is particularly significant in the context of increasing threats and the complex demands of modern web infrastructure.

Nonetheless, our investigation has identified that existing security measures frequently fall short of achieving the requisite standards of protection. In response, we advocate for the adoption of the OWASP Top 10 guidelines as a foundational benchmark for security awareness within this domain. This strategic endorsement is critical for addressing the current deficiencies in security practices effectively. Further, we propose a detailed examination of diverse hashing and encryption methodologies, coupled with advanced fingerprinting techniques and the strategic use of salt values, to detect and mitigate malformed request vectors. Such a multifaceted security approach is aimed at significantly enhancing the protective measures for document access within web applications, thereby contributing to a more secure and resilient digital environment.

In conclusion, despite progress in web application security, our findings emphasize the need for continuous research and innovative solutions. The proposed method represents a step forward in this endeavour, offering a blueprint for enhancing document security in an increasingly interconnected and digitalized world.

6. REFERENCES

[1] I. Kabanov, "Scalable frameworks for application security and data protection," in *In Proceedings of the 11th International Global Security, Safety and Sustainability Conference*, London, UK, 2017.

[2] OWASP Foundation, "OWASP Top 10," OWASP Foundation, [Online]. Available: https://owasp.org/www-project-top-ten/. [Accessed 11 01 2024].

[3] R. A. Oliveira, M. M. Raga, N. Laranjeiro and M. Vieira, "An approach for benchmarking the security of web service frameworks," *Future Generation Computer Systems*, vol. 110, no. 1, pp. 833 - 848, 2020.

[4] P. Milić, K. Kuk, T. Civelek, B. Popović and S. Kartunov, "The Importance of Secure Access to E-Government Services," in *In Proceedings of The Archibald Reiss Days*, Belgrade, Serbia, 2016.

[5] J. Li, "Vulnerabilities Mapping based on OWASP-SANS: A Survey for Static Application Security Testing (SAST)," *Annals of Emerging Technologies in Computing (AETiC)*, vol. 4, no. 3, pp. 1 - 8, 2020.

[6] S. Raffique, H. Mamoona, H. Bushra, A. Ansar, M. Akhtar and K. Iqbal, "Web application security vulnerabilities detection approaches: A systematic mapping study," in *In Proceedings of the 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, Takamatsu, Japan, 2015.

[7] I. Indu and P. R. Anand, "Hybrid authentication and authorization model for web based applications," in *In Proceedings of the International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET),* Chennai, India, 2016.

[8] J. V. Chandra, N. Challa and S. K. Pasupuletti, "Authentication and authorization mechanism for cloud security," *International Journal of Engineering and Advanced Technology*, vol. 8, no. 6, pp. 2072-2078, 2019.

[9] S. B. Almin, "Web Server Security and Survey on Web Application Security," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 2, no. 1, pp. 114 - 119, 2014.

[10] M. A. Helmiawan, E. Firmansyah, I. Fadil, Y. Sofiwan, F. Mahardika and A. Guntara, "Analysis of Web Security Using Open Web Application Security Project 10," in *In Proceedings of the 8th International Conference on Cyber and IT Service Management (CITSM)*, Pangkal, Indonesia, 2020.

[11] M. Jangjou and M. K. Sohrabi, "A comprehensive survey on security challenges in different network layers in cloud computing," *Archives of Computational Methods in Engineering*, vol. 29, no. 6, pp. 3587-3608, 2022.

[12] A. Chowdhary, H. Dijiang, M. S. JayjaSurya, D. Romo, D. Yuli and S. Abdulhakim, "Autonomous security analysis and penetration testing," in *In Proceedings of the 16th International Conference on Mobility, Sensing and Networking (MSN)*, Tokyo, Japan, 2020.

[13] F. Pereira, P. Crocker and V. R. Q. Leirthardt, "PA-DRES: Tool for PrivAcy, Data REgulation and Security," *SoftwareX*, vol. 17, no. 1, pp. 1 - 5, 2022.

[14] R. Amos, A. Gunes, E. Lucherini, M. Kshirsagar, A. Narayanan and J. Mayer, "Privacy policies over time: Curation and analysis of a million-document dataset," in *In Proceedings of the 2021 Web Conference*, Ljubljana, Slovenia, 2021.

[15] S. Duggenini, "Impact of Controls on Data Integrity and Information Systems," *Science and Technology*, vol. 13, no. 2, pp. 29 - 35, 2023.

[16] C. Peukert, S. Bechtold, M. Batikas and K. Tobias, "European Privacy Law and Global Markets for Data," ETH, Zurich, 2020.

[17] S. Kartunov, P. Milić, K. Kuk and U. Dinić, "Protection of user credentials in web application," in *In Proceedings of the "Archibald Reiss Days" International conference*, Belgrade, Serbia, 2020.

[18] P. Milić, K. Kuk, S. Trajković, D. Ranđelović and B. Popović, "Security analysis of open source databases in web application development," in *In Proceedings of the International Scientific Conference - UNITECH 2016*, Gabrovo, Bulgaria, 2016.

[19] A. Anis, "Securing web applications with secure coding practicies and integrity verification," Queen's University, Ontario, Canada, 2018.

[20] S. K. Lala, A. Kumar and T. Subbulakshm, "Secure Web development using OWASP Guidelines," in *In Proceedings of the 5th International Conference on Intelligent Computing and Control Systems (ICICCS 2021)*, Madurai, India, 2021.