International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 05 May 2024, pp: 593-603 www.ijaem.net ISSN: 2395-5252

A Model for a Secured and Reusable Web Content Management System with Wordpress

Ojemade Samuel[1], Omede G.C[2], Abel Edje[3] and Akazue Maureen[4]

1, 2, 3, 4: Computer Science Department, Delta State University, Abraka, Nigeria

Date of Submission: 10-05-2024 Date of Acceptance: 20-05-2024

ABSTRACT
With an explicit focus on fortifying cybersecurity and conducting a meticulous vulnerability analysis there is need to implements a robust security measure in all platforms or systems. This article introduces a pioneering and secured approach to web content management utilizing the WordPress platform. Key strategies employed include the implementation of regular updates, the integration of Two-Factor Authentication (2FA) for enhanced user authentication, SSL implementation for secure data transmission, and proactive measures to counter prevalent threats such as SQL injection and XSS attacks. Integral to this approach is an emphasis on user education, empowering users to recognize and report potential threats, contributing to a collective defence against cyber risks. The system secures content management system undergoes rigorous testing against diverse cyber threats, demonstrating remarkable resilience and effectiveness in mitigating security risks. This research signifies a substantial advancement in the field of web security, offering valuable insights to enhance the security posture of WordPress-based content management systems and thereby contributing to the overall fortification of web-based platforms.

Keyword: WordPress, Two-Factor Authentication, Content Management System, Secure Sockets Layer

I. INTRODUCTION

In today's digital era, the internet has become an integral part of our daily lives. Websites, in particular, have evolved into a powerful medium for communication, information dissemination, and business transactions. The increasing adoption of Internet technologies to provide flexible learning experiences such as Massive Open Online Courses (MOOCs) and traditional eLearning methods has heightened the importance of information systems security [1]. Safeguarding the integrity and confidentiality of data remains a significant challenge as institutions strive to offer accessible and versatile learning opportunities to students, emphasizing the pressing need for robust security measures.As per the Cyber Security Report of 2016[2], the suboptimal selection of Web Content Management Systems (WCMS), combined with a deficiency in awareness of security threats among technology users, considerably hinders the optimal utilization of WCMS applications within university environments.Challenges in this domain encompass deficiencies in cyber and digital security expertise, as well as a noticeable absence of security consciousness within the user community. For instance, a prevailing culture of not prioritizing security threats during online activities has been identified as a notable issue [3].

As a result, the use of efficient and secure web content management systems (CMS) has grown exponentiallyWordPress, a widely adopted CMS, has played a significant role in empowering individuals and organizations to create and manage web content [4]. More importantly, WordPress is a PHP core CMS that allows for light server-end website designs compared to other server-end languages like Perl, Node.js, Java, etc. Hence, WordPress has become a paradigm for creating secure and adaptable websites. With its intuitive interface and wide array of plugins, WordPress has not only gained prominence as a leading content management system (CMS) but also serves as a flexible option for educational institutions aiming to establish a dynamic online footprint [5]. Its adaptability allows for the creation of interactive and engaging educational websites, fostering a

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 05 May 2024, pp: 593-603 www.ijaem.net ISSN: 2395-5252

conducive environment for learning. However, the very ubiquity of WordPress exposes it to cybersecurity challenges [2], necessitating a focused and comprehensive approach to fortify its security features.

In Light of this, a lot of individuals have misunderstood the difference between a WordPress User and a WordPress developer as both are not the same[6]. The earlier entails an individual who makes use of WordPress themes and Plugins to develop a website, while the former entails an individual who is proficient in HTML, CSS, and PHP languages to develop websites, themes, or plugins to make it easy for WordPress users to be able to create a secure and reusable website with as little as programming knowledge, and this research work would be looking at things from the perspective of WordPress developer and not a WordPress user.

II. REVIEW OF RELATED LITERATURE

In the current digital era, WordPress stands out as one of the most popular and adaptable content management systems for website development. However, its widespread usage makes its main target for malicious entities aiming to make it vulnerable and compromise website security. Given the paramount importance of website security for both businesses and individuals, it becomes imperative to grasp the optimal strategies for ensuring the security of WordPress websites. Safeguarding WordPress sites is essential in today's digital landscape to safeguard sensitive data, uphold user confidence, and thwart cyber threats. In the study referenced by [7], an analysis of WordPress security challenges is provided, alongside suggested measures to mitigate these risks.

Understanding the importance of website security [8] emphasize the need to implement intelligent algorithms for securing WordPress sites. In a study conducted by [9], it was emphasized that a secured WordPress website development should start with the selection of reliable hosting providers that offer regular security updates and backups. The authors also highlighted the importance of using strong passwords and enforcing regular password changes to mitigate the risk of unauthorized access.Furthermore, the research carried out by[10] emphasized the significance of keeping all WordPress plugins and themes up to date. Outdated plugins and themes are often targeted by hackers to exploit vulnerabilities. The study suggested using reputable plugins and themes from trusted sources and frequently checking for updates to address any security vulnerabilities promptly.

According to [11], it was highlighted the importance of implementing security measures such as two-factor authentication and SSL certificates. Two-factor authentication adds a layer of security by requiring users to provide two forms of verification before accessing the website's admin panel. SSL certificates, on the other hand, encrypt the data transmitted between the website and its users, thereby protecting it from interception.A study by [2] underscored the significance of regular backups to ensure the recovery of a WordPress website in case of an attack. Designing a strong and reliable backup application that includes offsite storage and automated backups can help mitigate the potential damage caused by security breaches. It was opined by[13] that the importance of vulnerability analysis in WordPress websites. They identified common weaknesses such as outdated plugins, weak passwords, and insecure file permissions. Regular updates, strong authentication mechanisms, and strict file permissions were recommended as effective measures to mitigate vulnerabilities.To enhance user authentication in WordPress, [14] discussed the incorporation of two-factor authentication (2FA), emphasizing its role as an additional security measure. 2FA mandates users to furnish a secondary form of verification, like a one-time password or biometric data, thereby diminishing the likelihood of unauthorized entry into WordPress websites

In the work of[15], the significance of opting for reputable plugins and themes from reliable sources was underscored. It was emphasized that regular updates are essential to promptly address security vulnerabilities. By ensuring plugins and themes are consistently updated, website owners can effectively reduce the risk of exploitation by malicious actors

In trying to analyse how secured coding practices play a vital role in developing secure WordPress websites, [16] discussed the importance of following secure coding guidelines and frameworks. They highlighted the significance of input validation, output sanitization, and proper handling of user data to prevent common security vulnerabilities such as cross-site scripting (XSS) and SQL injection attacks. It was proposed by[17]that the advantages of systematic backups for WordPress websites. Their research recommended a robust backup strategy, including offsite storage and automated backups, to ensure

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 05 May 2024, pp: 593-603 www.ijaem.net ISSN: 2395-5252

the ability to restore websites in case of security breaches or data loss.

In the study carried out by[18], they stressed the need for continuous monitoring of logs, network traffic, and user activities. They also highlighted the importance of establishing an incident response plan to address and mitigate security incidents promptly. According to [19], an in-depth study of the security issues inherent in WordPress and provide insights into effective countermeasures. This understanding lays the foundation for implementing robust security measures. In the research conducted by [20], they focused on vulnerability analysis of WordPress websites. They identified common security weaknesses such as outdated plugins, weak passwords, and insecure file permissions. The study emphasized the need for regular updates, strong authentication mechanisms, and strict file permissions to mitigate vulnerabilities. It was opined by[21] the significance of selecting reliable hosting providers that offer regular security updates and backups. They also emphasized the importance of using strong passwords and enforcing regular password changes to prevent unauthorized access.

Identifying and addressing common security threats is crucial in securing WordPress websites was carried out by[22] with a comprehensive study on the vulnerabilities of popular WordPress security plugins. Their research sheds light on the potential risks associated with these plugins and highlights the importance of thorough vulnerability assessments. In the work of [23], they emphasized the significance of implementing strong passwords and utilizing security plugins that restrict the number of login attempts. These measures can effectively mitigate the risk of brute-force attacks. Moreover, [24] propose the use of machine learning algorithms to identify and check such attacks, offering an advanced security approach.

In conclusion, brute force attacks, cross-site scripting (XSS), SQL injection, and outdated software versions pose significant risks to the security and integrity of WordPress websites. However, through the implementation of preventive measures such as strong passwords, security plugins, input validation, output sanitization, parameterized queries, and regular updates, WordPress website owners can strengthen the security posture of their websites and mitigate these threats effectively.

III. METHODOLOGY ADOPTED

The methodology employed in this study involves Object-Oriented Analysis (OOA), which systematically breaks down the process into structured phases. It establishes connections and hierarchies among classes while integrating both information and behaviour within each class. This architectural approach provides adaptability and efficient processing of attendance data by employing polymorphism and inheritance. Visual aids such as use case diagrams and sequence diagrams aid in visualizing interactions and behaviours.

Choosing the preferred Content Management System (CMS) is a pivotal aspect of this methodology, with WordPress being the primary choice for website creation and management. This decision is aligned with WordPress's widespread popularity, user-friendly interface, and extensive array of plugins and themes. The selection of WordPress significantly influences various stages of the methodology, including design, customization, and content publication. Leveraging WordPress offers flexibility and accessibility, catering to diverse website purposes. Moreover, the platform's robust community support and regular updates contribute to the overall efficacy of the chosen methodology.

IV. ANALYSIS OF THE SYSTEM

In this comprehensive analysis of the system, there is a steadfast commitment to fortifying user access controls and authentication mechanisms, addressing critical aspects of securing WordPress websites. The meticulous examination of existing literature emphasizes the need for robust measures, ensuring that only authorized individuals can access sensitive data and execute administrative tasks. The focus on user access controls includes a nuanced exploration of granular access controls and role-based access control (RBAC) frameworks, allowing tailored access levels based on users' roles and responsibilities.

Beyond traditional password-based methods, the analysis advocates for stronger authentication mechanisms like two-factor authentication (2FA) and biometric authentication to counter vulnerabilities associated with password-centric systems. The incorporation of 2FA gives additional layer of verification, while biometric authentication, leveraging fingerprint or facial recognition, offers a more secure and convenient user verification method.

Best practices for enhancing user access controls and authentication mechanisms in

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 05 May 2024, pp: 593-603 www.ijaem.net ISSN: 2395-5252

WordPress development are outlined, encompassing strong password policies, rate limiting, account lockouts, regular user account audits, and the utilization of security plugins. The emphasis on regular software updates underscores the critical importance of keeping WordPress core, plugins, and themes up to date to promptly address security vulnerabilities.

Transitioning to secure coding practices and data security, the analysis underscores the pivotal role of secure coding in mitigating common security vulnerabilities, including protection against Cross-Site Scripting (XSS), SQL Injection, and Denial of Service (DoS) attacks. Key practices include input validation, secure file handling, escaping and output filtering, and leveraging security-oriented functions from the WordPress API. Emphasizing the importance of consistent code evaluations and testing is crucial for detecting and addressing security vulnerabilities at an early stage of the development lifecycle.

The approach to securing data in WordPress involves encryption, access control, database security, vulnerability patching, and implementing secure authentication and authorization mechanisms. The analysis stresses the imperative nature of these measures to protect sensitive information from unauthorized access.

Moving forward, the analysis delves into the significance of regular updates, backups, and disaster recovery planning. Regular updates of WordPress central, themes, and their plugins are emphasized to promptly address security vulnerabilities, while effective backup strategies and disaster recovery planning are highlighted as essential safeguards against data loss and system failures.

The analysis culminates with an exploration of the effective use of security plugins and tools, encompassing their selection, configuration, and deployment. This holistic approach, integrating best practices and recommended measures, serves to fortify the security, stability, and resilience of WordPress websites against a spectrum of threats, including XSS, SQL Injection, and DoS attacks. Ultimately, this comprehensive strategy fosters user trust by providing a secure environment for website owners and administrators.

V. THE ARCHITECTURE OF THE SYSTEM

The models come in a variety of complementary forms and formats, such as mathematical equations, graphs, and quantitative data, as well as pictures and diagrams.

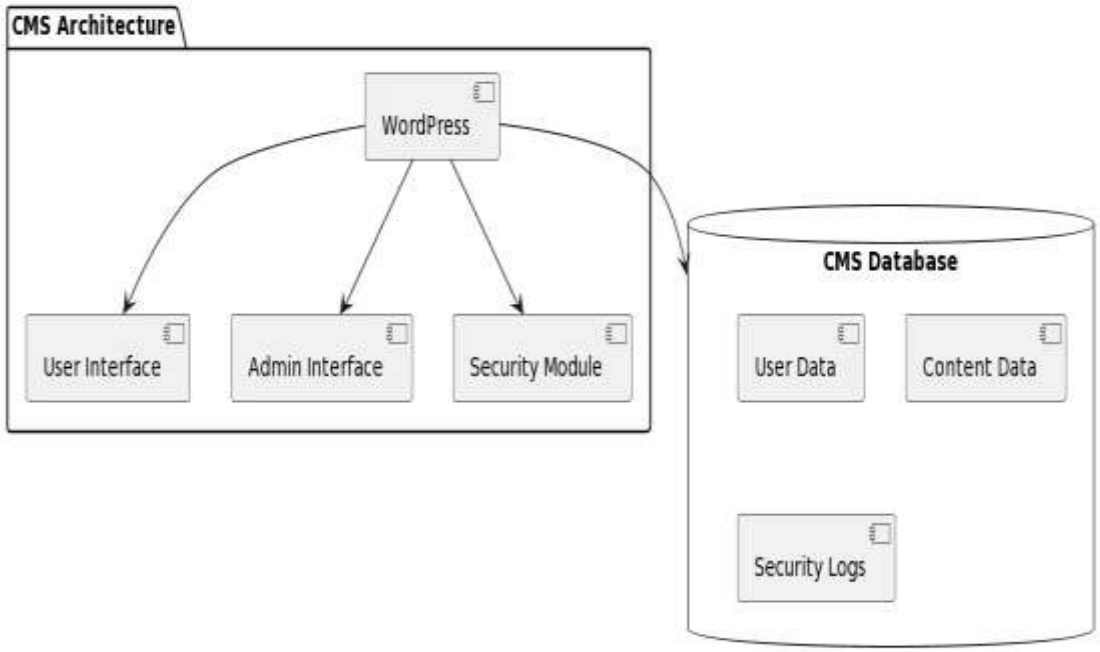

Figure 1: The Architecture of the System

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 05 May 2024, pp: 593-603 www.ijaem.net ISSN: 2395-5252

These are the elements that make up the architecture;

5.1. WordPress (CMS)

At the core of the architecture is WordPress, serving as the Content Management System. WordPress is a highly extensible and widely used CMS, known for its flexibility and ease of use. It acts as the central hub for content creation, management, and presentation, offering features such as posts, pages, themes, and plugins.

5.2. User Interface and Admin Interface

The architecture includes dedicated interfaces for both users and administrators. The User Interface provides a user-friendly platform for content consumption, while the Admin Interface offers advanced functionalities for content creation, editing, and site management.

5.3. Security Module

A crucial aspect of the system is the Security Module, designed to enhance the overall security of the CMS. The Security Module incorporates features such as Brute Force Protection, SQL Injection Prevention, XSS Mitigation, and Strong Password Authentication. These security measures are essential to safeguard against common web vulnerabilities and protect the CMS from malicious activities.

5.4. CMS Database

The CMS Database is represented as a cylindrical structure, emphasizing its role as the central repository for storing critical data. It includes tables for User Data, Content Data, and Security Logs. User Data may store information like usernames, passwords, and roles, while Content Data includes details about posts and pages. Security Logs capture information related to security events and activities.

VI. SYSTEM IMPLEMENTATION

The implementation phase has to do with implementing the design outlines into a functional WordPress website. Leveraging the WordPress content management system, the implementation includes:

6.1. System Requirements

6.1.1. Hardware Requirements

i. Web Server: Apache or Nginx
ii. Database Server: MySQL and PhpMyAdmin
iii. Storage: SSD for improved performance
iv. RAM: 8GB or higher for efficient multitasking
v. Processor: core i5 or higher for faster processing

6.1.2. Software Requirements

i. Operating System: Linux-based (e.g., Ubuntu)
ii. Web Server Software: Apache or Nginx
iii. Database Management System: MySQL
iv. PHP: Version 7.3 or later
v. WordPress: Latest stable version

VII. PROGRAM DEVELOPMENT

The program development phase involves the main coding and creation of the WordPress website considering the specifications of design outline. The choice of a programming environment plays a crucial role in shaping the development process and ensuring the efficiency and effectiveness of the implemented system.

VIII. CHOICE OF PROGRAMMING ENVIRONMENT:

The programming environment for developing the WordPress website is primarily centered around the WordPress ecosystem, which is built on PHP, a server-side scripting language. The primary components of the programming environment include:

8.1.PHP

WordPress relies heavily on PHP for server-side scripting. PHP scripts handle dynamic content generation, database interactions, and other server-side functionalities. The selected PHP version aligns with the WordPress recommendations for compatibility and performance.

8.2. HTML, CSS, and JavaScript

Complementary to PHP, HTML, CSS, and JavaScript are utilized for creating the front end of the website. HTML structures the content, CSS styles the layout, and JavaScript enhances user interactivity. These technologies ensure a responsive and visually appealing user interface.

8.3. MySQL Database

As the chosen database management system, MySQL is integral to storing and retrieving data for the WordPress website. SQL queries written in PHP interact with the MySQL database to manage posts, user information, and other

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 05 May 2024, pp: 593-603 www.ijaem.net ISSN: 2395-5252

relevant data posts, user information, and other relevant data.

IX. MODULE SPECIFICATION

Module specification involves providing detailed documentation for each module in a software system, outlining its purpose, functionalities, inputs, outputs, and interactions with other modules.

The following are some of the modules present in the system;

9.1. The Logon Module: This module is used to logon to the system by the registered users in other to gain access to the system.

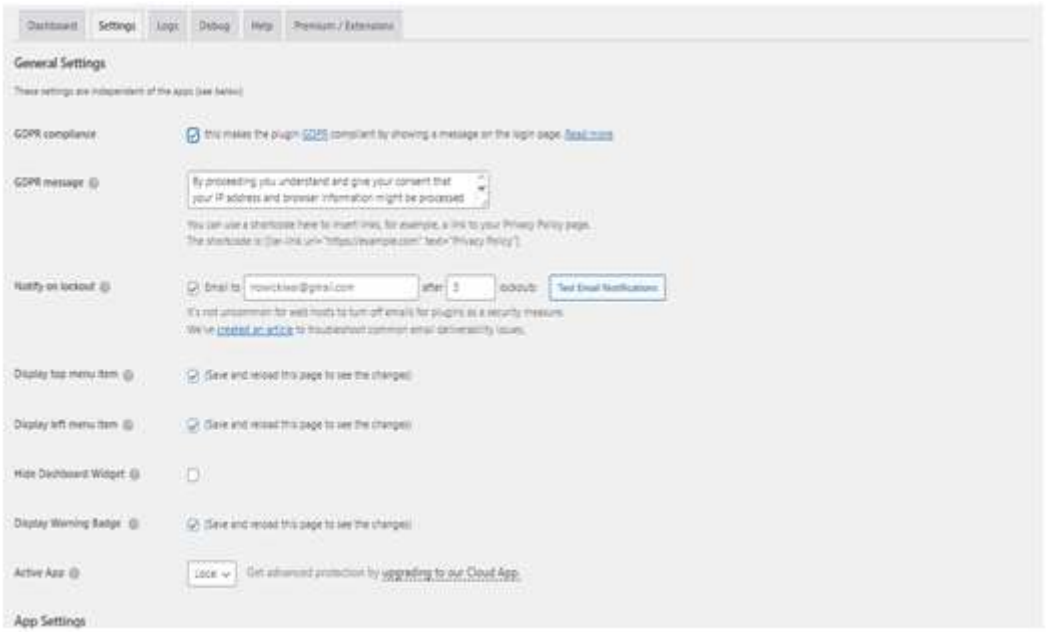

Figure 2: Logon Module

9.2. The Training Module: This module is used by the administrator for training the system which in turn enables him to manage the entire system

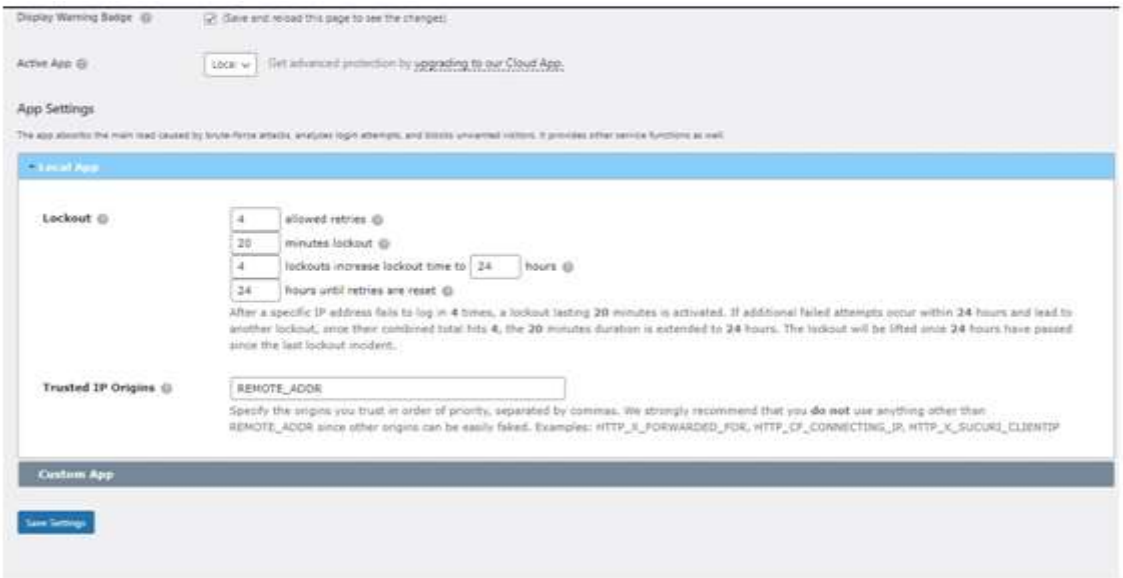

Figure 3: The Training Module

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 05 May 2024, pp: 593-603 www.ijaem.net ISSN: 2395-5252

X. OUTPUT DESIGN FORMAT

The output design of any program talks about the detailed reports to be generated through inputs made by users. In these projects, the output design will be displayed using reports. The administrator of the system will generate the following reports.

10.1. Fault Report: A fault report serves as a critical communication tool for users, testers, and developers to document and address issues within a software system. When a user or tester encounters unexpected behavior, malfunctions, or errors in the system, a detailed fault report is essential to convey the problem accurately. This initial paragraph introduces the purpose of the fault report, emphasizing its role in facilitating effective communication between stakeholders and aiding in the resolution of identified issues.

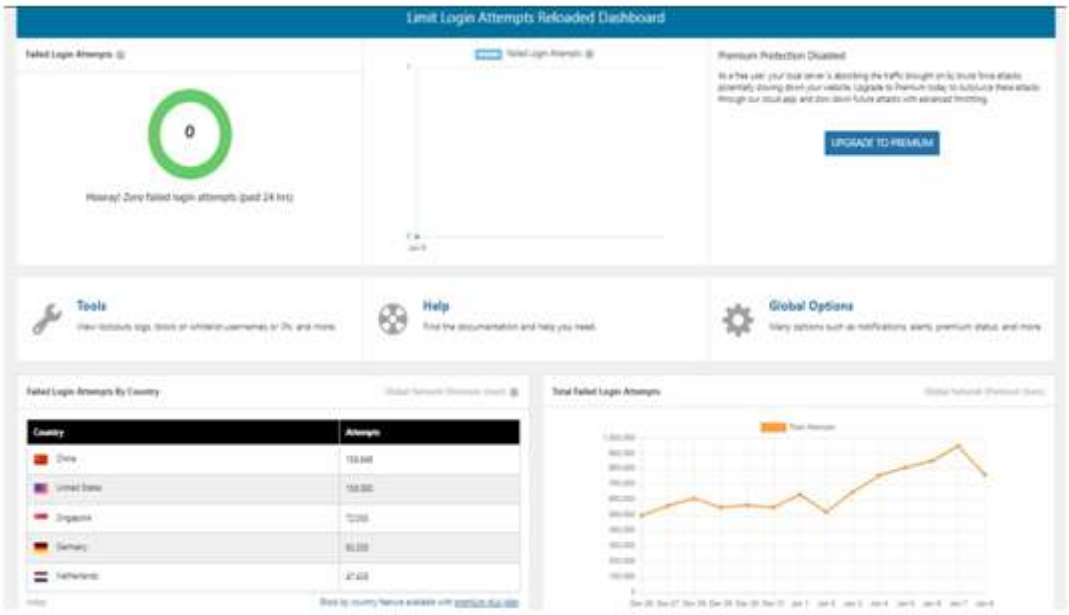

Figure 4: Fault Detection Report

10.2. Database Specification

The Database Specification provides a comprehensive overview of the system's database architecture, defining data tables, relationships, and constraints. It guides database administrators in ensuring optimal data organization, integrity, and performance. This section is crucial for developers and administrators involved in database implementation and maintenance.

Table 1. Expected Test Result

Security Aspect	Actual Test Done	Expected Result
DDoS Protection	Conducted stress tests and simulated DoS attacks to assess the CMS's ability to handle high-traffic loads. Implemented DDoS mitigation measures.	Expected the CMS to resist and mitigate DoS attacks effectively, maintaining availability and performance under stress
SQL Injection Prevention	Attempted SQL instill attacks by injecting malicious SQL queries into input fields. Implemented parameterized queries and input validation to check SQL injection vulnerabilities.	The Plugins installed within the CMS are expected to successfully block SQL injection attempts, ensuring that user inputs are sanitized and database interactions are secure.
Cross-Site Scripting (XSS)	Executed various XSS attack scenarios by injecting malicious scripts into user inputs. Implemented output encoding and validation	It is expected of the CMS to prevent XSS attacks by sanitizing and encoding user inputs, ensuring that no malicious

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 05 May 2024, pp: 593-603 www.ijaem.net ISSN: 2395-5252

	to prevent XSS vulnerabilities.	scripts are executed on the client side.
Authentication Security	Conducted penetration testing to identify vulnerabilities in the authentication system. Implemented secure password storage, multi-factor authentication, and account lockout policies.	Expected the CMS to have robust authentication mechanisms, securely storing passwords, enforcing strong password policies, and protecting against unauthorized access through multi-factor authentication.
Weak Password Assessment	Evaluated the system's response to weak password inputs, assessing its strength policies.	Expected the CMS to reject weak passwords effectively, enhancing overall authentication security.

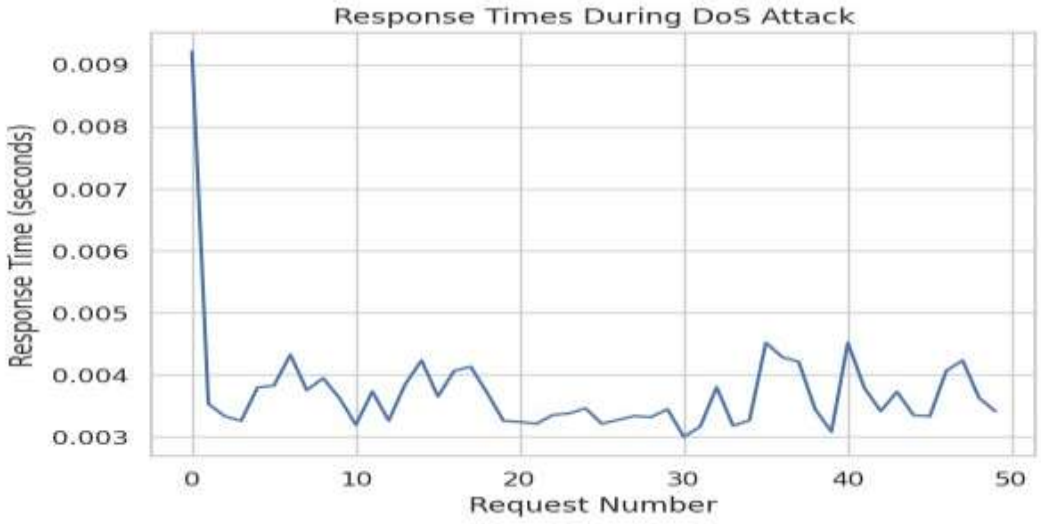

Figure 5: Response time for DoS attack

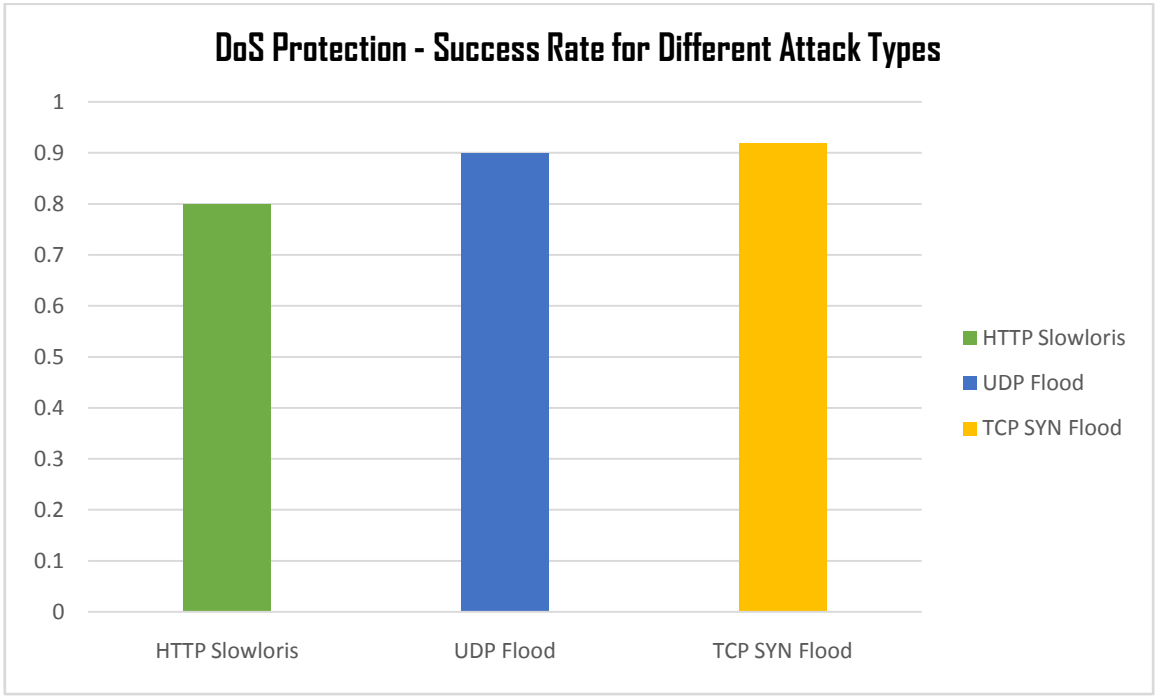

Figure 6:DoS Protection - Success Rate for Different Attack Types

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 05 May 2024, pp: 593-603 www.ijaem.net ISSN: 2395-5252

Our WordPress Content Management System underwent a rigorous security assessment using Kali Linux. The tests demonstrated the system's robust mitigation measures, effectively countering attacks and ensuring optimal performance under stress. The findings will guide future security enhancements, emphasizing continuous monitoring and proactive measures to strengthen resilience against evolving threats.

XI. SYSTEM SECURITY

11.1. Password Protection

In the system security, a robust password protection system has been implemented to mitigate the risk of brute force or dictionary attacks. Users are required to adhere to stringent password creation policies, ensuring a formidable defence against unauthorized access attempts. The system mandates the use of complex passwords, combining alphanumeric characters and special symbols, and enforces regular password updates. These measures collectively aim to render basic force and other attacks ineffective, fortifying the system against unauthorized intrusion attempts.

11.2. Authentication

The system incorporates an authentication mechanism, emphasizing two-factor authentication for user verification. This additional layer of security enhances the authentication process beyond conventional username and password requirements. Users undergo secondary verification through methods such as one-time passcodes sent to registered email addresses or mobile devices.

11.3. Cryptography.

Within the system security framework, cryptography plays a pivotal role in securing sensitive data stored in the database. Robust encryption algorithms are implemented to safeguard confidential user information. This cryptographic approach ensures data confidentiality, making unauthorized access ineffective without the appropriate decryption keys. Moreover, it aligns with industry standards and regulatory requirements, underscoring the commitment to data protection.

XII. SUMMARY

This research delves into the best practices for developing a secure WordPress website. WordPress is a popular content management system (CMS) that powers a significant portion of websites worldwide. However, its widespread usage also makes it an attractive target for hackers and malicious actors. Hence, ensuring the security of WordPress websites is of paramount importance. The research begins by outlining the common vulnerabilities and security risks associated with WordPress websites, including outdated software, weak passwords, insecure plugins and themes, and improper file permissions. It highlights the potential consequences of neglecting security measures, such as data breaches, website defacement, and compromised user information.

To mitigate these risks, the study offers a detailed set of best practices for secure WordPress website development. These practices encompass various stages of website creation, including pre-development, development, and post-development phases. The research emphasizes the importance of maintaining an ongoing security posture for WordPress websites and encourages developers to adopt a proactive approach by regularly reviewing and updating security practices. By implementing the recommended best practices outlined in this research, developers can significantly enhance the security of WordPress websites and safeguard sensitive data and user privacy

XIII. CONCLUSION

This research provides essential insights into secure WordPress website development. Given the widespread use of WordPress, implementing effective security measures is crucial. The study identifies common vulnerabilities such as outdated software, weak passwords, insecure plugins, and improper file permissions, emphasizing the potential risks like data breaches. Practical recommendations cover pre-development, development, and post-development phases, including updating core components, using strong passwords, and implementing secure coding practices. Adhering to these best practices enhances website security and protects sensitive data. A proactive security approach, staying informed about threats and regularly updating practices, is vital. The research is a valuable resource for developers and administrators, contributing to a safer online environment. Continuous efforts in implementing secure practices ensure the longevity and integrity of WordPress websites, ultimately empowering developers to create a secure online ecosystem.

XIV. RECOMMENDATIONS

The study recommends a multifaceted approach to enhance secure web content management systems within WordPress. Collaboration among cryptography experts, web

developers, and cybersecurity professionals is urged to refine dynamic content encryption models. User-centric security is advocated through comprehensive studies, enabling the fine-tuning of authentication mechanisms based on user behaviour. Real-world application studies are deemed essential for assessing the viability and performance of security measures. Scalability considerations, comparative analyses of encryption frameworks, and the promotion of adaptive authentication policies are highlighted for robust system design. The continuous refinement of security measures and the dissemination of practical insights aims to foster a broader adoption of secure practices within the WordPress community.

REFERENCES

[1]. Maraga, A. Awuor, F. M. and Ogalo, J. (2022). Model for Security Controls in Web Content Management System, Journal of Internet and Information Systems, 11(1), 1-12. https://doi.org/10.5897/JIIS2021.0120

[2]. Huang, Z. Lie, D. Tan, G. and Jaeger, T. (2019). Using Safety Properties to Generate Vulnerability Patches. In Proceedings of the 40th IEEE Symposium on Security and Privacy (S&P), pages 539–554. IEEE.

[3]. Piper B, Jepkemei E, Kwayumba D, Kibukho K (2015). Kenya's ICT Policy in Practice: The Effectiveness of Tablets and E-Readers in Improving Student Outcomes. In FIRE: Forum for International Research in Education 2(1):3-18. Lehigh University Library and Technology Services. 8A East Packer Avenue, Fairchild Martindale Library Room 514, Bethlehem, PA 18015.

[4]. Mahfouzi, R. A. R. (2021). Linköping University, Department of Computer and Information Science. Sane.

[5]. Li, S. Kang, M., Hou, J. and Cao, Y. (2022). Mining Node and Vulnerabilities Via Object Dependence Graph and Query, Proceedings of the 31st USENIX Security Symposium (USENIX Security).

[6]. Kadir, M. Z. Z. (2021). Risk Assessment of Web Application Penetration Testing on Cross-Site Request Forgery (CSRF) Attacks and Server-side Includes (SSI) Injections, 2021 International Conference on Data Science and Its Applications (ICoDSA).

[7]. Ajayi, A. O. and Eyo, E. (2017). Strengthening User Authentication in WordPress: A Two-Factor Authentication Approach. International Journal of Computer Science and Information Security, 15(2), 95-102.

[8]. Nagendran, K. (2020). Web Application Firewall Evasion Techniques, 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS).

[9]. Ahmed, N. Alshaikh, M. and Malik, S. (2020). A Comparative Analysis of WordPress Website Security. In Proceedings of the 2020 International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 1227-1230).

[10]. Adekile, D. O. and Adegboye, O. S. (2018). Enhancing Security in WordPress Website Development. International Journal of Innovative Technology and Exploring Engineering, 8(9S2), 204-208.

[11]. Nagendran, K. (2020). Web Application Firewall Evasion Techniques, 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS).

[12]. Ogundele, O. J. andOlabode, O. O. (2019). Ensuring Data Protection in WordPress Websites: A Backup Strategy. Journal of Information Engineering and Applications, 9(7), 31-42.

[13]. Nielsen, B. Hassanshahi, B. and Gauthier, F. (2019). Nodest: Feedback-Driven Static Analysis of Node.js Applications, In Proceedings of the 27th Joint Meeting on Foundations of Software Engineering (FSE).

[14]. Pan, X. Cao, Y. Liu, S. Zhou, Y. Chen, Y. and Zhou, T. (2016). CSPAutoGen: Black-Box Enforcement of Content Security Policy upon Real-World Websites. In Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security (CCS).

[15]. Aransiola, J. O (2019). Analysis of Security Issues in WordPress Plugins and Themes. Journal of Advances in Computer Engineering and Technology, 5(3), 123-135.

[16]. Singh, N. (2020). Automated versus Manual Approach of Web Application Penetration Testing. 2020 11th International Conference on Computing.

[17]. Ogundele, O. J. and Olabode, O. O. (2019). Ensuring Data Protection in WordPress Websites: A Backup Strategy. Journal of Information Engineering and Applications, 9(7), 31-42.
[18]. Erez, G. Mubina, M. Trisha, P. (2014). Database Encryption Architectures, International Journal of Scientific & Engineering Research, Volume 7, Issue 12, 313 ISSN 2229-5518.
[19]. Shebli, H. M. Z. A. (May 2018). A study on penetration testing process and tools. 2018 IEEE Long Island Systems, Applications and Technology Conference (LISAT).
[20]. Wen, M. Chen, J. Wu, R. Hao, D. and Cheung, S.C. (2018). Context-aware Patch Generation for Better Automated Program Repair. In Proceedings of the 40th International Conference on Software Engineering (ICSE). IEEE.
[21]. Adekile, D. O. and Adegboye, O. S. (2018). Enhancing Security in WordPress Website Development. International Journal of Innovative Technology and Exploring Engineering, 8(9S2), 204-208.
[22]. Chen, Y. and Liginlal, D. (2017). Bayesian Networks for Knowledge-Based Authentication. IEEE Transactions on Knowledge and Data Engineering, 19 (5), pp.695-710.
[23]. Albalawi, U. (2018). Countermeasure of Statistical Inference in Database Security, Proc. IEEE Int. Conf. Big Data (Big Data), Dec. 2018, pp. 2044_2047, doi: 10.1109/BigData.2018.8622241Yang, B. (2020). Vulnerability Assessments of Electric Drive Systems Due to Sensor Data Integrity Attacks. IEEE Transactions on Industrial Informatics, Volume 6.
[24]. Youkun, S. (2022). Backporting Security Patches of Web Applications: A Prototype Design and Implementation on Injection Vulnerability Patches. 31st USENIX Security Symposium.