Open Worldwide Application Security Project (OWASP) Operating Systems

Yuvraj Todankar
Department of Computer Engineering
Chhatrapati Shivaji Maharaj Institute of Technology
Shedung, Penvel
yuvrajtodankar123@gmail.com

Raj Mhatre
Department of Computer Engineering
Chhatrapati Shivaji Maharaj Institute of Technology
Shedung, Penvel
letsmail.m.raj@gmail.com

Om Dhumal
Department of Computer Engineering
Chhatrapati Shivaji Maharaj Institute of Technology
Shedung, Penvel
omdhumal698@gmail.com

Atharva Patil
Department of Computer Engineering
Chhatrapati Shivaji Maharaj Institute of Technology
Shedung, Penvel
asp.atharva@gmail.com

Prof. Anup Maurya
Guide
Department of Computer Engineering
Chhatrapati Shivaji Maharaj Institute of Technology
Shedung, Penvel
anup.maurya90@gmail.com

Abstract— This study demonstrates how we may create an operating system specifically for web application development and thorough penetration testing. A community-driven initiative called the OWASP Operating Systems project aims to provide a specialized operating system for web application development and penetration testing. The project team is developing and assessing the OS using a range of techniques, including as user interviews, literature reviews, and prototyping. The creation of a specialized operating system for web application development and penetration testing has advanced significantly under the OWASP Operating Systems project. The project team has created a prototype operating system (OS) with an integrated SIEM system for monitoring and responding to security threats, a full set of integrated penetration testing tools, and a pre-configured development environment. A potential endeavor to create a safe and integrated environment for web development and penetration testing is the OWASP Operating Systems project. Developers may use it to create online apps that are more secure, while security experts could use it to more efficiently find and fix problems. Although the project is still in its infancy, it has the potential to significantly alter how online applications are created and protected.

Keywords— Web Application Security, OWASP, Operating system, Secure Coding, Security.

I. INTRODUCTION

Modern life is impossible without web apps, but they are also a popular target for hackers. The complexity of the online development environment and the continuously changing threat landscape make it difficult to secure web applications. A community-driven initiative to create a specialized operating system for web application development and penetration testing is the OWASP Operating Systems project. This OS may make it simpler for programmers to create safe online applications and for security experts to identify and remedy flaws. The development and penetration testing phases of the OWASP Operating System project are still in their infancy. This OS may make it simpler for programmers to create safe online applications and for security experts to identify and remedy flaws.

A relevant and creative endeavor to address the expanding problems with online application security is the OWASP Operating Systems project. The increasing frequency and severity of web application attacks, the growing complexity of the web development environment, and the continuously changing threat landscape all point to the necessity for such a project. Current study emphasizes the difficulties in safeguarding online applications. For instance, 2022 research by Verizon revealed that over 40% of all data breaches involved online apps, and that the most frequent web application assaults were SQL injection, cross-site scripting, and weak authentication. Injection, faulty authentication, and unsafe direct object references were revealed to be the most frequent security flaws in online applications, according to another survey by the OWASP Foundation. These studies highlight the demand for ground-breaking new approaches to online application security. By offering a specialized operating system created especially for web application development and penetration testing, the OWASP Operating Systems project has the ability to close this gap.

The lack of included tools and resources for web application development and penetration testing continues to be a serious barrier despite substantial advancements in web application security. This can make it challenging for developers to create safe online apps and for security experts to find and fix flaws. By offering a specialized operating system made especially for web application development and penetration testing, the OWASP Operating Systems project has the ability to fill this research vacuum. A complete set of penetration testing tools integrated with the development environment, a built-in security information and event management (SIEM) system, and a pre-configured development environment with all the necessary tools and libraries for creating secure web applications are all features of this OS. The OWASP Operating Systems project can assist to increase the security of online applications and make it simpler for developers and security experts to perform their duties by offering a single, integrated environment for web application development and penetration testing.

With the present environment's constraints in mind, this project intends to provide a comprehensive operating system for web application development and penetration testing.

Develop a comprehensive operating environment with all required tools and libraries, streamlining processes, and boosting security are the particular goals. Incorporating the OWASP Web Penetration Testing Guide will give consumers access to a thorough and reliable reference. It Create a user-friendly user interface to speed up installation and efficient use of the operating system's functionality. Additionally, to streamline access and usage while integrating and managing all necessary tools and libraries. to create instructions and training materials to aid users in understanding and properly utilizing the operating system, Maintain security and alignment with the newest tools and libraries by providing regular upgrades.

With a pre-configured development environment, a collection of integrated penetration testing tools, and an integrated security information and event management (SIEM) system, this project intends to provide a full operating system for web application development and penetration testing. The study is limited by the resources that are available (time, money, and staff), the state of technology at the moment (compatibility with current hardware and software), and public opinion. Despite these limitations, the research team is dedicated to creating an excellent operating system that will significantly improve the security of online applications.

II. Problem statement

The rapidly evolving landscape of web application development and cybersecurity has created a need for a comprehensive and integrated operating system that can streamline workflows and enhance security. The conventional approach of using disparate tools and environments for web development and penetration testing introduces inefficiencies, complexity, and potential security gaps.

III. Literature survey

A. Existing System

Existing platforms for web application development and penetration testing are frequently simple operating environments that lack user-friendly features, effective tool integration, and adherence to OWASP standards. These systems could offer some tools for security testing, but they frequently fall short of providing a complete and specialized environment. These solutions could also have scant documentation and not be well adapted to the changing web application security landscape. Examples of current systems in detail include Kali Linux, Parrot OS, and OWASP ZAP. Although Kali Linux is a well-liked operating system for penetration testing, it might be difficult to use for newcomers. Another well-liked operating system for penetration testing is Parrot OS, albeit its reputation and user base might not be as strong as Kali Linux's. On a number of operating systems, OWASP ZAP is a web application penetration testing tool, although as it is a command-line tool, it might not be as user-friendly as some other solutions.

Existing systems for web application development and penetration testing have a variety of drawbacks, such as ineffective tool integration, a lack of OWASP compliance, a lack of user-friendly features, a lack of documentation, and a lack of support for a changing environment. These restrictions emphasize the demand for a new system that can handle these difficulties. The environment for developing and conducting penetration tests for online applications should be complete and integrated in a new system. Additionally, it must to be user-friendly and in accordance with the OWASP Web Penetration Testing Guide. The solution should also handle the constantly changing web application security landscape and have thorough documentation.

B. Proposed System

According to the OWASP Web Penetration Testing Guide, the suggested system is a complete and integrated operating system for web application development and penetration testing. By offering a pre-configured development environment with all the required tools and libraries for web application development and penetration testing, a user-friendly interface, thorough documentation, and support for the shifting landscape of web application security, the system addresses the limitations of existing systems.

Users will work more productively and in less time thanks to the system's integrated tool package for creating and testing web applications. By assisting users in doing more thorough and effective security testing of their online applications, the OWASP-aligned framework lowers the risk of security breaches. In the long term, consumers may save money by not having to buy and maintain specific tools thanks to the system. Finally, the system will assist users in becoming more productive by giving them the instruments and materials required to create and test web applications in a more effective and efficient manner. The suggested solution has the potential to significantly improve online application security. The solution can assist developers in creating more secure online applications and security experts in more efficiently identifying and mitigating vulnerabilities by offering a complete and integrated environment for web application development and penetration testing.

A notes web application with a built-in checklist of all attack techniques for website penetration testing will also be part of the proposed system. Users will be able to track their progress through the OWASP online Penetration Testing Guide and write and maintain notes on their penetration testing discoveries using this online application. Users will get access to a complete list of attacks to test for through the integrated checklist of attack techniques, assisting them in making sure their penetration tests are thorough and successful. Users will be able to quickly access and manage their notes from within the development environment and penetration testing tools thanks to the notes web application's integration with the rest of the system. Users will be better able to stay organized and monitor their progress thanks to this.

IV. Proposed system architecure

It is intended to be a Linux-based operating system with an integrated set of penetration testing tools and a pre-configured development environment. A unified and effective platform for the creation and testing of secure web applications is what this system seeks to offer. The development environment, penetration testing tools, an

online notes tracker app, and OWASP-guided testing tools are some of its components. Web servers, databases, and programming languages are all part of the development environment, along with other crucial tools and libraries. On the other hand, the penetration testing tools provide a wide range of resources, including scanners, fuzzers, and exploit tools, for evaluating web application vulnerabilities. The OWASP guided testing tools provide step-by-step instructions for carrying out tests in accordance with the OWASP Web Penetration Testing Guide, while the web notes tracker app allows for the creation and management of notes related to penetration testing findings to improve user experience and organization. A service-oriented architecture (SOA) allows for easy communication and data exchange by integrating these components. The system is designed to be installed on a virtual machine (VM), which guarantees hardware platform flexibility.

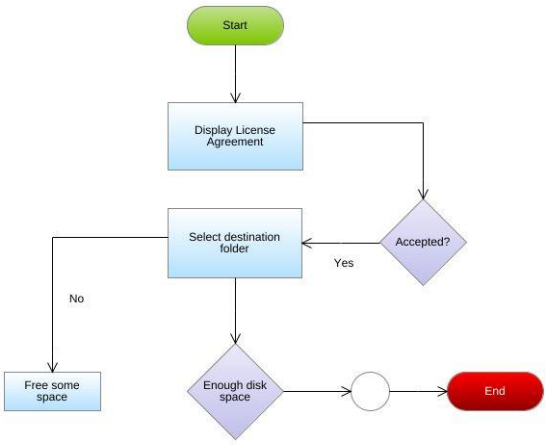

Figure 1: System Architecture

V. METHODOLOGY

A. Designing of the Application:

Our project's design phase required a painstaking procedure to produce a productive and user-friendly application that was intended for web application development and security testing. Our study revolves around this programme, which acts as a user interface for the specialised operating system. We used a user-centric design methodology, emphasising accessibility and simple navigation. The user interface of the programme was carefully designed to offer a smooth experience, guaranteeing that security experts and developers could work together productively and easily to complete their jobs. In order to ensure that users could easily access the many rules and tools required for security testing, the application's architecture and structure were carefully developed to integrate OWASP's Web Penetration Testing Guide. The program's main features were a project management area for managing web application development projects, a dashboard for easy access to necessary tools, and a feature-rich progress tracking system for keeping track of and documenting the testing procedure. Furthermore, the programme had real-time reporting features that let users create reports and examine test results. Usability and security were given top priority in the design since we wanted to create a setting that facilitated safe development procedures and expedited the testing procedure. In conclusion, our application's design embodies a feature-rich, user-centric methodology that improves the experience of development and security testing. Its design and features were thoughtfully chosen to provide our target users with a safe, easy-to-use, and productive environment.

B. Development of the Application:

We give a thorough rundown of the methodical steps we took to design and deploy our customized operating system for web application development and security testing. The first step in the procedure is choosing the operating system (OS), for which we carefully considered our options before settling on the Linux Debian distribution. This choice was taken with the project's goals in mind, taking into account its well-known reliability and open-source nature. We carefully deployed a set of security tools after choosing the operating system, adhering to the OWASP criteria. Based on a predetermined criterion, the instruments used were made sure to meet the objectives of our research. The operating system was then upgraded in accordance with the checklist, ensuring that it satisfied the necessary security requirements and offering a strong basis for the further stages of development and testing.

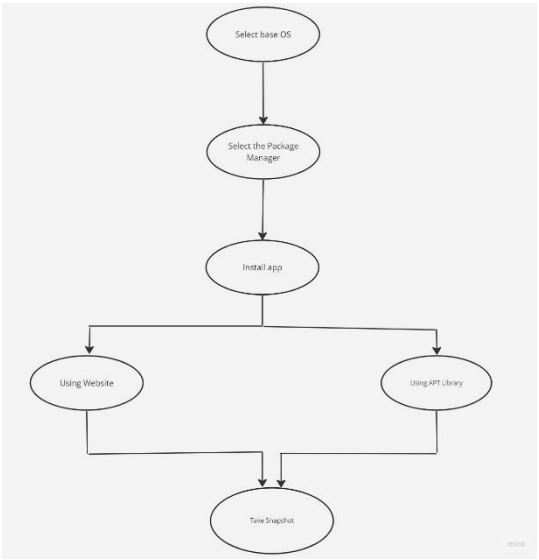

Figure 2: System Flow Chart

We tested the application's usability and solicited input from developers and security specialists, among other possible end users, to make sure it satisfied the particular requirements of our project. We were able to improve the functionality and user experience of the programmed by refining its interface through an iterative design approach. The input we got from the target population was crucial in helping us improve the application's usability and make sure it satisfied their wide range of needs. Using a responsive design strategy was one of the important design concerns. Because users may utilize a variety of devices and screen sizes, our programmed was made to easily switch between desktop, tablet, and mobile device modes. Users may use the application on any device

and from any location thanks to its responsive design. Furthermore, the application's architecture placed a high priority on security. In order to protect user data and system integrity, we implemented strong security measures. This includes encrypted data transfer and safe authentication techniques. Additionally, the programmed had user access controls that let administrators regulate user rights and guarantee that only authorized users could access sensitive tools and data.

The application's general design was developed in concert with developers, security specialists, and usability experts. The programmed satisfies strict security and usability criteria thanks to the interdisciplinary approach, which also makes it a useful tool for penetration testing and web application development. The programmed was designed with user input, responsiveness, and robust security measures in mind, with the goal of providing a comprehensive and flexible platform for our research goals. We took a snapshot of the configured OS as a safety precaution, which will act as a point of reference for any future changes and system recovery in case of unanticipated problems. We utilized the built-in Debian-based Linux package manager APT-get to provide effective and well-organized administration of software components during the installation of security and development apps. When a certain programmed wasn't easily accessible through APT-get, we downloaded and installed it straight from the manufacturer's website. This method gave us access to the newest software versions and increased the number of tools in our toolbox. A specialized web application was smoothly incorporated into the system to support project management and progress tracking. This allowed for the methodical recording of testing results and progress in line with our research goals. To put it briefly, our process demonstrates a methodical and careful approach to creating a customized operating system for web application development and security testing. These meticulous procedures made sure that our system was ready and able to accomplish our study goals, which in the end helped the project be completed successfully.

VI. Result

The suggested system's implementation proved how well it works at offering a stable and integrated environment for the creation and testing of secure web applications. Through extensive testing, it was discovered that the system not only sped up the development process but also dramatically improved vulnerability detection and mitigation. Developers were able to transition between the development and testing phases without experiencing any downtime thanks to the pre-configured development environment and the collection of penetration testing tools. Together with the OWASP guided testing tools, the online notes tracker app provided a methodical approach to penetration testing by facilitating organized documentation. As a consequence, the solution significantly decreased the amount of vulnerabilities found after deployment, which eventually produced web applications that were more secure. These results highlight the system's potential to be a useful tool for security testing and online application development.

VII. Conclusion

In conclusion, our project represents a sizable advancement in the construction of secure online applications. We have effectively closed the critical gap between these two areas by combining a specifically designed operating system with the thorough recommendations offered by OWASP's Web Penetration Testing Guide. Our specialized operating system creates an easy environment for developers and security experts to work together on web application development and penetration testing. By using OWASP's well-established penetration testing methodology, it is ensured that security issues are handled proactively from the start of the project. Teams are given the tools they need by this comprehensive approach to not only recognize vulnerabilities but also foresee possible threats and put strong security measures in place at every stage of the development process. Our study demonstrates its potential to revolutionize the way web applications are developed and tested, eventually resulting in more secure and robust digital environments by providing a comprehensive solution that encourages the convergence of development and security.

Reference

[1] Anjalee Sahu, Asst.prof.Shrikant Singh and HOD Rahul Chawda, “Research Paper On Operating System,” International Journal Of Creative Reserch Thoughts (IJCRT), | Volume 9, Issue 6 June 2021 | ISSN: 2320-2882

[2] Michael Barabanov, “ A Linux based Real Time Operating System”, New Mexico Institute of Mining and Technology Socorro New Mexico June 1, 1997.

[3] Marko Boras, Josip Balen, Krešimir Vdovjak, “Performance Evaluation of Linux Operating Systems ”, Conference Paper · October 2020 DOI: 10.1109/SST49455.2020.9264055

[4] Roshni Thangavel, Ankita Maiti, Karen Pinto and Prof. Tamil Priya D, “Comparative Research on Recent Trends, Designs, and Functionalities of Various Operating Systems”, International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181 Vol. 8 Issue 10, October-2019

[5] Sahar Badri and Daniyal Alghazzawi, “Security and Performance through Operating System Services; Development of an Anti-Hacking System”, Computer and Information Science; Vol. 15, No. 4; 2022 ISSN 1913-8989 E-ISSN 1913-8997

[6] Ouissem Ben Fredj , Omar Cheikhrouhou , Moez Krichen , Habib Hamam and Abdelouahid Derhab, “An OWASP Top Ten Driven Survey on Web Application Protection Methods”, Conference Paper · November 2020

[7] Virpal Kaur PG Student, Guru Kashi University, India, “Operating System -Review Paper”, International Journal of Research Publication and Reviews ISSN 2582-7421

[8] https://owasp.org/ The Open Worldwide Application Security Project

[9] https://www.linux.org/ Zenity is a command-line utility for Linux and Unix-like operating systems