I.J. Information Technology and Computer Science, **2024, 2, 45-56**
Published Online on April 8, 2024 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2024.02.04

Web Application Penetration Testing on Udayana University's OASE E-learning Platform Using Information System Security Assessment Framework (ISSAF) and Open Source Security Testing Methodology Manual (OSSTMM)

I Gusti Agung Surya Pramana Wijaya*
Dept of Information Technology, Faculty of Engineering, Udayana University, Bali, Indonesia
E-mail: pramanawijaya@student.unud.ac.id
ORCID iD: https://orcid.org/0009-0009-2445-8263
*Corresponding Author

Gusti Made Arya Sasmita
Dept of Information Technology, Faculty of Engineering, Udayana University, Bali, Indonesia
E-mail: aryasasmita@unud.ac.id

I Putu Agus Eka Pratama
Dept of Information Technology, Faculty of Engineering, Udayana University, Bali, Indonesia
E-mail: eka.pratama@unud.ac.id
ORCID iD: http://orcid.org/0000-0001-9574-4835

Received: 31 May 2023; Revised: 04 July 2023; Accepted: 17 September 2023; Published: 08 April 2024

Abstract: Education is a field that utilizes information technology to support academic and operational activities. One of the technologies widely used in the education sector is web-based applications. Web-based technologies are vulnerable to exploitation by attackers, which highlights the importance of ensuring strong security measures in web-based systems. As an educational organization, Udayana University utilizes a web-based application called OASE. OASE, being a web-based system, requires thorough security verification. Penetration testing is conducted to assess the security of OASE. This testing can be performed using the ISSAF and OSSTMM frameworks. The penetration testing based on the ISSAF framework consists of 9 steps, while the OSSTMM framework consists of 7 steps for assessment. The results of the OASE penetration testing revealed several system vulnerabilities. Throughout the ISSAF phases, only 4 vulnerabilities and 3 information-level vulnerabilities were identified in the final testing results of OASE. Recommendations for addressing these vulnerabilities are provided as follows. Implement a Web Application Firewall (WAF) to reduce the risk of common web attacks in the OASE web application. input and output validation to prevent the injection of malicious scripts addressing the stored XSS vulnerability. Update the server software regularly and directory permission checks to eliminate unnecessary information files and prevent unauthorized access. Configure a content security policy on the web server to ensure mitigation and prevent potential exploitation by attackers.

Index Terms: ISSAF, OSSTMM, RAV, STAR, Security Testing.

1. Introduction

The website is a versatile platform that serves various purposes in our daily lives. It is a collection of interconnected web pages that offer a wealth of information through text, images, audio, video, and files. Moreover, websites enable users to access and store data on web servers, granting them the convenience of retrieving information anytime and anywhere. As society progresses, websites have become increasingly significant, serving as valuable resources for seeking information, facilitating learning, building brand presence, and engaging in commercial activities such as buying and selling. Consequently, the number of websites has experienced a remarkable surge, showcasing their pervasive growth.[1]

However, as web-based applications continue to evolve rapidly, the significance of security testing cannot be overstated. With technological advancements, the number of security threats and attacks targeting web-based applications has also increased [2]. Unfortunately, security concerns are often not prioritized adequately, relegating them to secondary or even lower positions on the list of priorities [3]. This is especially concerning considering the pivotal role played by the OASE E-learning system in supporting Udayana University's academic activities. Given its importance, robust security testing becomes paramount to ensure the integrity and protection of this crucial component.

Among the various methods available for testing web applications, there are several approaches such as NIST, ISSAF, and OSSTMM. However, OSSTMM stands out as a more effective, efficient, and comprehensive method for security testing [4]. It is regarded as a globally recognized approach in security testing [5]. Based on this, the author conducted security testing research on the OASE system, combining both OSSTMM and ISSAF methodologies to assess its level of security. This research aims to provide valuable insights to the developers.

The research methodology employed for data collection in this study was a literature review. Relevant papers and resources related to the topic were collected from the internet. For implementation and testing, an experimental approach based on the gathered information was utilized.

Our main objective in this paper is to emphasizes the importance of security testing in Udayana University's OASE E-learning Platform applications, discusses the potential risks faced by the OASE E-learning system, introduces the OSSTMM methodology as an effective approach for security testing at Udayana University, and outlines the research objective of assessing the security level of the OASE system using a combined approach of OSSTMM and ISSAF methodologies.

2. Related Work and Review of Previous Studies

2.1. ISSAF (Information Systems Security Assessment Framework)

ISSAF is a comprehensive security testing framework that focuses on assessing and improving the security of information systems. It provides a structured approach to security testing and covers various aspects of system evaluation and reporting. Unlike other penetration testing frameworks such as OWASP and OSSTMM, ISSAF emphasizes security testing from the implementation perspective [6]. The ISSAF framework penetration testing methodology is designed to evaluate network, system and application control, it provides guidance on conducting penetration testing to identify vulnerabilities and assess the overall security posture of information systems [7].

Research conducted by [8, 9] uses the ISSAF framework to analyze website security, according to him, the ISSAF framework is suitable for doing penetration testing on websites because there are structured guidelines so that testing gets complete and clear directions, highlighting ISSAF as an effective framework for achieving this goal. They emphasize that ISSAF offers a holistic perspective by considering not only technical aspects but also organizational and procedural factors, thereby providing a well-rounded approach to information systems security assessment.

The literature review reveals a comprehensive body of work that underscores the significance of ISSAF as a robust framework for information systems security assessment. The studies discussed in this section provide a solid foundation for our research, informing our methodology and approach to utilizing ISSAF in our study. The insights gained from the literature study will guide our research efforts to contribute to the existing body of knowledge on ISSAF and its practical implementation in securing information systems.

2.2. OSSTMM (Open Source Security Testing Methodology Manual)

The OSSTMM methodology encompasses tests across various channels, including Human, Physical, Wireless, Telecommunications, and Data Networks. This comprehensive coverage makes it well-suited for conducting effective security tests in diverse environments such as cloud computing, virtual infrastructures, messaging middleware, mobile communication infrastructures, high-security locations, human resources, trusted computing, and other logical processes that span multiple channels and require distinct security assessments. OSSTMM framework covers a wide range of security testing areas, including operational, physical, and human security. It emphasizes a holistic approach to security testing by considering not only technical vulnerabilities but also the human factor and physical security controls. By addressing these different aspects, OSSTMM ensures a more comprehensive evaluation of the overall security posture of an organization's systems [10].

OSSTMM also provides detailed guidelines for different types of security tests, including penetration testing, vulnerability scanning, and social engineering assessments. It offers a standardized methodology for executing these tests, ensuring consistency and reliability in the results. Moreover, OSSTMM emphasizes the importance of proper documentation and reporting, enabling organizations to track their security testing activities and communicate the findings effectively [11].

Research conducted by [12, 13] uses the OSSTMM framework. provides detailed guidelines for data network security channel. Yendri Ikhlas Fernando emphasize the importance of a structured and comprehensive methodology for security testing, highlighting OSSTMM as a valuable resource in achieving this goal. They stress that OSSTMM provides a robust foundation by addressing various aspects of security testing, including physical security, operational security, and technical security.

The comprehensive body of work revealed in the literature review underscores the significance of OSSTMM as a robust methodology for evaluating security across various channels. The studies examined in this section provide a solid basis for our research, informing our methodology and guiding our utilization of OSSTMM. The valuable insights gained from the literature study will drive our research efforts towards contributing to the existing knowledge on OSSTMM and its effective implementation in assessing the security of information systems.

2.3. Penetration Testing

Penetration testing, also known as ethical hacking or pen testing, is a proactive security assessment approach that involves simulating real-world attacks on a computer system, network, or web application. The objective of penetration testing is to identify vulnerabilities, weaknesses, and potential entry points that malicious actors could exploit [14].

Penetration testing typically follows a systematic and controlled process. It involves thorough reconnaissance, vulnerability scanning, and targeted exploitation of identified vulnerabilities. The testing may include activities such as network probing, password cracking, social engineering, and attempting to gain unauthorized access to sensitive data [15].

The results of a penetration test provide organizations with valuable insights into their security posture. Detailed reports are generated, outlining the vulnerabilities discovered, their potential impact, and recommendations for remediation. This allows organizations to prioritize and address the identified weaknesses, ultimately improving their overall security resilience [16].

3. Proposed Methods

Research Methodology

The research methodology employed is based on the ISSAF and OSSTMM frameworks. In an effort to enhance system resilience, this study adopts an approach rooted in ISSAF and OSSTMM. The preparation and planning phase involve risk mapping based on the ISSAF and OSSTMM frameworks, while the assessment phase entails comprehensive testing using the OSSTMM methodology.

This research applies a comprehensive approach by integrating elements from both ISSAF and OSSTMM. The assessment phase encompasses defined processes within ISSAF and OSSTMM, involving vulnerability identification, penetration testing, and security analysis.

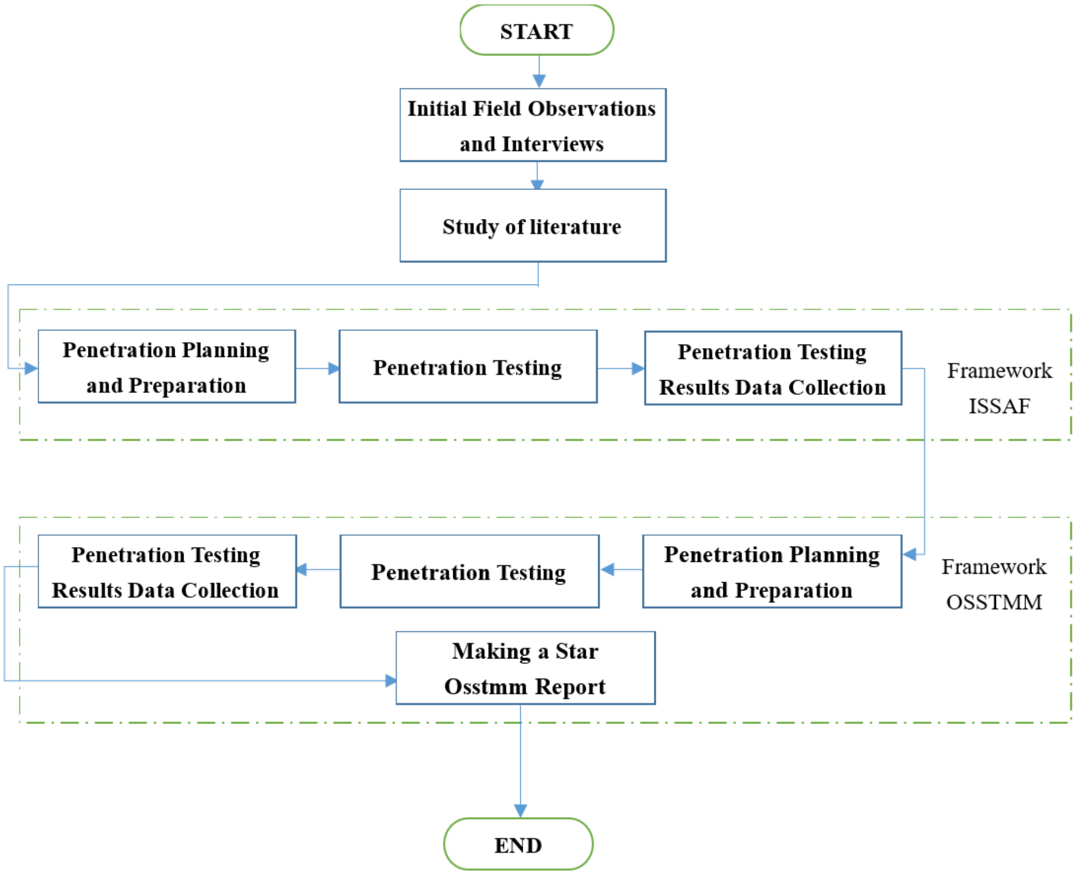

START
Initial Field Observations and Interviews
Study of literature
Penetration Planning and Preparation
Penetration Testing
Penetration Testing Results Data Collection
Framework ISSAF
Penetration Testing Results Data Collection
Penetration Testing
Penetration Planning and Preparation
Framework OSSTMM
Making a Star Osstmm Report
END

Fig.1. Research methodology

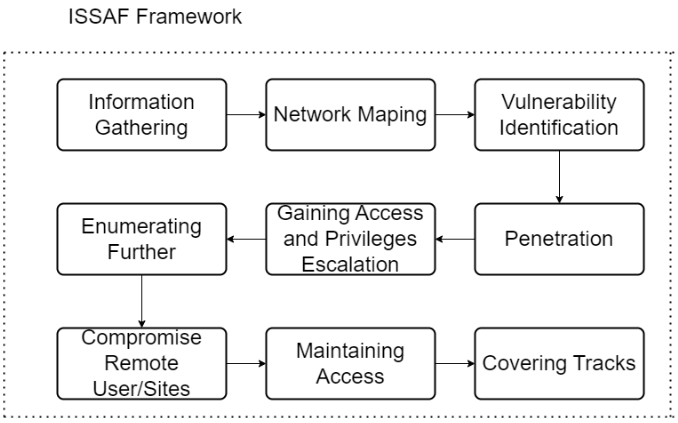


Fig.2. ISSAF framework

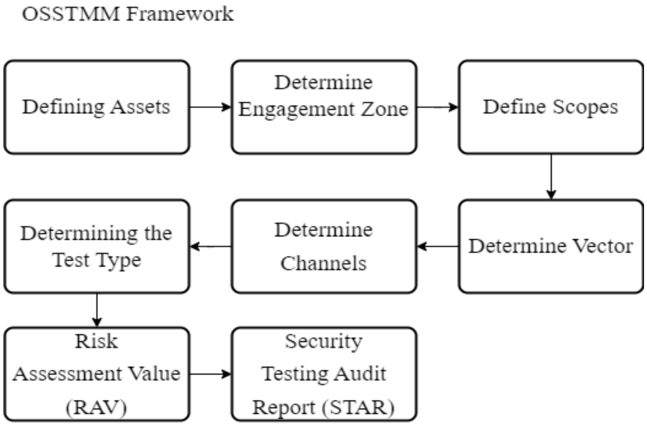

OSSTMM Framework
Defining Assets
Determine Engagement Zone
Define Scopes
Determine Vector
Determine Channels
Determining the Test Type
Risk Assessment Value (RAV)
Security Testing Audit Report (STAR)

Fig.3. OSSTMM framework

4. Test Results

In the process of identifying security vulnerabilities in the OASE web application, an analysis was conducted using two method frameworks, namely the ISSAF framework and the OSSTMM framework. The penetration testing was performed using the ISSAF framework, which consists of a total of 9 sub-stages. All of these sub-modules were tested on the OASE website, resulting in the discovery of several vulnerability findings on the website.

The ISSAF framework commences with Information Gathering, involving the collection of general information pertaining to the target website. The subsequent stage is Network Mapping, wherein specific information regarding the target website's network is gathered. The third stage encompasses Vulnerability Identification, entailing the scanning of vulnerabilities present on the target website. Penetration constitutes the fourth stage, entailing an attack simulation aimed at discovering security vulnerabilities within the website. Gaining Access and Privilege Escalation represents the fifth stage, which involves testing access to the target system. Enumerating Further is the sixth stage, focused on extracting password-related information from the target website. Compromising Remote Users/Sites constitutes the seventh stage, facilitating remote access to the target system. Planting a backdoor within the target system occurs during the eighth stage. Lastly, Covering Tracks is the final stage, encompassing the removal of any traces of the attack log from the target system.

The first step of OSSTMM framework involves defining the protected entity, known as an Asset. It encompasses identifying what needs to be safeguarded. Next, it is essential to understand the surrounding environment of the Asset, which includes protection mechanisms, processes, or services in close proximity. This interaction is referred to as the Engagement Zone. Beyond the Engagement Zone lies the Scope, which encompasses everything necessary to maintain the security of the Asset. The Scope interacts internally and externally. Assets within the Scope are categorized based on their interaction direction, which determines the direction of security testing. This categorization is known as the Vector. Each Vector requires separate testing.

To conduct the tests, the required equipment must be identified. Interactions within each Vector can occur at various levels, categorized by function. These levels are referred to as Channels, such as Human, Physical, Wireless, Telecommunication, and Data Network. Each Channel must undergo separate testing for each Vector. Furthermore, each Channel consists of 17 modules, each with its own unique task depending on the specific Channel, the type of test is determined based on the desired information. This includes conducting tests solely on the interaction with the Asset or expanding the scope to obtain responses regarding security handling. Once the necessary information has been obtained, the next stage is the implementation of security measures. After the implementation stage, the security of the system is evaluated using the Risk Assessment Value (RAV) and Security Testing Audit Report (STAR) assessments. RAV is utilized to generate a security value or score that reflects the level of security provided by the implemented measures.

4.1. ISSAF Framework

A. Information Gathering

Discovery of Target Website's Existence using Google Search Engine: Attackers can uncover sensitive information that may be exploited to launch website attacks. Through this process, several publicly available pieces of information were found, but no sensitive information was discovered.

Retrieval of Domain Registration and IP Block Information using Whois: Domain registration information reveals that the domain was registered in 2011. The IP block and contact details of the domain registrant (name, email, and contact information) were obtained using the Whois tool.

Examination of the target using Google Dork: A vulnerability report was found on the web URL "OASE," indicating a vulnerability to XSS (Cross-Site Scripting).

Identification of Web Server and OS: The web server for "oase" was identified as using the HAProxy load balancer and running on Ubuntu OS. The web server software, Nginx server version 1.14.0, was also identified.

Identification of Web Application Firewall (WAF): The web application firewall is not implemented on the "oase" website.

B. Network Mapping

During the Network Mapping process using the Nmap tools, several findings were discovered. Here are key points Host: Up (0.037s latency) Port Scanning Results:

- 80/tcp: Open, HTTP service
- 443/tcp: Open, HTTPS service
- 8080/tcp: Open, HTTP-proxy service

Banner Grabbing: Port Service and Version Identification:

- 80/tcp: Open, HTTP service with nginx server
- 443/tcp: Open, SSL/HTTP service with nginx server
- 8080/tcp: Open, HTTP-proxy service with HAProxy http proxy version 1.3.1 or later

Tracerouting TCP: Successful traceroute TCP performed, revealing the network path to the OASE server.

C. Vulnerability Identification

During the vulnerability identification process using the Wapiti module, several findings were discovered. Here are key points:

- Content Security Policy Configuration: The configuration of the content security policy was examined for potential vulnerabilities. The goal was to ensure that the policy is appropriately defined and implemented to mitigate risks associated with malicious content execution.
- HttpOnly Flag Cookie: The presence of the HttpOnly flag for cookies was assessed. This flag is crucial in preventing client-side script access to cookies, reducing the risk of cross-site scripting (XSS) attacks.
- Cross-Site Scripting (XSS): The application was analyzed for possible cross-site scripting vulnerabilities. XSS vulnerabilities can allow attackers to inject malicious scripts into web pages, potentially compromising user data and website security. It is important to identify and remediate any instances of XSS to ensure a robust and secure application. The use of the Wapiti module within the ISSAF framework facilitated the systematic examination of these vulnerabilities, providing valuable insights for enhancing the security posture of the system under study.

D. Penetration

During the Penetration process, XSS, Cookie Stealing via Stored XSS, Discovery of Informational Files in the Root Server Directory, Moodle Version. several findings were discovered. Here are the key points:

- XSS: The tester successfully injected malicious scripts into the chat column, manipulating the appearance and behavior of the web page. The test results indicate that the web application is vulnerable to stored XSS attacks, necessitating actions to enhance its security.

- Cookie Stealing via Stored XSS: XSS Cookie Stealing using BurpSuite was successfully performed in the web application's chat column. During the test, the attacker managed to manipulate the chat column and inject an img src element pointing to the tester's website. When the web application user opens that message, the img src element sends the user's cookie to the tester's website, allowing the attacker to steal sensitive information like authentication tokens.
- Discovery of Informational Files in the Root Server Directory: Nikto successfully identified several informational files in the root directory of the server. The test results revealed the existence of confidential files that should be deleted to prevent unauthorized access.
- Moodle Version: During Nikto scanning, a default file named "update.txt" was discovered in the server directory, containing information about the Moodle version 3.10.4.

These findings, discovered during the penetration process, shed light on the vulnerabilities present in the web application. It is imperative for the responsible parties to address these issues promptly and implement robust security measures to mitigate the risk of unauthorized access, data breaches, and potential privilege escalation. By taking decisive actions based on these findings, the web application can fortify its defenses and protect both the system and its users from potential security threats.

E. Gaining Access and Privilege Escalation

During the Gaining Access and Privilege Escalation process, several findings were discovered. Here are the key points:

- PHP Shell Embedding in Image Upload: In the penetration module, the attempt to embed a PHP shell in an image upload file using the Google Chrome browser was unsuccessful. The PHP script on the server did not execute when the image file was uploaded. This indicates that the server has implemented adequate security measures, such as file type validation filters or features that prevent the execution of PHP scripts on user-uploaded files. Employing preventive measures like validation filters and features that hinder script execution on uploaded files greatly reduces the risk of attacks and enhances server security.
- SQL Injection: In the penetration module, the SQL injection test using sqlmap was unsuccessful. This is because the server had an effective anti-CSRF token security mechanism in place, protecting it from SQL injection attacks. The anti-CSRF token mechanism ensures that each request sent to the server is valid and originated from an authenticated user, making SQL injection attacks ineffective. Effective security features are crucial to safeguard servers from various attacks, including SQL injection. Implementing security mechanisms like anti-CSRF tokens or robust parameter validation can help prevent SQL injection attacks and enhance overall server security.

These findings emphasize the importance of addressing the identified vulnerabilities and implementing appropriate security measures to safeguard the web application and its users.

F. Enumerating Further

Table 1. Enumerating further

Vulnerability	Information	Status	Results
Sql injection	Password	failed	Sql injection not found errors in database which can be access to database
XSS	Cookie	Succeed	Finds user cookie information
Local File Inclusion (LFI)	Password	failed	Cannot find vulnerability from local file inclusion so results are not found

G. Compromise Remote User/Sites

The testing phase is carried out by leveraging the obtained results from the previous stages, including acquiring root access. This phase can be conducted once the tester gains entry into the server. Subsequently, utilizing this access privilege, the next step involves exploiting the remote system owned by the OASE system. However, research at this stage cannot be performed due to the tester's inability to access the system.

H. Maintaining Access

This phase can be carried out after successfully obtaining the necessary access to enter the server. The subsequent step involves maintaining access by implementing the simplest form of backdoor, which entails creating a user access with existing privileges to regain entry. Alternatively, a root-kit can be employed as a backdoor mechanism. However, the research did not progress to this stage due to the absence of an entry point into the server via the OASE system.

I. Covering Tracks

Based on the previous testing results, the covering tracks phase did not include testing for successful system infiltration or the installation of a backdoor on the web. Therefore, the covering tracks phase was not conducted in this particular testing.

4.2. OSSTMM Framework

A. Assets, Defines

Examining the security vulnerabilities of the OASE web application at Udayana University. The process begins with an initial analysis of the assets under test to determine the specific points that need to be addressed during the security testing process. These assets encompass the web server, database server, and web service, collectively referred to as the "assets" in this research. They serve as the primary objects of investigation in this study.

B. Engagement Zone

The process of determining the Zone of Engagement is established around Udayana University, and it encompasses two key points: protection mechanisms and processes/services as follows:

- Protection Mechanisms: The gathered data regarding the web application system of OSAE at Udayana University reveals several protection mechanisms, including the utilization of HTTP protocol, TLS versions 1.1 and 1.2, as well as Content Security Policy.
- Processes/Services: One of the services offered is the E-learning website, which plays a significant role in providing educational resources and facilitating online learning.

C. Scope Several Factors Influencing the Scope are as Follows

- Electrical Energy: The assets rely on electrical energy to power the web server and maintain its operation.
- Hosting and Internet Bandwidth: The website operates through hosting services and utilizes internet bandwidth to ensure seamless access and optimal performance.
- Internet Network: The stability and efficiency of the internet network play a crucial role in enabling connectivity and accessibility to the website.

D. Vector

This research focuses on conducting unidirectional testing, specifically security testing through the internet network towards the target system. By utilizing the internet network as the testing medium, the study aims to evaluate the security aspects of the target system and identify potential vulnerabilities or threats that may arise during network-based interactions.

E. Channel

The OSSTMM method incorporates five observation channels, asset in the research are the website of the Udayana University OASE which can be accessed from anywhere via the internet. The engagement zone of the asset contains the HTTPS protocol and services in the form of providing information that can be accessed by the public through the internet network Based on the explanation, all asset scopes can be accessed through the internet network, so that the most appropriate type of channel used in this research is specifically focuses on the Data Network Security channel. The testing revolves around assessing the risk evaluation of security vulnerabilities and conducting penetration testing on the identified weaknesses within the OASE web application at Udayana University. Through this approach, the study aims to identify potential security gaps and strengthen the overall security posture of the web application.

F. Type of Test

The choice of testing methodology should align with the specific conditions that researchers encounter during their study. Therefore, for this particular research, the appropriate testing methodology that aligns with the objectives is double blind this is also known as a black box testing. This approach allows researchers to assess the application's functionality and security aspects from an external perspective, without having access to the internal structure or implementation details. By focusing on the system's inputs and outputs, black box testing enables a comprehensive evaluation of the OASE web application's behavior and security resilience.

Summary of Udayana University OASE Web Findings

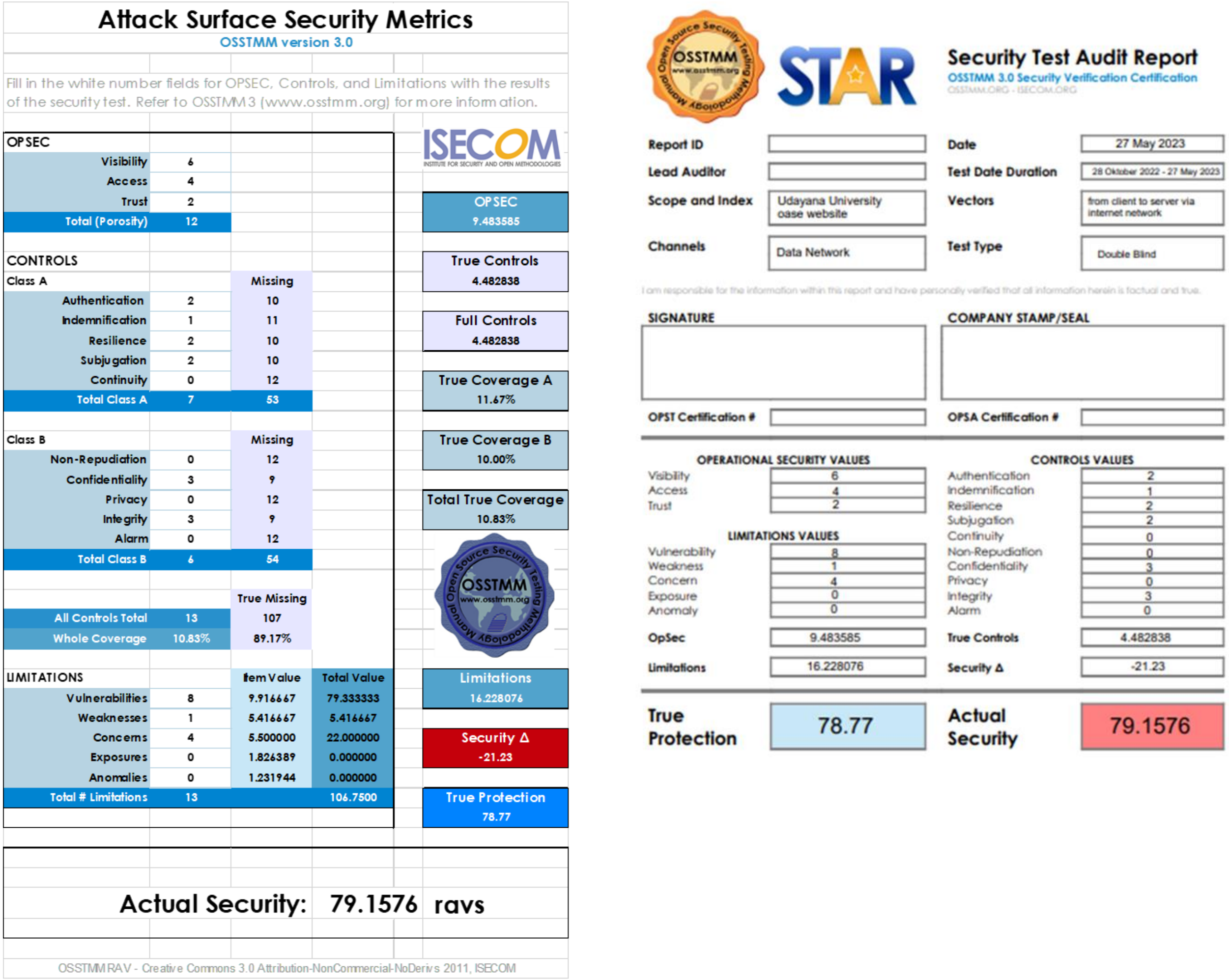
Table 2 presents a summary of the RAV testing results conducted on the Osstmm framework. The analysis using the Osstmm framework yielded valuable information during the testing process, from the table above, the value of each category will be summarized for later calculation of the RAV value.

Table 2. Summary of udayana university oase web findings

Module	No	Information	Amount
Visibility (P)	1	Through the main domain	1
	2	Internet Service Provider (ISP)	1
	3	Web Server	1
	4	Via internal private ip	1
	5	TCP Traceroute	1
	6	Firewall	1
Total			**6**
Access (P)	1	There are 4 ports with open status, 8080,80,443,21	4
Total			**4**
Trust (P)	1	Cross-site Request with 2 domains	2
Total			**2**
Authentication (LC)	1	User login system	2
Total			**2**
Indemnification (LC)	1	Assets use TLSv1.2	1
Total			**1**
Resilience (LC)	1	Denial of login access if the username/password is incorrect	2
Total			**2**
Subjugation (LC)	1	Access URL https://oauth2.unud.ac.id/ to access system user login access URL https://oase.unud.ac.id to become a guest in the system	2
Total			**2**
Confidentiality (LC)	1	Web application using TLS 1.2 and RSA ECDHE algorithm with AES-256 with 256 bit key length	2
	2	SSL Certificate	1
Total			**3**
Integrity (LC)	1	Web application using TLS 1.2 and RSA ECDHE algorithm with AES-256 with 256 bit key length	2
	2	SSL Certificate	1
Total			**3**
Alarm (LC)	1	Excessive login attempts do not get blocks and warnings from the web application	0
Total			**0**
Vulnerability (L)	1	xxs reflect 3 and 1 cookie steal	4
	2	Still using the HTTP protocol	1
	3	OSVDB-3092 - Found a License.txt file that contains sensitive information such as the application site	1
	4	OSVDB-3092 - Found a README.txt, update.txt file that contains the application site's moodle version information	1
	5	CVE-2021-36568 moodle version 3.10.4 In certain Moodle after creating a course it is possible to add an arbitrary "Topic" resource, in this case "Database" with type "Text" where the values are "Field name" and "Field description" " vulnerable to Cross Site Scripting Stored(XSS)	1
Total			**8**
Weakness (L)	1	Excessive login attempts do not get blocks and warnings from the web application	1
Total			**1**
Concerns (L)	1	XSS reflect 3 and 1 cookie steal	3
	2	not redirected to https 1 login form that is not used the default file is not deleted	1
Total			**4**

Table 3. Rav value summary table

Actual security			Amount
Opsec		Visibility	6
		Access	4
		Trust	2
Control	Class A	Authentication	2
		Indemnification	1
		Resilience	2
		Subjugation	2
	Class B	continuity	0
		Non-repudiation	0
		Confidentiality	3
		privacy	0
		Integrity	3
		Alarm	0
Limitations		Vulnerability	8
		Weakness	1
		Concerns	4
		exposures	0
		anomaly	0



Attack Surface Security Metrics

OSSTMM version 3.0

Fill in the white number fields for OPSEC, Controls, and Limitations with the results of the security test. Refer to OSSTMM3 (www.osstmm.org) for more information.

OPSEC		
Visibility	6	
Access	4	
Trust	2	
Total (Porosity)	12	
CONTROLS		
Class A		Missing
Authentication	2	10
Indemnification	1	11
Resilience	2	10
Subjugation	2	10
Continuity	0	12
Total Class A	7	53
Class B		Missing
Non-Repudiation	0	12
Confidentiality	3	9
Privacy	0	12
Integrity	3	9
Alarm	0	12
Total Class B	6	54
		True Missing
All Controls Total	13	107
Whole Coverage	10.83%	89.17%

LIMITATIONS		Item Value	Total Value
Vulnerabilities	8	9.916667	79.333333
Weaknesses	1	5.416667	5.416667
Concerns	4	5.500000	22.000000
Exposures	0	1.826389	0.000000
Anomalies	0	1.231944	0.000000
Total # Limitations	13		106.7500

OPSEC	9.483585
True Controls	4.482838
Full Controls	4.482838
True Coverage A	11.67%
True Coverage B	10.00%
Total True Coverage	10.83%
Limitations	16.228076
Security Δ	-21.23
True Protection	78.77

Actual Security: 79.1576 ravs

OSSTMM RAV - Creative Commons 3.0 Attribution-NonCommercial-NoDerivs 2011, ISECOM

Security Test Audit Report

OSSTMM 3.0 Security Verification Certification

OSSTMM.ORG - ISECOM.ORG

Report ID		Date	27 May 2023
Lead Auditor		Test Date Duration	28 Oktober 2022 - 27 May 2023
Scope and Index	Udayana University oase website	Vectors	from client to server via internet network
Channels	Data Network	Test Type	Double Blind

I am responsible for the information within this report and have personally verified that all information herein is factual and true.

SIGNATURE

COMPANY STAMP/SEAL

OPST Certification #

OPSA Certification #

OPERATIONAL SECURITY VALUES		CONTROLS VALUES	
Visibility	6	Authentication	2
Access	4	Indemnification	1
Trust	2	Resilience	2
		Subjugation	2
LIMITATIONS VALUES		Continuity	0
Vulnerability	8	Non-Repudiation	0
Weakness	1	Confidentiality	3
Concern	4	Privacy	0
Exposure	0	Integrity	3
Anomaly	0	Alarm	0
OpSec	9.483585	True Controls	4.482838
Limitations	16.228076	Security Δ	-21.23
True Protection	78.77	Actual Security	79.1576

Fig.4. RAV & STAR report

The figure above represents the RAV score obtained from the testing results, which is 79.1576. The obtained RAV score is below 100, indicating that the website's controls are insufficient. Therefore, to improve the RAV score, additional controls need to be implemented and vulnerabilities need to be reduced.

5. Discussions

The discussions regarding the findings obtained from the ISSAF Framework and the OSSTMM Framework provide valuable insights into the security vulnerabilities and risks associated with the OASE web application at Udayana University. These discussions focus on the implications of the identified vulnerabilities and the importance of addressing them to enhance the overall security posture of the web application.

The information gathering phase of the ISSAF Framework revealed the existence of the target website and provided essential details such as domain registration information, IP block information, web server, and OS. The absence of a web application firewall (WAF) signifies a potential vulnerability that could expose the web application to various attacks. It is crucial for the responsible parties to consider implementing a WAF to mitigate the risks associated with common web-based attacks.

The network mapping process identified open ports and services associated with the target website, including HTTP, HTTPS, and HTTP-proxy services. These findings highlight the need for proper configuration and security hardening of these services to prevent unauthorized access or exploitation. Additionally, the banner grabbing process provided information about the specific server software versions, such as the Nginx server version and the HAProxy load balancer version. Keeping server software up to date with the latest security patches and updates is essential to protect against known vulnerabilities.

During the vulnerability identification phase, various vulnerabilities were discovered, such as weaknesses in the content security policy configuration, the absence of the HttpOnly flag for cookies, and cross-site scripting (XSS) vulnerabilities. These vulnerabilities could potentially allow attackers to execute malicious scripts, steal sensitive information, or compromise user data. It is crucial to remediate these vulnerabilities promptly by implementing proper security measures, such as configuring the content security policy correctly, enabling the HttpOnly flag for cookies, and implementing input validation and output encoding techniques to mitigate XSS attacks.

In the penetration phase, the web application was found to be vulnerable to stored XSS attacks, which can lead to the manipulation of web page content and the theft of user information. This vulnerability underscores the need for robust input validation and output encoding practices to prevent the injection of malicious scripts. The discovery of informational files in the root server directory and the disclosure of the Moodle version also highlight potential security risks. It is essential to review the server's file and directory permissions, remove any unnecessary informational files, and keep software versions up to date to prevent unauthorized access and exploitation.

Applying the OSSTMM Framework provided further insights into the assets, engagement zone, scope, testing channels, and testing methodology. Understanding the assets under test and the engagement zone helps in identifying potential vulnerabilities and assessing the protection mechanisms and processes/services in place. The scope considerations, such as electrical energy, hosting and internet bandwidth, and the stability of the internet network, highlight the importance of maintaining the underlying infrastructure's security and availability.

The choice of testing methodology, in this case, blackbox testing, allowed for an external evaluation of the web application's behavior and security resilience. By focusing on the system's inputs and outputs, blackbox testing provides a comprehensive assessment of the web application's vulnerabilities and can help identify potential attack vectors. The evaluation of various factors, including visibility, access, trust, authentication, resilience, and confidentiality, contributes to understanding the overall security posture and the potential risks associated with the web application.

In conclusion, the discussions highlight the critical vulnerabilities and risks identified during the assessment of the OASE web application. The findings emphasize the importance of promptly addressing these vulnerabilities to enhance the web application's security posture and protect against potential attacks and unauthorized access. By implementing robust security measures, such as implementing a web application firewall, configuring security policies correctly, enabling HttpOnly flag for cookies, and conducting regular updates and patching, the web application can significantly reduce the risk of exploitation and safeguard user data and privacy.

6. Conclusions

In conclusion, this work significantly advances the field by applying the ISSAF Framework and the OSSTMM Framework to assess the security posture of the OASE web application at Udayana University. The analysis conducted using the ISSAF Framework provided valuable insights into the target website's infrastructure, network mapping, and vulnerability identification. The findings highlighted critical weaknesses such as cross-site scripting vulnerabilities and the absence of a web application firewall, emphasizing the urgent need for implementing robust security measures.

Furthermore, the application of the OSSTMM Framework allowed for a comprehensive evaluation of the web application's assets, protection mechanisms, and risk assessment. The chosen double-blind testing methodology provided an external perspective on the application's functionality and security, revealing areas of concern and potential improvements. The obtained RAV score indicated that the current controls were insufficient, calling for additional measures to reduce vulnerabilities and enhance security.

The scientific justification for this work lies in the importance of proactive security assessments and the identification of vulnerabilities in web applications. By promptly addressing the identified weaknesses, such as improving content security policies and preventing stored XSS attacks, the web application can effectively protect against unauthorized access, data breaches, and potential privilege escalation. This research contributes to the field by providing concrete insights and recommendations for enhancing the security posture of web applications, ultimately safeguarding user information, privacy, and data integrity.

In terms of uses and extensions, the findings and methodologies presented in this work can be utilized by other organizations and researchers to evaluate the security of their own web applications. The ISSAF Framework and the OSSTMM Framework offer systematic approaches that can be applied in various contexts, enabling comprehensive security assessments. Additionally, future studies can build upon this research by exploring other frameworks, conducting deeper analyses, and investigating new techniques for identifying and mitigating web application vulnerabilities.

References

[1] A. W. Wardhana dan H. B. Seta, “Analisis Keamanan Sistem Pembelajaran Online Menggunakan Metode ISSAF pada Website Universitas XYZ,” Jurnal Informatik, vol. 17, no. 3, hal. 226-237, 2021. DOI: 10.52958/iftk.v17i3.3653

[2] Y. I. Fernando dan R. Abdillah, “Security Testing Sistem Penerimaan Mahasiswa Baru Universitas XYZ Menggunakan Open Source Security Testing Methodology Manual (OSSTMM),” Jurnal CoreIT, hal. 33-40, 2016. DOI: 10.24014/coreit.v2i1.2354

[3] A. Rochman, R. R. Salam dan S. A. Maulana, “Analisis Keamanan Website Dengan Information System SecurityAssessment Framework (ISSAF) dan Open Web Application Security Project (OWASP) di Rumah Sakit XYZ,” Jurnal Indonesia Sosial Teknologi, vol. 2, no. 4. DOI: 10.59141/jist.v2i04.124

[4] P. Herzog, “Open Source Security Testing Methodology Manual 3.0,” United States of America: ISECOM, 2010.

[5] M. Prandini dan M. Ramilli, “Towards a practical and effective security testing methodology,” The IEEE symposium on Computers and Communications, hal. 320-325, 2010. DOI: 10.1109/ISCC.2010.5546813

[6] D.P. Anggraeni, B.P. Zen, dan M. Pranata, “SECURITY ANALYSIS ON WEBSITES USING THE INFORMATION SYSTEM ASSESSMENT FRAMEWORK (ISSAF) AND OPEN WEB APPLICATION SECURITY VERSION 4 (OWASPv4) USING THE PENETRATION TESTING METHOD,” Vol 8. No. 3, hal. 497-506, 2022. DOI: 10.33172/jp.v8i3.1777

[7] M. Ayuningtyas dan P. F. Tanaem, “Information Technology Asset Security Risk Management at the Secretariat of the Salatiga City DPRD Using ISO 31000,” Jurnal Sistem Informasi dan Teknologi Informasi (J-SAKTI), vol. 9, no. 1, hal. 92-101, 2022. DOI: 10.1234/j-sakti.v9i1.1439

[8] I. Sanjaya, G. Sasmita, dan D. Arsa, “Evaluasi Keamanan Website Lembaga X Melalui Penetration Testing Menggunakan Framework ISSAF,” Jurnal Ilmiah Merpati (Menara Penelitian Akademika Teknologi Informasi), 2020. DOI: 10.24843/JIM.2020.v08.i02.p05

[9] Herman, et al. “Analisis Keamanan Website Menggunakan Information System Security Asessment Framework (ISSAF),” Jurnal Teknologi Informatika dan Komputer, vol. 9, no. 1, hal. 1-10, 2023. DOI: 10.1234/jtik.v9i1.1439

[10] M.R. Albrecht dan R.B. Jensen, “The Vacuity of the Open Source Security Testing Methodology Manual,” arXiv preprint, 2020. DOI: 10.48550/arXiv.2010.06377

[11] W. Agustiara, A. Pratama dan S. Junaidi, “Analisis Keamanan Protokol Secure Socket Layer Terhadap Serangan Packet Sniffing pada Website Portal Berita Harian Umum Koran Padang,” Jurnal Teknik Informatika Kaputama, vol. 6, no. 1, hal. 10-15, 2022.

[12] A. Ilmi, H. B. Seta, dan I. W. W. Pradnyana, “Evaluasi Risiko Celah Keamanan Menggunakan Metodologi Open-Source Security Testing Methodology Manual (OSSTMM) Pada Aplikasi Web Terbaru Fakultas Ilmu Komputer UPN Veteran Jakarta,” Informatik: Jurnal Ilmu Komputer, vol. 18, no. 2, 2022. DOI: 10.52958/iftk.v18i2.4672

[13] M.A. Nabila, P.E. Mas’udia, dan R. Saptono, “Analysis and Implementation of the ISSAF Framework on OSSTMM on Website Security Vulnerabilities Testing in Polinema,” Journal of Telecommunication Network (Jurnal Jaringan Telekomunikasi), vol. 13, no. 1, 2023. DOI: 10.33795/jartel.v13i1.511

[14] A. Yeboah-Ofori, “Cyber Intelligence and OSINT: Developing Mitigation Techniques Against Cybercrime Threats on Social Media,” International Journal of Cyber-Security and Digital Forensics, vol. 7, no. 1, hal. 87-98, 2017. DOI: 10.17781/P002378

[15] I Putu Agus Eka Pratama, Anak Agung Bagus Arya Wiradarma, "Open Source Intelligence Testing Using the OWASP Version 4 Framework at the Information Gathering Stage (Case Study: X Company)", International Journal of Computer Network and Information Security, Vol.11, No.7, pp.8-12, 2019.

[16] Muhammad Zunnurain Hussain, M. Z. H., & Muhammad Taimoor Aamer Chughtai, "Penetration Testing In System Administration", International Journal Of Scientific & Technology Research, Vol.6, No.06, 2017.

Authors’ Profiles

I Gusti Agung Surya Pramana Wijaya, Dept of Information Technology, Faculty of Engineering, Udayana University, Denpasar, Bali, Indonesia

I Gusti Agung Surya Pramana Wijaya is a student currently pursuing a degree in Information Technology at the Faculty of Engineering, Udayana University Denpasar, Bali, Indonesia His main interests in Computer Network and Security, cloud computing, Virtualization, Computer systems and networks software, as well as Linux.

Gusti Made Arya Sasmita, Dept of Information Technology, Faculty of Engineering, Udayana University, Denpasar, Bali, Indonesia

Gusti Made Arya Sasmita lecturer at Department of Information Technology, Faculty of Engineering, Udayana University Bali, Indonesia. He got his bachelor's degree in electrical engineering, Udayana University, Bali in 1997 and master's degree in Informatics Engineering, Gadjah Mada University in 2003. His research interests are Audit and Network Security.

Google Scholar: https://scholar.google.com/citations?user=Mmo-PjEAAAAJ&hl=id&oi=sra

Scopus: https://www.scopus.com/authid/detail.uri?authorId=56263738300

I Putu Agus Eka Pratama, Dept of Information Technology, Faculty of Engineering, Udayana University, Denpasar, bali, Indonesia

I Putu Agus Eka Pratama took his bachelor's degree at Institut Teknologi Telkom (Telkom University) and master's degree at Institut Teknologi Bandung (ITB), both of them at Informatics. He has been working as a researcher and lecturer at Information Network and System (INS) Research Lab at ITB. From 2015 until now as a lecturer at the Department of Information Technology, Faculty of Engineering, Udayana Universit, Denpasar, Bali, Indonesia. His interest fields are Information Technology, computer network, network security, smart city, and big data. He is also an ICT book author and IT consultant.

Google Scholar: https://scholar.google.co.id/citations?user=KZno-G8AAAAJ&hl=id

Scopus: https://www.scopus.com/authid/detail.uri?authorId=57200177433

How to cite this paper: I Gusti Agung Surya Pramana Wijaya, Gusti Made Arya Sasmita, I Putu Agus Eka Pratama, "Web Application Penetration Testing on Udayana University's OASE E-learning Platform Using Information System Security Assessment Framework (ISSAF) and Open Source Security Testing Methodology Manual (OSSTMM)", International Journal of Information Technology and Computer Science(IJITCS), Vol.16, No.2, pp.45-56, 2024. DOI:10.5815/ijitcs.2024.02.04