Vulnerability Analysis of the KALAM UMI Website Using Penetration Testing

Afriani[a,1], Syahrul Mubarak Abdullah[a,2]

[a] *Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Muslim Indonesia, Jl. Urip Sumoharjo km.05, Makassar dan 90231, Indonesia*
[1] 13020200068@umi.ac.id; [2] syahrul.mubarak@umi.ac.id
**corresponding author*

ARTICLE INFORMATION

Received : 03 – 10 – 2024
Revised : 02 – 10 – 2025
Published : 29 – 10 – 2025

Keywords:
Kalam
SQL Injection
Brute Force

ABSTRACT

KALAM is a web-based online learning platform provided by Universitas Muslim Indonesia (UMI) and hosted on servers managed by the Center for Data and Information Technology (PDTI). In practice, KALAM still exhibits security weaknesses that could be exploited by unauthorized parties for personal gain. One identified weakness lies on the login page, which lacks a CAPTCHA mechanism, thereby enabling brute-force and SQL injection attacks. Potential abuses include altering and exfiltrating data. The researcher employed initial seed data in the form of usernames to identify accounts that could be forcibly accessed. Two outcomes were observed: a response value of 200 indicates that the account can be attacked, whereas a value of 303 signifies that the account is blocked by the firewall.

I. Introduction

According to the National Cybersecurity Operations Center (Pusopskamsinas) and the National Cyber and Crypto Agency (BSSN), 88 million cyberattacks were reported between January and April 2020 [1]. Within that period, there were 25 million attacks in January, 29 million in February, 26 million in March, and 7.5 million from April 1–12. The recorded attack types comprised 56% trojan activity, 43% information gathering, and 1% web application attacks [2]. One example of an organization relying on web infrastructure is Universitas Muslim Indonesia (UMI), which uses the umi.ac.id website for information dissemination and the integrated learning platform KALAM for instruction [3].

KALAM is a web-based online learning facility provided by UMI and hosted on servers managed by the Center for Data and Information Technology (PDTI) [4]. In practice, KALAM still exhibits security weaknesses that can be exploited by unauthorized parties for personal gain. One identified weakness is the absence of a CAPTCHA on the login page, which makes brute-force attacks feasible. Potential abuses include data alteration and data theft. Furthermore, KALAM administrators still employ default username and password configurations, enabling account access by non-owners and exposing data. A frequent incident is the theft of assignments uploaded to KALAM, which can cause harm to account owners.

Prior work [5] employed penetration testing to analyze potential attacks on the SMK Al-Kautsar website. The results indicated susceptibility to DDoS attacks, during which the website became inaccessible [6]. Another study [7] detected brute-force attacks against CCTV IP addresses using computer forensics methods. The analysis revealed a security gap on the MikroTik router login page, with a medium level of vulnerability and an attack success rate reaching 100% [8][9][10].

II. Method

In conducting the penetration-testing study of the KALAM UMI website, the data collection process involved several steps. Multiple sources were consulted, including literature such as journals, books, scholarly works, theses, as well as digital media (the internet). The research stages are shown in Figure 1.

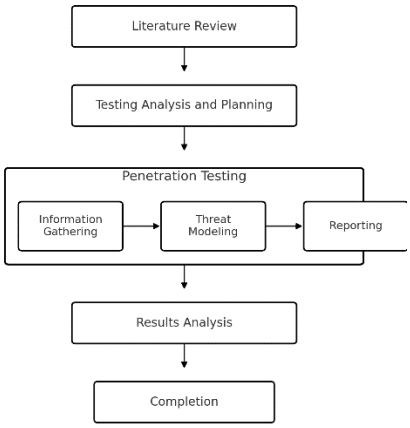

Figure 1. Research Design

A. Literature Review

A literature review was conducted to gather information relevant to the research topic. Sources included books, articles, and other written materials—covering theories, prior research reports, and findings—drawn from both online and offline repositories.

B. Testing Analysis and Planning

At this stage, the authors analyzed and devised a testing plan based on the results of prior system reconnaissance and vulnerability discovery. The analysis and plan informed how the penetration testing would be executed.

C. Penetration Testing

This stage entails simulating attacks to test and evaluate the susceptibility of the system/website to potential threats. The phases applied in this study are as follows:

1) Information Gathering

In this phase, the authors collected data using what is commonly termed passive penetration testing. Data collection was performed manually through documentation from relevant parties or publicly available information related to the system under test. The analysis included an evaluation of the active network topology, user requirements, and a review of the software and hardware used by the system.

2) Threat Modeling

Here, the authors attempted attacks against the KALAM UMI website and prepared the software installations required to conduct them. The attack techniques applied were brute force and SQL injection, with auxiliary tools such as Burp Suite and SQLMap.

3) Reporting

In this phase, the authors drew conclusions from the findings obtained in the preceding stages, including the identification of vulnerabilities observed during testing. Recommendations are also provided regarding measures to mitigate the identified weaknesses and improve overall system security.

D. Results Analysis

The authors analyzed the various tests conducted, including identified vulnerabilities and their resulting impacts.

E. Completion

In the final stage, all analyzed data are compiled into a report, to serve as valid digital evidence that is generally accepted.

As shown in Figure 1, the attack testing within this research design involved a security analysis of the KALAM UMI website (https://kalam.umi.ac.id) using brute-force and SQL injection methods, with Burp Suite as the primary tool. After executing the attacks, the next step was to analyze them in order to evaluate the effectiveness of the brute-force method used, identify security gaps, and determine the extent of the system's vulnerabilities. Following the analysis, the subsequent step was attack closure, which encompassed mitigation

and remediation actions to address the identified weaknesses, thereby enhancing system security and preventing similar future attacks [11][12].

1) Brute-Force Attack Scenario

An attacker compiles a list of student usernames as potential targets. The attacker then uses Burp Suite to automate login attempts across the listed accounts, aiming to assess the strength of the system's security and obtain unauthorized access.

2) SQL Injection Attack Scenario

The attacker identifies a vulnerable, SQL-based website component and injects malicious SQL queries through an input field. The malicious queries are validated and executed by the database. As a result, the attacker may gain access to view and modify records or potentially act with database administrator privileges.

III. Results and Discussion

A. Penetration Testing

1) Information Gathering

At this stage, information was collected from web sources, including general details about the target website—KALAM UMI at https://kalam.umi.ac.id

. The web server in use is nginx, with PHP/7.4.12 detected. There are uncommon HTTP headers present, namely content-style-type: text/css (indicating a CSS content type) and content-script-type: text/javascript (indicating a JavaScript content type). The site uses TLS, but the Strict-Transport-Security (HSTS) header is not defined. HSTS is important to protect against man-in-the-middle (MITM) attacks by ensuring the browser connects only via HTTPS. In addition, the X-Content-Type-Options header is not set, which is important for preventing MIME type sniffing (where a browser guesses a content's MIME type). The Content-Encoding header is set to deflate, which can expose the server to BREACH attacks against HTTP compression. The server also uses a wildcard certificate for the domain *.umi.ac.id, allowing the same certificate to be used across multiple subdomains (see Figure 2).

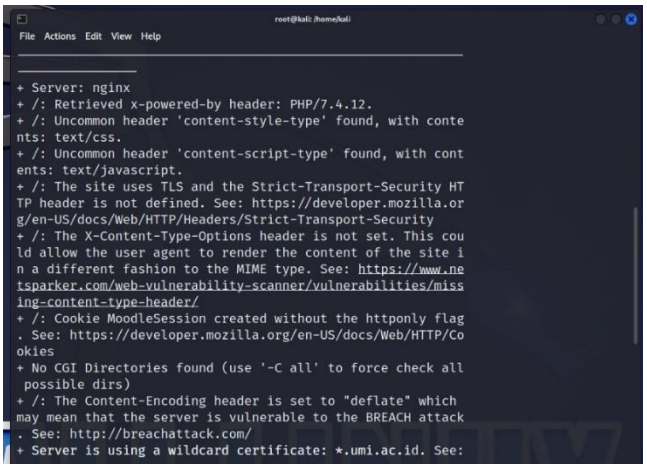

Figure 2. Server Security Scan

2) Threat Modelling

a. Brute Force

At this stage, after successfully matching the proxy and port configuration in the network settings with Burp Suite, an attacker can use Burp Suite to capture login information from the KALAM system. After supplying a list of usernames, Burp Suite performs scanning to determine which usernames receive a successful response. If the response status is 200 and the response length is 55,840, the login attempt is considered successful (see Figure 3).

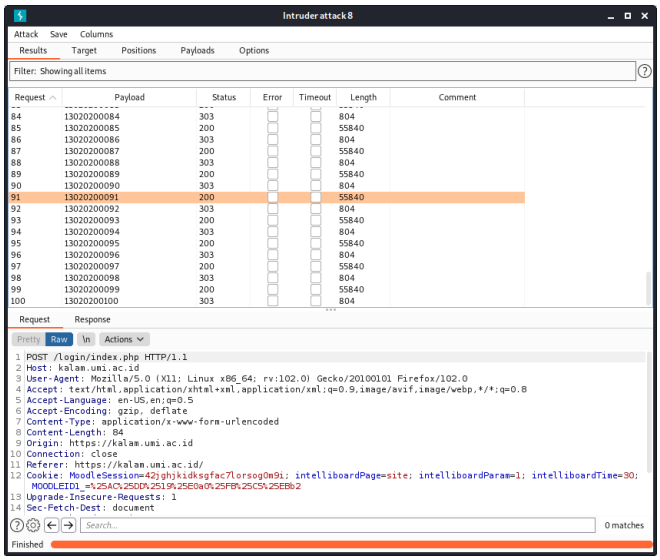

Figure 3. Scan Results

b. SQL Injection

The attacker runs the following sqlmap command:

sqlmap -u https://kalam.umi.ac.id/course/view.php?id=20803 -dbs

This instructs sqlmap to test the provided URL and attempt to enumerate databases if an SQL injection vulnerability exists. When attempting to access the target URL, sqlmap receives a 303 response, indicating a redirect to the login page (https://kalam.umi.ac.id/login/index.php). This shows that direct access to the target URL requires authentication.

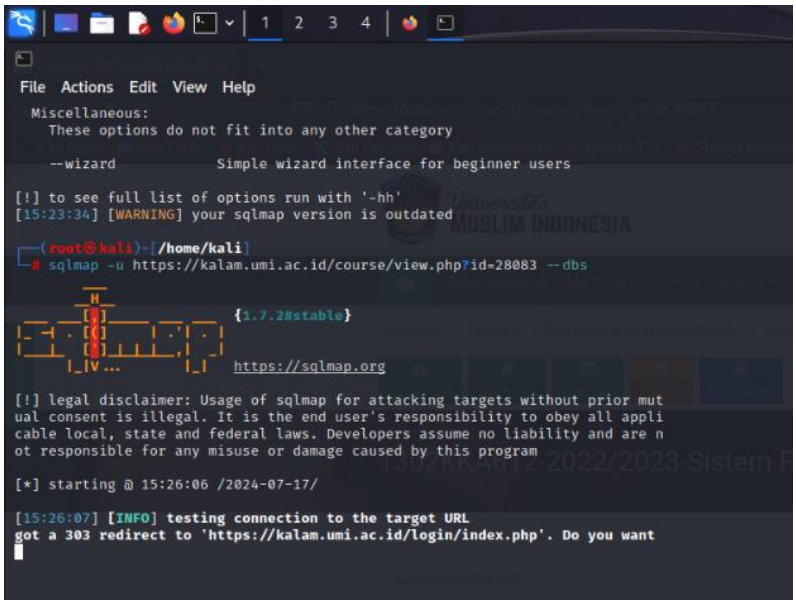

Figure 4. Login Page Redirection Response

The penetration-testing activities on https://kalam.umi.ac.id using brute force and SQL injection identified exploitable conditions with those techniques. Based on Figure 8, several student IDs (stambuk) were susceptible to brute force. Using the brute-force technique, the researcher grouped results by category as shown in Table 1.

Table 1. Lamp Control Delays

No	Username	Description
1	13020200001	OK
2	13020200002	See Other
3	13020200003	OK
4	13020200004	See Other
5	13020200005	OK
6	13020200006	See Other
7	13020200007	OK
8	13020200008	See Other
9	13020200009	OK
10	13020200010	See Other

Status Notes:
"See Other": Further action is required to complete the request.
"OK": The request has been successfully received, understood, and accepted [13].

In this phase, the success of the brute-force technique is attributed to the website lacking effective firewall defenses and to account owners not using passwords that combine letters, numbers, and symbols with a minimum length of eight characters. These conditions enabled successful brute-force attempts against the KALAM UMI website. The attack results were recorded and organized by group [14]. The researcher also attempted SQL injection on the same domain (https://kalam.umi.ac.id) using sqlmap. Based on Figure 11, the SQL injection test was performed only once because a WAF (Web Application Firewall) / IPS (Intrusion Prevention System) was present, indicating that the website has a relatively good level of protection specifically against SQL injection attacks [15].

IV. Conclusion

This study examined vulnerabilities in https://kalam.umi.ac.id using brute-force and SQL injection techniques. The results show that brute force can still be used to identify accounts vulnerable to unauthorized access when initial seed data (usernames) are available. Two response types were observed: a 200 response indicates an account that can be brute-forced, while a 303 response indicates that the action is blocked or intercepted.

References

[1] M. E. Whitman and H. J. Mattord, *Principles of Incident Response and Disaster Recovery*. Thomson Course Technology, 2007.

[2] R. Merlang, "Simulasi Penetration Testing Center Of E-Learning And Education For Students (Cerdas) Universitas Islam Riau Dengan Metode Brute Force Menggunakan Hatch," repository.uir.ac.id, 2022.

[3] S. Charania and V. Vyas, "SQL Injection Attack :Detection and Prevention," Int. Res. J. Eng. Technol., vol. 03, no. 04, pp. 2395–56, 2016.

[4] S. Andriyani, M. F. Sidiq, and B. P. Zen, "Analisis Celah Keamanan Pada Website Dengan Menggunakan Metode Penetration Testing Dan Framework Issaf Pada Website SMK Al-Kautsar," J. Inform. Inf. Technol., vol. 8798, pp. 1–13, 2023.

[5] F. Fachri, "Optimasi Keamanan Web Server Terhadap Serangan Brute-Force Menggunakan Penetration Testing," J. Teknol. Inf. dan Ilmu Komput., vol. 10, no. 1, pp. 51–58, 2023, doi: 10.25126/jtiik.20231015872.

[6] T. Putra, Y. Andrian, and B. Force, "Analisis Kemampuan Url Terenkripsi Base64," vol. 3, no. 04, pp. 31–40, 2020.

[7] R. Suriadi, R. Satra, and F. Fattah, "Peningkatan Keamanan Data dengan Menggunakan Equation pada Metode Playfair Cipher," vol. 1, no. 1, pp. 266–269, 2020.

[8] S. Alam and Y. N. Kunang, "Analisis Serangan Brute Force Pada IP Address Cctv (Closed Circuit Television) Menggunakan Metode Komputer Forensic," Bina Darma Conf. Comput. Sci., vol. 3, no. 3, pp. 544–553, 2021.

[9] B. Darra Deandra Modesta, "Abstract Analysis of Network Security Testing At Faculty of Mathematics and Natural Sciences Lampung University Using Brute Force Method," 2021.

[10] Y. Mulyanto and A. Algi Fari, "Analisis Keamanan Login Router Mikrotik Dari Serangan Bruteforce Menggunakan Metode Penetration Testing (Studi Kasus: Smk Negeri 2 Sumbawa)," J. Inform. Teknol. dan Sains, vol. 4, no. 3, pp. 145–155, 2022, doi: 10.51401/jinteks.v4i3.1897.

[11] K. Nagendran, A. Adithyan, R. Chethana, P. Camillus, and K. B. Bala Sri Varshini, "Web application penetration testing," Int. J. Innov. Technol. Explor. Eng., vol. 8, no. 10, pp. 1029–1035, 2019, doi: 10.35940/ijitee.J9173.0881019.

[12] A. M. Elu, "Rancang Bangun Aplikasi Pendeteksian Vulnerability Structured Query Language (Sql) Injection Untuk Keamanan Website," Respati, vol. 8, no. 22, pp. 111–124, 2017, doi: 10.35842/jtir.v8i22.53.

[13] Jian Chang Chur, A Security Assessment of Egovernment Website in Malaysia, no. November. 2018.

[14] N. L. A. Dewi, A. A. I. I. Paramitha, and E. G. A. Dewi, "Analisis dan Perancangan Sistem Informasi E-Learning Berbasis Learning Management System (LMS) Moodle di SMA Negeri 1 Sukawati," JTKSI (Jurnal Teknol. …, vol. 5, no. 2, pp. 31–42, 2022, doi: 10.56327/jtksi.v5i1.1123.

[15] S. Lika, R. Dwi, P. Halim, and I. Verdian, "Analisa Serangan Sql Injeksi Menggunakan Sqlmap," J. Sist. dan Teknol. Inf., vol. 4, no. 2, pp. 88–94, 2018.