KnE Social Sciences

The 4th International Conference on Science, Mathematics, Environment, and Education

KnE Publishing

Research Article

Analysis of Security on XYZ Institution's Web Server Using Penetration Testing Execution Standard

Effan Najwaini[1]*, Mohammad Wahyu Wardhana[1], and Agus Pebrianto[2]

[1]Digital Business, Politeknik Negeri Banjarmasin, Banjarmasin, Indonesia
[2]Business Administration, Politeknik Negeri Banjarmasin, Banjarmasin, Indonesia

ORCID
Effan Najwaini: https://orcid.org/0000-0002-1693-273X

Corresponding Author: Effan Najwaini; email: effan@poliban.ac.id

Published 27 May 2025

Publishing services provided by Knowledge E

Selection and Peer-review under the responsibility of the ICoSMEE 2023 Conference Committee.

Abstract.
A website serves as an important promotional tool and an information system that facilitates the management of an institution's activities. Therefore, website security must be properly maintained to prevent attacks from hackers who may steal data or damage the website. This research aims to evaluate and identify security vulnerabilities on the institution XYZ's website using the Penetration Testing Execution Standard (PTES) method. PTES is a global standard used to systematically test the security of systems and networks, providing a comprehensive framework for security professionals to conduct penetration testing with the goal of identifying and addressing security gaps. The results of this research are expected to assist institution XYZ in improving its website security and providing a sense of safety and comfort for website users in terms of data security and user experience. Based on the research findings, numerous security vulnerabilities were found on the institution XYZ's web server.

Keywords: website security, security vulnerabilities, PTES

1. INTRODUCTION

Maintaining a secure web presence in today's digital landscape is crucial for institutions managing sensitive operations and data. Institutional websites and internal systems often play a dual role: acting as public-facing platforms for outreach and promotion while simultaneously supporting internal operations such as data management and service delivery. The reliability and security of these systems are not merely technical concerns—they are critical for safeguarding sensitive information and preserving the organization's credibility [1].

Unfortunately, web servers are frequent targets of malicious actors, who seek to exploit weaknesses for unauthorized access or other harmful purposes[2]. A robust and methodical approach to assessing security is necessary to prevent these threats.

One such approach is the Penetration Testing Execution Standard (PTES), a globally recognized framework that outlines systematic steps to identify vulnerabilities, assess risks, and fortify systems against potential breaches. PTES divides the process into clear phases, from planning and information gathering to risk evaluation and reporting [3–8].

This study centers on the security evaluation of XYZ Institution’s web server. Previously reliant on shared hosting and legacy systems with outdated components, the server faced frequent issues such as spam attacks and unauthorized access attempts. These vulnerabilities prompted a transition to a Virtual Private Server (VPS) with updated frameworks to enhance security. While these steps have improved the infrastructure, a thorough evaluation is still needed to identify any residual risks and address them proactively.

By applying the PTES framework, this research aims to uncover potential vulnerabilities in the institution’s web server, assess their risks, and propose actionable recommendations. The findings will not only bolster the institution’s digital security but also provide a roadmap for other organizations aiming to strengthen their systems against evolving cyber threats.

2. METHODOLOGY/MATERIALS

This research employed a structured methodology based on the Penetration Testing Execution Standard (PTES) to evaluate the security of XYZ Institution’s web server. The methodology encompassed four main stages: literature review, data collection, data analysis, and reporting.

The literature review stage was conducted to build a foundational understanding of the PTES framework, web server security, the CodeIgniter framework, and common cyberattack techniques. Relevant studies and publications were analyzed to identify effective methods, tools, and approaches for web security assessment. This stage provided theoretical insights that informed the practical components of the research.

In the data collection stage, two complementary approaches were utilized. First, a technical analysis was conducted using security tools such as Nmap, Burp Suite, and Acunetix. These tools were employed to scan the web server and identify potential vulnerabilities, such as open ports, misconfigurations, and exploitable weaknesses [9]. Second, structured interviews were carried out with the IT administrators responsible for managing the server. These interviews provided qualitative information regarding server configurations, existing security measures, and known challenges. Combining automated technical tools with human input ensured a comprehensive understanding of the server's security posture.

The data analysis stage involved applying the PTES framework systematically to evaluate the collected data. This included [3–8]:

1. Pre-engagement and Scope Definition, where the objectives and boundaries of the penetration test were outlined.

2. Intelligence Gathering, which aggregated technical and contextual information about the server environment.

3. Threat Modeling, identifying potential threats and prioritizing vulnerabilities based on their impact and likelihood.

4. Vulnerability Analysis, where the results from scanning tools were reviewed to pinpoint specific weaknesses, such as cross-site scripting (XSS) and SQL injection vulnerabilities.

5. Exploitation, simulating real-world attacks to understand how these vulnerabilities could be exploited.

6. Post-Exploitation, exploring the extent to which an attacker could maintain access and further compromise the system.

Finally, in the reporting and recommendations stage, the findings were consolidated into a comprehensive report. The report included a detailed list of identified vulnerabilities, categorized by severity, and an assessment of their associated risks. Practical recommendations were provided to mitigate these risks, prioritizing solutions based on their feasibility and impact. This report served as a roadmap for enhancing the institution's web security and offered guidance for similar organizations seeking to fortify their systems.

By following this methodology, the study aimed to provide an accurate assessment of the web server's security posture and actionable insights for improvement. The application of the PTES framework ensured a systematic and comprehensive approach, contributing to both the institution's cybersecurity efforts and the broader field of information security research.

3. RESULTS AND DISCUSSIONS

This research employed the Penetration Testing Execution Standard (PTES) methodology to assess the security of XYZ Institution's web server. Each phase of PTES was conducted systematically to identify vulnerabilities, evaluate risks, and recommend mitigation strategies.

1. Pre-engagement Phase

In the pre-engagement phase, the objectives and scope of the security assessment were defined. The analysis focused on three primary areas: the institution's web profile, internal network penetration, and a web-based application handling administrative data. This phase also involved formal agreements between the research team and the institution, ensuring the process adhered to ethical and professional standards. The main goal was to establish a comprehensive testing plan tailored to the institution's needs.

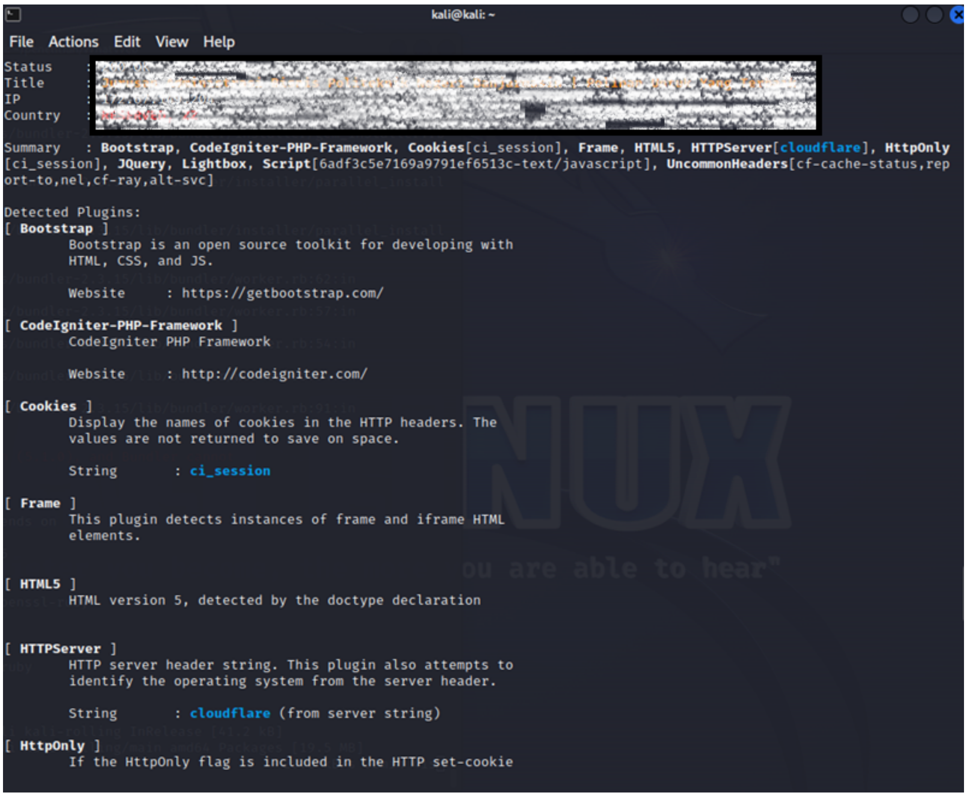
2. Intelligence Gathering

The intelligence-gathering phase involved collecting detailed information about the server and its configurations. Tools such as Whatweb, Censys.io, and Nmap were utilized to gather data on server infrastructure, software versions, and open ports. Key findings indicated that the server used Cloudflare for obfuscating its direct IP address and relied on frameworks such as CodeIgniter and jQuery. Additionally, the server's original IP address and open ports were successfully identified, providing critical insights into potential entry points for attacks.

3. Threat Modeling

The threat modeling phase identified and prioritized potential risks to the web server and its applications. The main risks included unauthorized access to sensitive data stored in administrative applications and the injection of malicious code into the web profile. For instance, vulnerabilities in the web profile could harm the institution's reputation if the site were blocked by security tools or flagged for hosting inappropriate content. Moreover, the administrative application contained sensitive data that, if compromised, could disrupt operational workflows.

4. Vulnerability Analysis

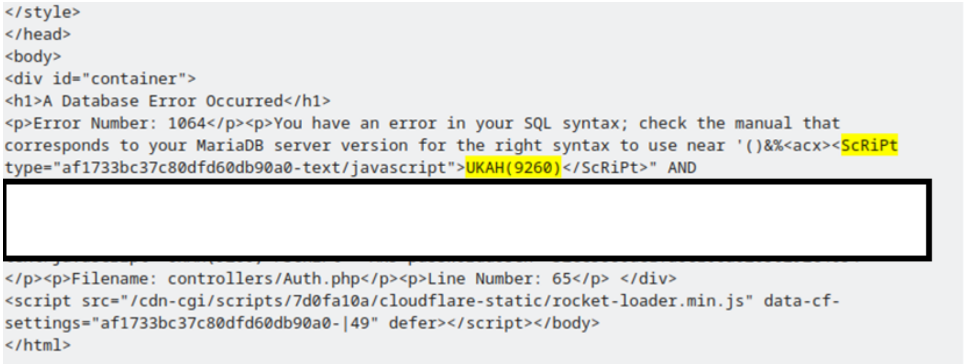


```
Status     : [redacted]
Title      :
IP         :
Country    :
Summary    : Bootstrap, CodeIgniter-PHP-Framework, Cookies[ci_session], Frame, HTML5, HTTPServer[cloudflare], HttpOnly
[ci_session], JQuery, Lightbox, Script[6adf3c5e7169a9791ef6513c-text/javascript], UncommonHeaders[cf-cache-status,rep
ort-to,nel,cf-ray,alt-svc]

Detected Plugins:
[ Bootstrap ]
        Bootstrap is an open source toolkit for developing with
        HTML, CSS, and JS.

        Website     : https://getbootstrap.com/

[ CodeIgniter-PHP-Framework ]
        CodeIgniter PHP Framework

        Website     : http://codeigniter.com/

[ Cookies ]
        Display the names of cookies in the HTTP headers. The
        values are not returned to save on space.

        String       : ci_session

[ Frame ]
        This plugin detects instances of frame and iframe HTML
        elements.


[ HTML5 ]
        HTML version 5, detected by the doctype declaration


[ HTTPServer ]
        HTTP server header string. This plugin also attempts to
        identify the operating system from the server header.

        String       : cloudflare (from server string)

[ HttpOnly ]
        If the HttpOnly flag is included in the HTTP set-cookie
```

Figure 1: The Whatweb Display.

In the vulnerability analysis phase, the server and its applications were examined using Acunetix, revealing significant weaknesses. The results highlighted 26 high-severity vulnerabilities, including Cross-Site Scripting (XSS), SQL Injection, and Server Directory Traversal, along with 61 medium and 28 low-severity issues.

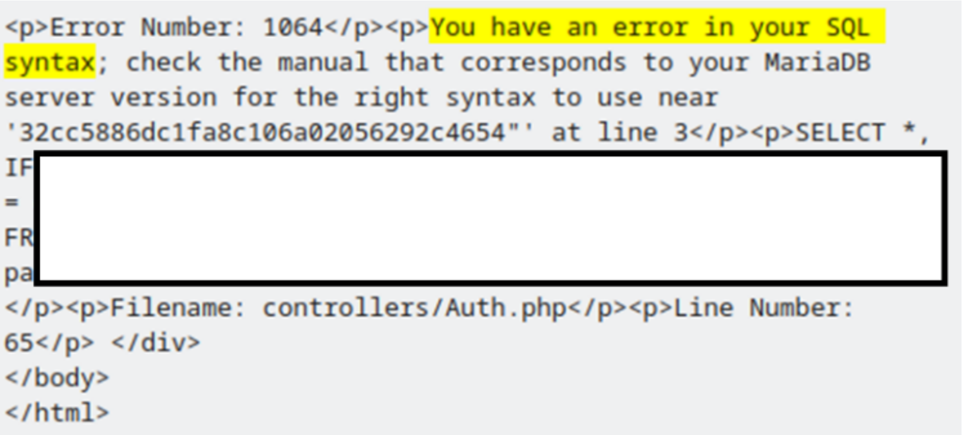
1. XSS vulnerabilities were found in login forms, allowing attackers to inject scripts that could compromise user sessions or steal sensitive information.



```
</style>
</head>
<body>
<div id="container">
<h1>A Database Error Occurred</h1>
<p>Error Number: 1064</p><p>You have an error in your SQL syntax; check the manual that
corresponds to your MariaDB server version for the right syntax to use near '()&%<acx><ScRiPt
type="af1733bc37c80dfd60db90a0-text/javascript">UKAH(9260)</ScRiPt>" AND
[redacted]
</p><p>Filename: controllers/Auth.php</p><p>Line Number: 65</p> </div>
<script src="/cdn-cgi/scripts/7d0fa10a/cloudflare-static/rocket-loader.min.js" data-cf-
settings="af1733bc37c80dfd60db90a0-|49" defer></script></body>
</html>
```

Figure 2: XSS Response.

2. SQL Injection issues enabled attackers to bypass authentication and manipulate database queries, potentially granting unauthorized access to critical data [10].



```
<p>Error Number: 1064</p><p>You have an error in your SQL
syntax; check the manual that corresponds to your MariaDB
server version for the right syntax to use near
'32cc5886dc1fa8c106a02056292c4654"' at line 3</p><p>SELECT *,
[redacted]
</p><p>Filename: controllers/Auth.php</p><p>Line Number:
65</p> </div>
</body>
</html>
```

Figure 3: SQL Injection Issues.

3. Directory Traversal vulnerabilities exposed restricted directories, increasing the risk of unauthorized file access.

These vulnerabilities were linked to insufficient input validation, inadequate configuration settings, and outdated security practices.

5. Exploitation

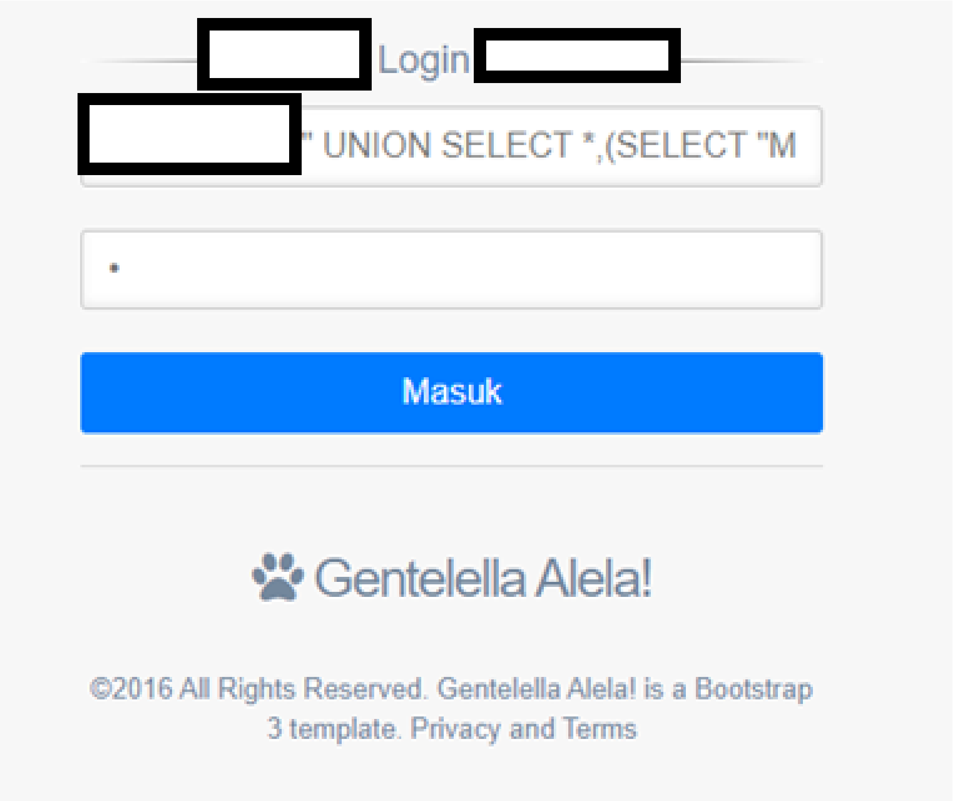
During the exploitation phase, simulated attacks demonstrated how vulnerabilities could be exploited. For example, SQL Injection allowed unauthorized access to both user and administrative accounts. Attackers could bypass authentication by manipulating query inputs and gain control over administrative panels. Additionally, attackers uploaded a backdoor file disguised as a harmless document, enabling persistent access to server resources.

6. Post-Exploitation

In the post-exploitation phase, the consequences of a compromised system were evaluated. Attackers with administrative access could upload malicious scripts, alter critical data, and disrupt operational processes. Backdoor scripts further enabled attackers to maintain access, potentially leading to the complete compromise of the server and its applications.

7. Reporting

The findings were compiled into a detailed report that included recommendations to address vulnerabilities and enhance security. Proposed measures included enabling Cross-Site Request Forgery (CSRF) protection, sanitizing inputs to prevent SQL Injection, and reconfiguring the server environment for production to limit error exposure. Additionally, implementing file access restrictions through .htaccess and developing secure



[redacted] Login [redacted]

[redacted]" UNION SELECT *,(SELECT "M

•

©2016 All Rights Reserved. Gentelella Alela! is a Bootstrap 3 template. Privacy and Terms

Figure 4: SQL Injection.

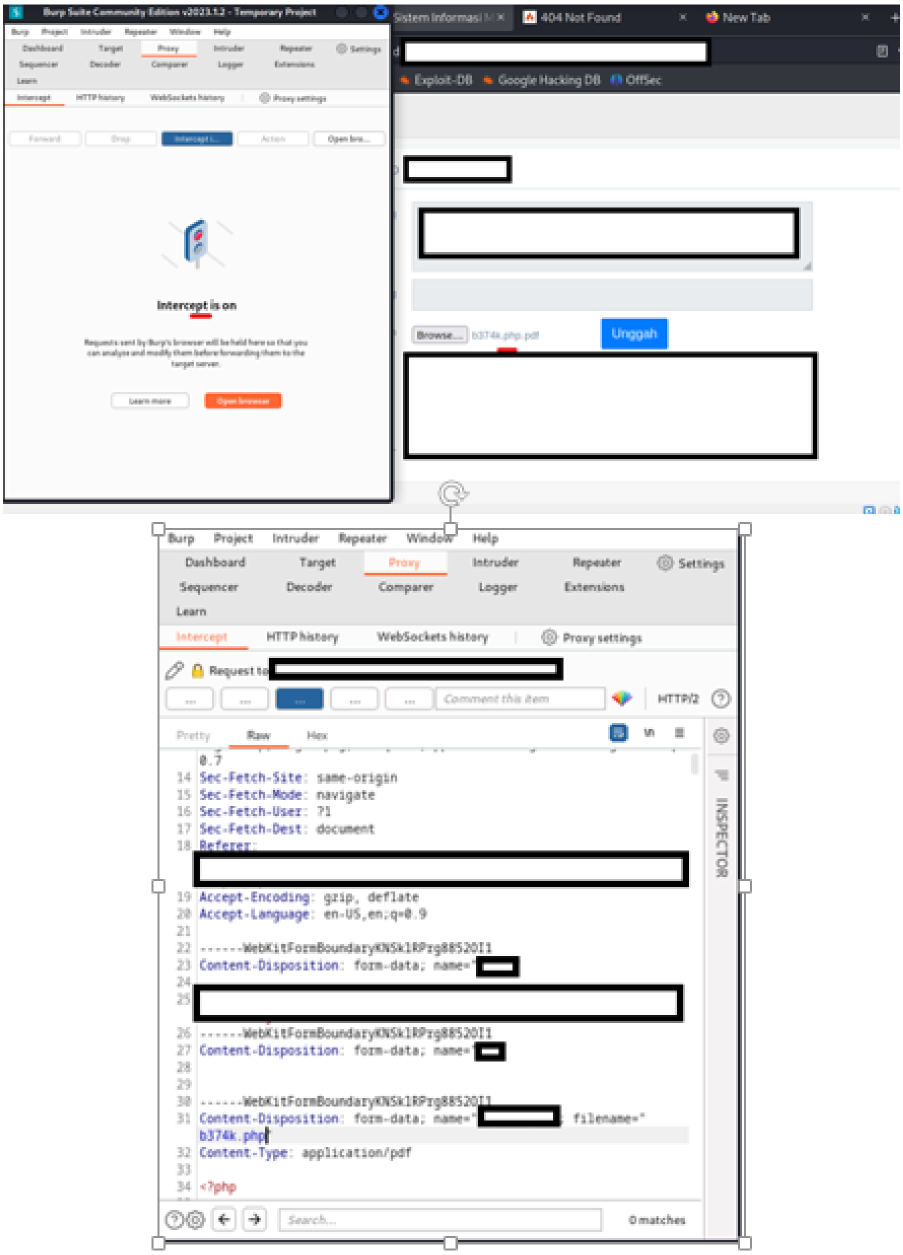
upload mechanisms were advised to prevent unauthorized access and execution of malicious files.

This research identified critical security flaws in the institution's web server and provided actionable recommendations for mitigating risks. The study underscores the importance of regular security audits and proactive measures to safeguard sensitive data and maintain system integrity.

4. CONCLUSION

This study applied the Penetration Testing Execution Standard (PTES) to assess the security of the XYZ Institution's web server. The findings revealed significant vulnerabilities that could pose serious risks to the system's integrity, confidentiality, and availability. Key conclusions derived from this research are as follows:

1. The web server and its applications were found to have multiple high-severity vulnerabilities, including Cross-Site Scripting (XSS), SQL Injection, and Server Directory Traversal. These vulnerabilities allowed attackers to bypass authentication, manipulate sensitive data, and gain unauthorized access to system resources.



Requests sent by Burp's browser will be held here so that you can analyze and modify them before forwarding them to the target server.

[redacted]

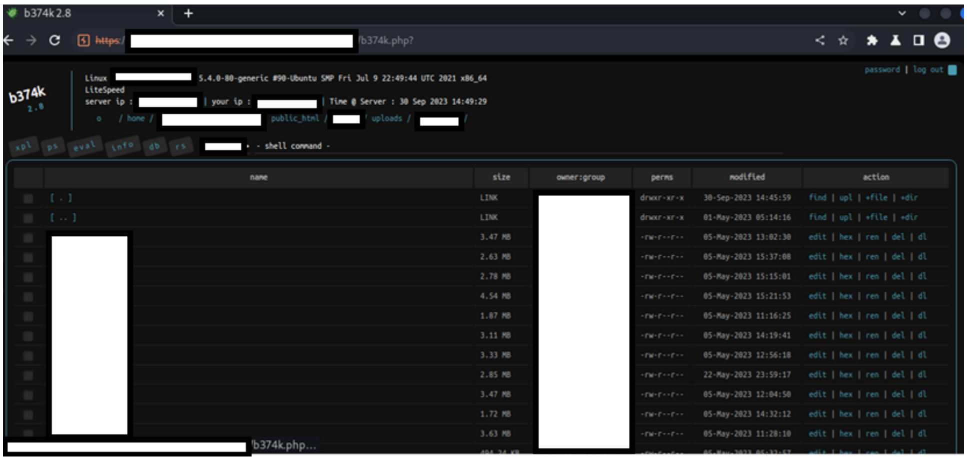
Request to [redacted]

```
0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer:
[redacted]
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

------WebKitFormBoundaryKNSk1RPrg8852OI1
Content-Disposition: form-data; name="[redacted]

[redacted]
------WebKitFormBoundaryKNSk1RPrg8852OI1
Content-Disposition: form-data; name="[redacted]


------WebKitFormBoundaryKNSk1RPrg8852OI1
Content-Disposition: form-data; name="[redacted]; filename="
b374k.php"
Content-Type: application/pdf

<?php
```

Figure 5: Backdoor Upload Process.

2. The current server configuration, operating in a development environment, exposed critical error messages that facilitated exploitation. This improper configuration significantly increased the risk of attacks.

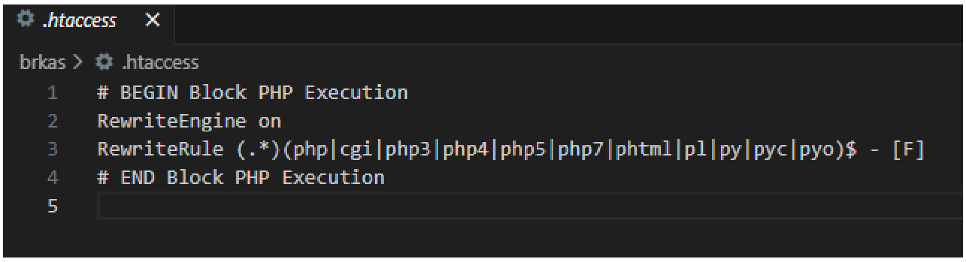
3. Simulated exploits demonstrated how vulnerabilities could be leveraged to gain access to both user and administrator accounts, upload backdoor files, and sustain control over the server. These activities highlighted the severity of existing security gaps.



```
Linux [redacted] 5.4.0-80-generic #90-Ubuntu SMP Fri Jul 9 22:49:44 UTC 2021 x86_64
LiteSpeed
server ip : [redacted] | your ip : [redacted] | Time @ Server : 30 Sep 2023 14:49:29
  o   / home / [redacted] public_html / [redacted] / uploads / [redacted] /
[redacted] - shell command -
```

name	size	owner:group	perms	modified	action
[.]	LINK	[redacted]	drwxr-xr-x	30-Sep-2023 14:45:59	find | upl | +file | +dir
[..]	LINK	[redacted]	drwxr-xr-x	01-May-2023 05:14:16	find | upl | +file | +dir
[redacted]	3.47 MB	[redacted]	-rw-r--r--	05-May-2023 13:02:30	edit | hex | ren | del | dl
[redacted]	2.63 MB	[redacted]	-rw-r--r--	05-May-2023 15:37:08	edit | hex | ren | del | dl
[redacted]	2.78 MB	[redacted]	-rw-r--r--	05-May-2023 15:15:01	edit | hex | ren | del | dl
[redacted]	4.54 MB	[redacted]	-rw-r--r--	05-May-2023 15:21:53	edit | hex | ren | del | dl
[redacted]	1.87 MB	[redacted]	-rw-r--r--	05-May-2023 11:16:25	edit | hex | ren | del | dl
[redacted]	3.11 MB	[redacted]	-rw-r--r--	05-May-2023 14:19:41	edit | hex | ren | del | dl
[redacted]	3.33 MB	[redacted]	-rw-r--r--	05-May-2023 12:56:18	edit | hex | ren | del | dl
[redacted]	2.85 MB	[redacted]	-rw-r--r--	22-May-2023 23:59:17	edit | hex | ren | del | dl
[redacted]	3.47 MB	[redacted]	-rw-r--r--	05-May-2023 12:04:50	edit | hex | ren | del | dl
[redacted]	1.72 MB	[redacted]	-rw-r--r--	05-May-2023 14:32:12	edit | hex | ren | del | dl
[redacted]	3.63 MB	[redacted]	-rw-r--r--	05-May-2023 11:28:10	edit | hex | ren | del | dl

Figure 6: Access to the Backdoor.



```
# BEGIN Block PHP Execution
RewriteEngine on
RewriteRule (.*)(php|cgi|php3|php4|php5|php7|phtml|pl|py|pyc|pyo)$ - [F]
# END Block PHP Execution
```

Figure 7: htaccess file.

4. The use of tools such as Acunetix, Nmap, and Whatweb effectively identified vulnerabilities and provided a comprehensive analysis of the server's security posture. However, the lack of proactive mitigation measures and outdated security practices exacerbated the risks.

To address these challenges, immediate implementation of recommended measures, such as enabling Cross-Site Request Forgery (CSRF) protection, sanitizing input fields, and configuring the server for production use, is essential. These steps will not only enhance security but also safeguard sensitive data and ensure the reliability of the institution's services.

This research highlights the importance of conducting regular security assessments and adopting systematic methodologies like PTES to mitigate risks. The findings provide actionable insights for improving web server security, offering a valuable reference for institutions aiming to fortify their systems against evolving cyber threats.

ACKNOWLEDGMENTS

The authors express their deepest gratitude to XYZ Institution for providing the opportunity and support to conduct this research. Special thanks are extended to the institution's IT team for their collaboration, valuable insights, and assistance throughout the data collection and analysis processes.

References

[1] Kurniawan E, Riadi I. Analisis Tingkat Keamanan Sistem Informasi Akademik Berdasarkan Standard ISO/IEC 27002:2013 Menggunakan SSE-CMM. Intensif J Ilm Penelit Teknol Penerapan Sist Inf. 2018;2(1):12.

[2] Nazwita, Ramadhani S. Analisis Sistem Keamanan Web Server Dan Database Server Menggunakan Suricata. Seminar Nasional Teknologi Informasi, Komunikasi dan Industri (SNTIKI) 9, 2017, p. 308–17.

[3] Ditendra E. EVALUASI KEAMANAN SISTEM INFORMASI AKADEMIK ROKANIA MENGGUNAKAN METODEPENETRATIONTESTING EXECUTION STANDARDS(PTES). Pekanbaru: 2022.

[4] Suradji EL, Chandra DW. Penetration Testing Sistem Jaringan Komputer Untuk Mengetahui Kerentanan Keamanan Server Dengan Menggunakan Metode Penetration Testing Execution Standart (PTES) studi kasus Rumah Sakit Santa Clara Madiun. Salatiga: 2014.

[5] Utoro S, Andi Nugroho B, Rheno Widianto S. Analisis Keamanan Website E-Learning SMKN 1 Cibatu Menggunakan Metode Penetration Testing Execution Standard. Jurnal Mutinetics 2020;6:169–78. https://doi.org/https://doi.org/10.32722/multinetics.v6i2.3432

[6] Fachri F, Fadlil A, Riadi I. Analisis Keamanan Webserver Menggunakan Penetration Test. JURNAL INFORMATIKA. 2021;8:183–90.

[7] W Y. Riadi I, Yudhana A. Analisis Keamanan Webserver Menggunakan Metode Penetrasi Testing (PENTEST). Prosiding Annual Research Seminar 2016, vol. 2, 2016, p. 300–4.

[8] Cunong DN, Saputra M, Puspitasari W. ANALISIS RESIKO KEAMANAN TERHADAP WEBSITE DINAS PENANAMAN MODAN DAN PELAYANAN TERPADU SATU PINTU PEMERINTAHAN XYZYZ MENGGUNAKAN STANDAR PENETRATION TESTING EXECUTIONSTANDARD (PTES). e-Proceeding of Engineering, vol. 7, 2020, p. 2090–5.

[9] Fauzan FY. Syukhri. Jurnal Vocational Teknik Elektronika dan Informatika. Jurnal Vocational Teknik Elektronika Dan Informatika. 2021;9:105–11.

[10] Rumaf N, Anwar K, Safiroh Utsalina D. ANALISIS KEAMANAN WEB SERVER TERHADAP WEBSITE PT. VICTORY INTERNASIONAL FUTURES MALANG DENGAN TEKNIK SQL INJECTION. Jurnal Dinamika Dotcom. 2022;13:73–83.