Data and Metadata. 2025; 4:190
doi: 10.56294/dm2025190

ORIGINAL

Website Protection: An Evaluation of the Web Application Firewall

Protección de Sitios Web: Una evaluación del Web Application Firewall

Gabriela Elizabeth Cárdenas Rosero[1], Cathy Pamela Guevara Vega[1,2], Pablo Landeta-López[1,3]

[1]Universidad Técnica del Norte, 17 de Julio 5-21 Avenue 100150. Ibarra, Ecuador.
[2]eCIER Research Group. 17 de Julio 5-21 Avenue 100150. Ibarra, Ecuador.
[3]Universidad de Sevilla, Pabellón de México. Paseo de las Delicias S/N. 41013. Sevilla, Spain.

Cite as: Cárdenas Rosero GE, Guevara Vega CP, Landeta-López P. Website Protection: An Evaluation of the Web Application Firewall. Data and Metadata. 2025; 4:190. https://doi.org/10.56294/dm2025190

Submitted: 09-05-2024 **Revised:** 18-10-2024 **Accepted:** 20-02-2025 **Published:** 21-02-2025

Editor: Dr. Adrián Alejandro Vitón Castillo

Corresponding author: Gabriela Elizabeth Cárdenas Rosero

ABSTRACT

Introduction: in recent years, a significant increase in attacks targeting web applications has been observed. These attacks compromise application integrity, disrupt services, and have devastating consequences regarding data loss, reputational damage, and financial costs.
Objective: the objective was to evaluate the effectiveness of the Web Application Firewall (WAF) using the OWASP methodology to detect and neutralize attacks on the Universidad Técnica del Norte's web server.
Results: the results were to categorize the main types of attacks detected by the WAF, analyze the most frequent attacks blocked by the firewall, and implement an additional layer of security on the web server.
Conclusions: it was concluded that the WAF detects suspicious or potentially malicious activity in web traffic but fails to identify all cyber threats comprehensively. In addition, the WAF report, broken down each month with the number of frequent attack events identified as malicious, is a crucial tool for the web administrator.

Keywords: WAF; OWASP; Firewall; Security; Web applications; Cybersecurity.

RESUMEN

Introducción: en los últimos años se ha observado un aumento significativo de ataques dirigidos a aplicaciones web. Estos ataques comprometen la integridad de las aplicaciones, interrumpen los servicios y tienen consecuencias devastadoras en términos de pérdida de datos, daño a la reputación y costos financieros.
Objetivo: se evaluó la efectividad del Web Application Firewall (WAF) utilizando la metodología Open Web Application Security Project (OWASP) para detectar y neutralizar ataques a un servidor web académico.
Resultados: se implementó una capa adicional de seguridad en el servidor web, se categorizó los principales tipos de ataques detectados por el WAF y se determinó la eficacia del firewall en la detección y prevención de ataques mediante pruebas de penetración OWASP.
Conclusiones: el WAF detectó actividad sospechosa o potencialmente maliciosa en el tráfico web, pero no logra identificar todas las amenazas cibernéticas de manera integral. Además, el informe del WAF, desglosado cada mes con el número de eventos de ataque frecuentes identificados como maliciosos, es una herramienta crucial para el administrador web.

Palabras clave: WAF; OWASP; Firewall; Seguridad; Aplicaciones web; Ciberseguridad.

INTRODUCTION

Organizations in charge of critical infrastructure have been forced to implement IT security measures due to the increase in cyber-attacks; among the most common are SQL injection, cross-site scripting, brute force attacks, zero-day attacks, among others. Aware of the impact they may suffer on the performance of their web applications, they have increased investment in staff training, monitoring technologies and rapid response to mitigate potential vulnerabilities and threats that affect business continuity. In today's environment, every security strategy can make the difference between success and failure. The web administrator must have a complete understanding of the software and hardware architecture to ensure that the content is efficient and effective. In addition, it is essential that he or she is familiar with the technologies available to accelerate and secure the delivery of content from the web server to the end user's browser.[1] The use of the Internet is not only limited to surfing the web and visiting sites; it now encompasses aspects such as socializing, online shopping, banking transactions, watching streaming movies, participating in real-time events, online education and a wide range of services at the reach of a single click, making it an essential tool for modern society.[2] According to Dalalana,[3] today, the growth and expansion of the internet are remarkable phenomena, over the years, we have witnessed technological progress, where billions of devices connected online are constantly evolving and new cybersecurity threats emerge every second, making our world increasingly susceptible to cyber-attacks. For the present study we considered evaluating the academic information assets of a public university, in this case the Universidad Técnica del Norte (UTN) of Ecuador, specifically in coordination with its Dirección de Desarrollo Tecnológico Informático (DDTI). Motivated by supporting data security in the academic environment and guaranteeing the confidentiality and integrity of the information, the following research questions were proposed:

- RQ_1.- What are the main vulnerabilities and threats that a WAF can help mitigate?
- RQ_2.- What are the most frequent attacks detected according to the OWASP methodology on the Universidad Técnica del Norte's web server?

To answer the research questions posed, an additional Web Application Firewall (WAF) security layer was implemented on the web server. Three specific objectives were set: i) Implement an additional layer of security to protect web applications against potential threats and security vulnerabilities in the UTN web server. ii) Characterize the main types of attacks detected by the WAF that can be exploited and take appropriate protective measures to reduce the risk of a web application being compromised. iii) Determine the effectiveness of the Firewall in detecting and preventing web attacks through OWASP penetration tests involving simulated attacks against web applications protected by the WAF. The Open Web Application Security Project (OWASP) Methodology was applied, which is divided into several groups of security tests that check specific aspects of web applications, considering that the OWASP Test Guide is complete and adequate as a basis for web application penetration testing.(4,5)

Finally, the main types of attacks detected by the WAF were categorized and the effectiveness of the firewall in detecting and preventing attacks was determined using OWASP penetration tests.

The following sections are presented in the document: Related work, where some related work on the application of WAF and the different types of attacks are explained. Method, it explains the phases of the OWASP methodology. Development, it explains the activities performed in each phase of the OWASP methodology. Discussion, the results of the research questions raised are discussed. Conclusions and future work specify the future lines of research.

RELATED WORK

In recent years, there has been a significant increase in targeted attacks on web applications as Kemp says,[6] cybercriminals exploit known and unknown vulnerabilities to infiltrate systems, steal sensitive data, compromise application integrity or disrupt services. These attacks can have devastating consequences in terms of data loss, reputational damage and financial costs. IT security plays a significant role in generating best practices, aimed at ensuring secure and reliable information systems in the face of potential risks, to reduce the likelihood of risks materializing. Srokosz,[4] in his research states, a web application firewall, known as a WAF, is a system designed to protect HTTP applications. The standard WAF employs a static analysis of HTTP requests, which are defined by a set of rules, in order to detect potentially risky elements in these requests.[7,8] Moreover, the proposed concept allows for creating new temporary rules that supply the WAF while increasing the chances of detecting previously undiscovered attacks.[18] Web applications offer powerful internet-based solutions for delivering online services, but they also introduce significant security challenges. Therefore, strengthening the security of web applications to defend against hacking attempts is critical to ensuring their reliability and trustworthiness.[19] Regardless of the specific technical processes used, the overarching principle of these protections is the ongoing monitoring and, potentially, modification of user inputs to align with a predefined set of rules or established statistical patterns. User inputs, such as HTTP GET/POST parameters,

https://doi.org/10.56294/dm2025190

cookies, or headers, are tracked and analyzed as they reach the protected application.[20] Typically, these rules focus on preventing common attacks, such as XSS and SQL injection, which represent targeted threats to the server.[8] The exponential increase in the amount of data circulating on the web and the growing threats to legitimate web applications have become more sophisticated, over time attackers have developed advanced techniques to evade security measures and compromise the security of applications.

UTN is a public university and has 16,000 users who connect simultaneously to technological services, one of them is the web service that hosts about 50 websites distributed for congresses, career blogs, postgraduate, library and main portal, which provide a virtual meeting point between the university and the community in general with topics of interest on research projects, cultural events, sports activities and outreach programs with the local and national community. UTN maintains a massive audience at the higher education level and becomes a cyber attack target due to its visibility. Detecting potential risks represents a significant responsibility for cybersecurity professionals, as it implies having a complete understanding and knowledge of the technological structure of the protected entity.[9] All academic information becomes a frequent target for cybercriminals, who exploit security flaws[2] and breach unprotected information.

However, brute force attacks are repeated, automated attempts to guess administrator login credentials, which can give attackers unauthorized access to legitimate UTN websites, websites are exposed to a wide range of threats, if a web application does not adequately protect sensitive information, such as usernames, passwords or other personal information, attackers can access and use that data for malicious purposes, such as identity theft.[10] Some types of attacks occur when malicious scripts are inserted into legitimate web pages, allowing code execution in the browser of users accessing those pages. This leads to the theft of confidential information such as passwords or session cookies. The exponential increase in the amount of data circulating on the network and the growing threats to web applications have evolved and become more sophisticated. Over time, attackers have developed advanced techniques to evade security measures and compromise the security of applications, making it necessary for academic data to have an effective and efficient protection mechanism.[1]

METHODS

The OWASP methodology was applied, which is one of the most complete guides for analyzing vulnerabilities in web applications.[11] This methodology is based on the identification, assessment and mitigation of security vulnerabilities in web applications, which allows establishing behavioral patterns and quantifying the different types of attacks. This research focuses on the following phases:

Table 1. Research phases

Phase	Description	Procedure
WAF implementation.	Implement an additional layer of security to protect web applications against potential security threats and vulnerabilities in the UTN web server.	WAF configuration in Fortinet, select attack signatures, make IP address changes in the DNS record, this is a strategic process to redirect web traffic before it reaches the server. Verify if the website is protected by the WAF, this allows inspecting and detecting malicious and potential activity, based on a set of rules being the main functionality provided by a WAF.
Vulnerability Analysis and Penetration Testing	Characterize the main types of attacks detected by the WAF that can be exploited and take appropriate protection measures in order to reduce the risk of a web application being compromised. Determine the effectiveness of the Firewall in detecting and preventing web attacks through OWASP penetration tests involving simulated attacks against web applications protected by the WAF.	Prepare the laboratory to perform a vulnerability scan using the OWASP ZAP tool, which identifies security flaws, assigns a risk level and recommends exploitation and solution mechanisms. Security tests were also performed using the OWASP methodology, the process involves active analysis for weaknesses, technical flaws, or vulnerabilities related to the most common OWASP TOP 10 attacks.
WAF Reports.	Provide web administrators with WAF analytics of the most frequent attacks that have been blocked by the Firewall.	Reports can show trends and patterns of attacks, evaluating a website's WAF involves reviewing its configuration, performing security tests, monitoring logs, keeping it up to date and ensuring that it does not adversely affect site performance. WAF reports refer to the distribution or proportion of events according to different categories or characteristics of attacks.

https://doi.org/10.56294/dm2025190

Implementation

The following is an explanation of the activities carried out in each phase of the research.

Phase 1.- WAF Implementation

In this phase the security policy was configured in the WAF, the attack signatures were selected and enabled, block was selected in the action field and high was selected in the severity field. This phase required the participation of the DDTI Operational Level officer, who is in charge of domain administration at Hurricane Electric, and the new IP addresses were registered in the external and internal DNS; this is a strategic process to redirect traffic through the WAF, this allows the WAF to inspect and filter traffic before it reaches the server.

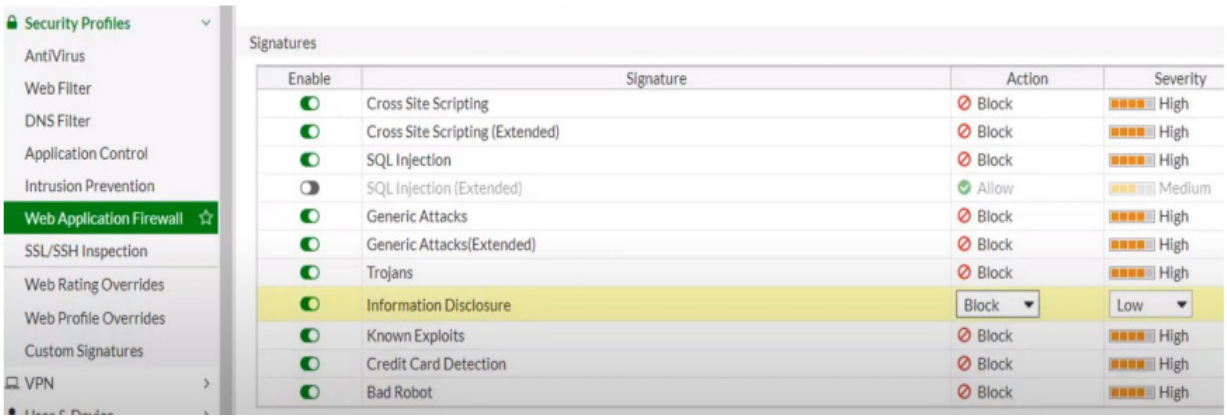

Figure 1. WAF Configuration

To check security, a terminal was used in Kali Linux, and the open source wafw00f tool was used to identify if a website is protected by WAF. Fig 2 shows that the website utn.edu.ec is behind FortiWeb (Fortinet).

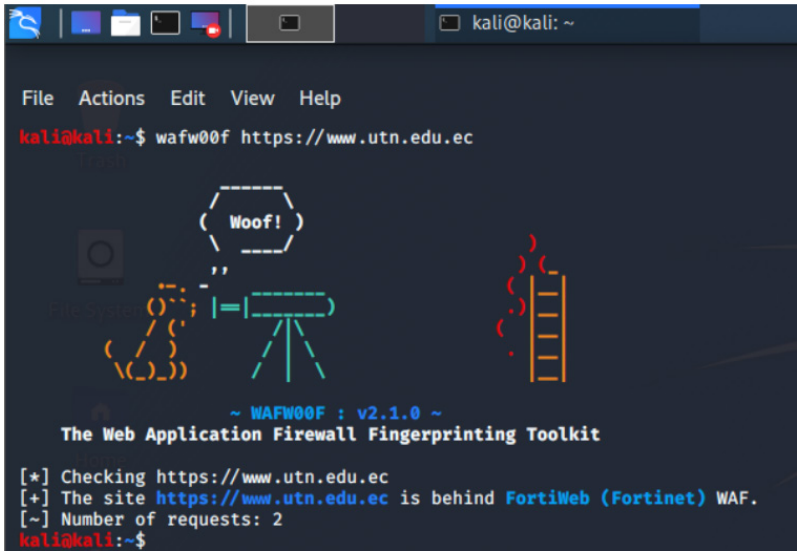

Figure 2. Check if the website is protected

Phase 2.- Vulnerability Analysis and Penetration Testing

To develop this phase, the following test lab was implemented with the following features: Installation of MV Kali Linux: This tool is designed for security professionals, it is widely used to perform security tests and identify vulnerabilities in applications, the virtualization software can be VirtualBox or VMware. OWASP ZAP Installation: OWASP ZAP (Zed Attack Proxy) is an open-source tool widely used to test and assess the security of web applications. It is a vulnerability scanner that identifies security flaws, assigns a risk level and recommends exploitation and remediation mechanisms. wafw00f installation: to identify whether a website is protected by a WAF.

Exploration (Scanning)

For this phase, OWASP ZAP was downloaded and installed from the official website, the application was

https://doi.org/10.56294/dm2025190

run, and the browser was configured in order that it would pass traffic through the ZAP proxy. To define the target, the URL of the web application to be analyzed was required and the scope of the scan was established. [12] OWASP ZAP (Zed Attack Proxy) is a vulnerability scanner that identifies security flaws, assigns a risk level and recommends exploitation and remediation mechanisms.[13] Security testing will never be an exact science, where a complete list of all possible drawbacks to be evaluated can be defined. In fact, security testing is an effective technique for testing the security of web applications under certain circumstances.

Table 2. Vulnerabilities and risk level A01:2021-Broken Access Control

Indicator	Vulnerability	Risk
A01:2021-Broken Access Control	Cross-Domain Misconfiguration	Half
	Cookie without SameSite Attribute	Low
	Timestamp Disclosure - Unix	Low
	Information Disclosure - Suspicious Comments	Informative

Table 2 shows that according to the vulnerabilities found by OWASP ZAP, indicator A01:2021-Broken Access Control can be verified where it can load data from the web browser due to misconfiguration of Cross Origin Resource Sharing (CORS) on the web server, the SameSite attribute is an effective countermeasure against cross-site request forgery, cross-site scripting and synchronization attacks. A timestamp was revealed by the application/web server. The response appears to contain suspicious comments that may help an attacker, Matches made within blocks or script files refer to the entire content, not just the comments.

The solution was to configure the "Access-Control-Allow-Origin" HTTP header for a more restrictive set of domains, or remove all CORS headers altogether, to allow the web browser to enforce the Same-Origin Policy (SOP) in a more restrictive manner. Ensure that the SameSite attribute is set to "lax" or, "strict" for all cookies. Manually verify that the timestamp data is not sensitive, and that the data cannot be aggregated to reveal exploitable patterns.[14] Remove all comments that return information that could help an attacker and fix any underlying problems they refer to.

Table 3. Vulnerabilities and risk level A05:2021-Security Misconfiguration

Indicator	Vulnerability	Risk
A05:2021-Security Misconfiguration	Content security policy (CSP).	Half
	Missing Anti-clickjacking Header	Half
	Cookie Without Secure Flag	Low
	Strict-Transport-Security Header Not Set	Low
	X-Content-Type-Options Header Missing	Low

Table 3 shows the results of the OWASP ZAP scan, it is evident that the Content Security Policy (CSP) represents an additional layer of protection that helps to identify and mitigate various types of attacks, such as Cross Site Scripting (XSS) and data injection attacks. These attacks can be used for a wide range of malicious activities, from information theft to website destruction or malware propagation. However, the response does not address the implementation of the frame-ancestors or X-Frame-Options policy, which guards against ClickJacking attacks. It has been detected that a cookie has been configured without the "secure" indicator, which implies that this cookie can be accessed through unencrypted connections, presenting a security risk. In addition, the absence of the HTTP Strict Transport Security (HSTS) header configuration is highlighted. This mechanism enforces a web security policy that states that user agents, such as web browsers, should interact only over secure HTTPS connections, i.e. via HTTP over TLS/SSL. In addition, it is noted that the Anti-MIME-Sniffing X-Content-Type-Options header has not been set to 'nosniff'. This omission allows older versions of Internet Explorer and Chrome to perform MIME sniffing on the response body, potentially causing the response body to be interpreted and displayed as a different content type than initially declared. The solution was to ensure that the web server is configured to set the Content-Security-Policy header. Modern web browsers support the HTTP Content-Security-Policy and X-Frame-Options headers. Verify that one of them is set on all web pages returned by the website. Whenever a cookie contains sensitive information or is a session token, it should always be transmitted over an encrypted channel. Verify that the secure flag is set for cookies containing sensitive information. Verify that the web server is configured to enforce "Strict-Transport-Security".

Table 4. Vulnerabilities and risk level found by OWASP ZAP"

Indicator	Vulnerability	Risk
WSTG-v42-ATHN-06	Re-examine Cache-control Directives	Informative
	Retrieved from Cache	Informative
	Session Management Response Identifie	Informative
	Modern Web Application	Informative

Table 4 evidence that the cache control header is not configured correctly or is missing, allowing the browser and proxy servers to cache content. For static assets such as css, js or image files, this could be the intent; however, resources should be reviewed to ensure that no sensitive content is cached. This is primarily an issue when caching servers, such as "proxy" caches, are configured on the local network. This configuration is typically found in corporate or educational environments. The presence of the 'Age' header indicates that an HTTP/1.1 compliant caching server is being used. It has been identified that the response provided contains a session management token. The application appears to be a modern web application. If you need to scan it automatically, then Ajax Spider may be more effective than standard.

The solution is to make sure that the HTTP cache control header is configured with "no-cache, no-store, must-revalidate". If an asset is to be cached, you must configure the "public, max-age, immutable" directives. Validate that the response does not contain sensitive, personal or user-specific information. If so, consider using the following HTTP response headers to limit or prevent another user from storing and retrieving cached content, these alerts are informative rather than vulnerability, so there is nothing to fix. Links have been found that do not have traditional href attributes. This is an informational alert and therefore no changes are required.

Table 5. Vulnerabilities and risk level A08:2021 – Software and Data Integrity

Indicator	Vulnerability	Risk
A08:2021 – Software and Data Integrity Failures	Cross-Domain JavaScrip	Low
	Source File Inclusion	Low

Table 5 shows that OWASP ZAP identified that the page includes one or more script files with fonts. The solution was to ensure that JavaScript source files are loaded only from trusted sources and that end users of the application cannot control the sources.

Phase 3.- WAF Reports

After a period of execution, it is possible to analyze the reports generated by WAF. In these reports, "events" refer to individual occurrences or specific incidents of web traffic that the WAF has detected and recorded as possible threats or suspicious activity. The "percentages" in the reports indicate the distribution or proportion of those events according to different categories or characteristics, such as the type of threat, the severity of the attack, the source of the traffic or the number of times a specific attack pattern has been detected. This information is crucial for identifying trends and prioritizing corrective actions or adjustments to WAF rules to improve security and optimize system performance.

RESULTS

To answer RQ_1 and RQ_2, figure 3 shows the data for September 2023, it is evident that they are mostly generic attacks that exceed 1400 daily events, this indicates that the WAF did detect a suspicious or potentially malicious activity in web traffic, but has not identified the specific type of attack; followed by the Ip Reputation attack that exceeds 600 daily events that become malicious attempts in network management.

Figure 4 shows data from October 2023, it is evident that most of them are information disclosure attacks, which are events in which confidential or private data are exposed or disclosed without authorization. These attacks can have serious security consequences. A web application security test focuses solely on evaluating the security of an application. The process involves active analysis for weaknesses, technical flaws, or vulnerabilities. The main types of attacks detected by the WAF when inspecting application-level traffic include malicious activity and potential threats, based on a set of rules being the main functionality provided by a WAF. It logs and detects a large number of attacks against web application vulnerabilities such as injection attacks, forced browsing, unauthorized use or denial of service. It is important to note that the WAF supports detection of all traffic passing through it; in general the purpose of the WAF is to block malicious traffic.[8,14]

Figure 4 shows that most of them are information disclosure attacks, which are events in which confidential or private data are exposed or disclosed without authorization. These attacks can have serious security

consequences.

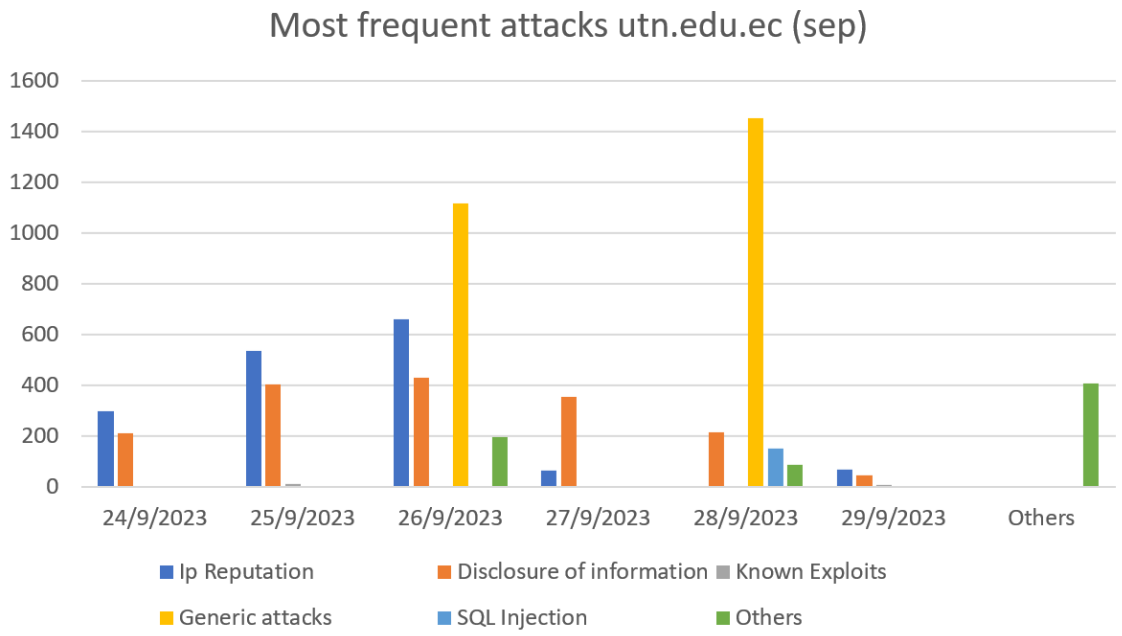

Figure 3. WAF Report- most frequent attacks period: 2023-09-24 to 2023-09-30

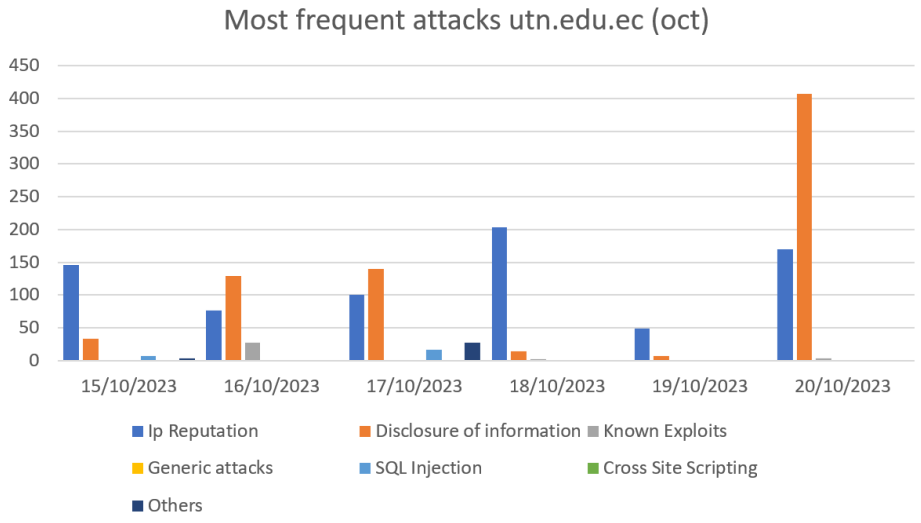

Figure 4. Most frequent attacks period: 2023-10-15 through 2023-10-20

Top Attack Sources

Figure 5 shows the IP addresses with the most attack attempts, which showed a high number of intrusions attempts or malicious activities.

Top Attack Sources		
Source	Events	Percent
190.107.236.163	326	18.40
180.254.70.33	311	17.55
190.107.236.172	177	9.99
190.107.236.176	164	9.26
186.159.16.90	96	5.42
220.250.48.131	38	2.14
Other(227)	660	37.25
Total(233)	**1772**	**100.00**

Figure 5. Ip addresses with most attack attempts

https://doi.org/10.56294/dm2025190

Top Triggered Source Countries

Figure 6 shows the number of events carried out for the month of October 2023, with Ecuador being the country with the most attempted attacks, due to the tests carried out with WASP ZAP, proving the effectiveness of the WAF as a security solution.

Top Triggered Source Countries		
Source Country	Events	Percent
Ecuador	671	37.87
Indonesia	312	17.61
United States	207	11.68
Colombia	96	5.42
Netherlands	81	4.57
Germany	81	4.57
Other(37)	324	18.28
Total(43)	**1772**	**100.00**

Figure 6. Top Attack Sources

Top Attacked http Hosts

The most attacked hosts associated with the domain utn.edu.ec are shown in figure 7.

Top Attacked Http Host		
Http Host	Events	Percent
www.utn.edu.ec	817	46.11
estudiante.utn.edu.ec	385	21.73
colegio.utn.edu.ec	341	19.24
biblioteca.utn.edu.ec	72	4.06
estudiante.utn.edu.ec:443	61	3.44
posgrado.utn.edu.ec	47	2.65
Other(9)	49	2.77
Total(15)	**1772**	**100.00**

Figure 7. Top Attacked http Hosts

COMPARATIVE

Before the WAF configuration, it is not recorded to identify any type of threats, it is to decide that the web administrator does not have a record of attacks or security incidents. After the implementation of the WAF, the number of threats detected can be evaluated and classified according to their type of attack (table 6).

Colors on the heat map:

- Red: High risk, unknown without threat registration (before configuring the WAF).
- Yellow: Moderate risk, with traffic record.
- Green: Remarkable improvement, with attack log (after configuring the WAF).

Table 6. Comparison between before and after implementation of the WAFF proposal

Security metrics	Before setting up the WAF	After setting up the WAF
Visibility of attacks	0 %	100 %
Types of threats detected	0 %	98 %
Number of attacks detected	0 %	100 %
Traffic origin	0 %	100 %
False positives rate	0 %	2 %

Contribution

The cyber threats found by the WAF allow us to identify vulnerabilities that affect digital infrastructure such as student and employee data, financial data, publications, and research, as it helps implement information security measures. Many universities depend on servers for educational platforms, and knowing these threats

https://doi.org/10.56294/dm2025190

helps implement security strategies such as firewalls and other defense mechanisms that promote a constant update of technologies for protection against cyberattacks. In this way, we contribute to the Computer Security Incident Response Team (CSIRT) of universities in charge of analyzing the incident, the impact, and acting appropriately in the event of an emergency.

DISCUSSION

According to the results found in the survey, the following comparative analysis is presented: Operational Level professionals of the DDTI of the UTN know the threats in the cloud and the possible risks that can bring a bad configuration, but it is required to cover needs at the level of human resources in the area of computer security complying with the requirement of the structural organizational chart. This analysis agrees with the study of Reddy (2018), where IT security professionals are very scarce, organizations must address the challenges of cybersecurity and become aware of how leaders must actively participate.[5] On the other hand, Gangineni & Tasmiya,[15] state that they have experience with cloud management and know the most common security threats and the risks that can be caused by misconfiguration, where attackers can take advantage to infiltrate systems and networks, compromising the integrity, confidentiality or availability of data in the organization. In this sense, this study implements an additional layer of security in the UTN web server to perform frequent pentesting tests to mitigate and identify vulnerabilities and weaknesses in the web server before attackers exploit them. In addition, a Content Management System (CMS) was used and the OWASP methodology was applied to identify, correct and protect web applications against possible threats and vulnerabilities in the UTN server as it is a good practice in computer security. These results are consistent with what Gonzáles states,[16] that the Open Web Application Security Project (OWASP) methodology is the most widely used methodology for securing web applications and is the most comprehensive guide for security testing. However, there are some limitations that exist in the study, the results obtained fail to identify all cyber threats in a comprehensive manner, this indicates that a WAF alone is not enough, it must be complemented with another detection solution such as an IDS (Intrusion Inspection System), which is an Intrusion Detection System.

CONCLUSIONS

The main types of attacks detected by the WAF were characterized, where it could be evidenced that the vast majority are generic attacks, this means that the WAF has detected suspicious or potentially malicious activity in web traffic, but has not identified the specific type of attack, it was also recognized Ip reputation attacks, information disclosure, SQL Injection, Cross-Site Scripting and known exploits, these data may vary according to filters per month, which means that the WAF does inspect HTTP requests and responses, but does not recognize unknown threats, that is why it shows the reports as generic attacks.[17]

As part of the WAF evaluation process, security tests were performed with the OWASP ZAP tool, which were executed ethically and responsibly, classifying vulnerabilities according to their severity, resulting in a medium - low - informative level of risk, which helps security evaluators and developers to prioritize corrections. The WAF report is broken down each month with the number of frequent attack events identified as malicious. These reports are a crucial tool for the web administrator because they provide detailed information about the security and performance of a web application. It helps to detect threats, make informed decisions, comply with regulations and continuously improve the security and performance of the web application. Before configuring the WAF, there was no visibility on the number of threats or attacks that could be occurring, since there was no incident record, and the origin of the traffic was not identified. This lack of control and monitoring generated significant technological uncertainty, leaving the site vulnerable and exposed to possible attacks at any time, with no tools to detect or mitigate them effectively.

As future work, an Intrusion Detection System (IDS) can be implemented to detect attacks on the web server and once detected, notify the system administrator by means of a Telegram bot.

REFERENCES

1. Wen SF, Katt B. A quantitative security evaluation and analysis model for web applications based on OWASP application security verification standard. Computers & Security. 2023;135:103532. https://doi.org/10.1016/j.cose.2023.103532.

2. Corao FP, Vanegas MP. Web services administration: Anatomy of the internet. Alpha Editorial; 2021.

3. Dalalana Bertoglio D, Zorzo AF. Overview and open issues on penetration test. Journal of the Brazilian Computer Society. 2017;23:1-16. https://doi.org/10.1186/s13173-017-0051-1.

4. Srokosz M, Rusinek D, Ksiezopolski B. A new WAF-based architecture for protecting web applications against CSRF attacks in malicious environment. En: 2018 Federated Conference on Computer Science and

Information Systems. 2018. p. 391-395.

5. Reddy Y. Big data security in cloud environment. En: 2018 IEEE 4th International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing, IEEE International Conference on Intelligent Data and Security. 2018. p. 100-106. https://doi.org/10.1109/BDS/HPSC/IDS18.2018.00033.

6. Kemp C, Calvert C, Khoshgoftaar TM, Leevy JL. An approach to application-layer DoS detection. Journal of Big Data. 2023;10:22. https://doi.org/10.1186/s40537-023-00699-3.

7. Thein TT, Shiraishi Y, Morii M. Personalized federated learning-based intrusion detection system: Poisoning attack and defense. Future Generation Computer Systems. 2024;153:182-192. https://doi.org/10.1016/j.future.2023.10.005.

8. Kumar H, otros. Securing Web Application using Web Application Firewall (WAF) and Machine Learning. En: 2023 First International Conference on Advanced Electrical, Electronics, Computer and Intelligence. 2023. p. 1-8. https://doi.org/10.1109/ICAEECI58247.2023.10370872.

9. Ponomareva OA, Stepanenko DV, Chernova OV. Modeling Features Threats to the Security of Information in the Process Threat Hunting. En: 2023 IEEE Ural Conference on Biomedical Engineering, Radioelectronics and Information Technology. 2023. p. 305-308. https://doi.org/10.1109/USBEREIT58508.2023.10158844.

10. Kandasamy K, Srinivas S, Achuthan K, Rangan VP. IoT cyber risk: A holistic analysis of cyber risk assessment frameworks, risk vectors, and risk ranking process. EURASIP Journal on Information Security. 2020;2020:1-18. https://doi.org/10.1186/s13635-020-00111-0.

11. Syafiq MS, Norazlina M, Faqihah MF. Enhancement of OWASP Monitoring System with Instant Notification. En: Asia Simulation Conference. 2023. p. 479-487. https://doi.org/10.1007/978-981-99-7243-2_39.

12. Abdulghaffar K, Elmrabit N, Yousefi M. Enhancing Web Application Security through Automated Penetration Testing with Multiple Vulnerability Scanners. Computers. 2023;12:235. https://doi.org/10.3390/computers12110235.

13. Alazmi S, de Leon DC. Customizing OWASP ZAP: A Proven Method for Detecting SQL Injection Vulnerabilities. En: 2023 IEEE 9th International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing, IEEE International Conference on Intelligent Data and Security. 2023. p. 102-106. https://doi.org/10.1109/BigDataSecurity-HPSC-IDS58521.2023.00028.

14. Lathifah A, Amri FB, Rosidah A, otros. Security vulnerability analysis of the sharia crowdfunding website using owasp-zap. En: 2022 10th International Conference on Cyber and IT Service Management. 2022. p. 1-5. https://doi.org/10.1109/CITSM56380.2022.9935837.

15. Jahanavi G, Mubeen T, Aishwarya R, Yogitha R. Cloud Computing using OWASP: Open Web Application Security Project. En: 2023 7th International Conference on Intelligent Computing and Control Systems. 2023. p. 740-743. https://doi.org/10.1109/ICICCS56967.2023.10142457.

16. González Brito HR, Montesino Perurena R. Capacidades de las metodologías de pruebas de penetración para detectar vulnerabilidades frecuentes en aplicaciones web. Revista Cubana de Ciencias Informáticas. 2018;12:52-65.

17. Abikoye OC, Abubakar A, Dokoro AH, Akande ON, Kayode AA. A novel technique to prevent SQL injection and cross-site scripting attacks using Knuth-Morris-Pratt string match algorithm. EURASIP Journal on Information Security. 2020;2020:1-14. https://doi.org/10.1186/s13635-020-00113-y.

18. Sepczuk M. Dynamic web application firewall detection supported by cyber mimic defense approach. Journal of Network and Computer Applications. 2023;213:103596.

19. Alaoui RL, Nfaoui EH. Deep learning for vulnerability and attack detection on web applications: A systematic literature review. Future Internet. 2022;14(4):118.

20. Prokhorenko V, Choo KKR, Ashman H. Web application protection techniques: A taxonomy. Journal of Network and Computer Applications. 2016;60:95-112.

FINANCING

This work has been supported by the Universidad Técnica del Norte, Ecuador.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

AUTHORSHIP CONTRIBUTION

Conceptualization: Gabriela Cárdenas, Cathy Guevara.
Data curation: Gabriela Cárdenas.
Formal analysis: Gabriela Cárdenas.
Acquisition of funds: Cathy Guevara.
Research: Gabriela Cárdenas.
Methodology: Gabriela Cárdenas, Pablo Landeta.
Project management: Cathy Guevara.
Resources: Pablo Landeta.
Software: Gabriela Cárdenas.
Supervision: Cathy Guevara.
Validation: Pablo Landeta.
Display: Pablo Landeta.
Drafting - original draft: Gabriela Cárdenas.
Writing - proofreading and editing: Pablo Landeta.