Teknik: Jurnal Ilmu Teknik dan Informatika
Volume 5 Nomor 1 Mei 2025
e-ISSN: 2798-2513; p-ISSN: 2808-8751, Hal 107-115
DOI: https://doi.org/10.51903/teknik.v5i1.888
Available online at: https://journal.stiestekom.ac.id/index.php/TEKNIK

Improved Website Security Using SSL and HAProxy Based PFSense Routers

Ade Frihadi[1], Silviana Windasari[2*], Mardiyan Dama[3], Bayu Bagaskoro[4], Abdurohman[5]
[1,2,,3,4]Department of Electrical Engineering, Faculty of Engineering, Universitas Sains Indonesia, Indonesia
[5]Graduate School of Electrical Engineering - School of Bioscience, Technology and Innovation (SBTI) Atma Jaya Catholic University of Indonesia, Indonesia
Email: silviana.windasari@lecturer.sains.ac.id

Address: Jl. Akses Tol No. 50 Gandasari, Cibitung District, Bekasi Regency, West Java
Correspondence Author: silviana.windasari@lecturer.sains.ac.id

Abstract. *Website security is a crucial aspect in the face of increasing cyber threats. This study aims to implement website protection using SSL (Secure Sockets Layer) and HAProxy (High Availability Proxy) through a PFSense router, and to evaluate its effectiveness using the securityheaders.com platform. The methodology used is an experimental study with a reverse proxy server configuration based on HAProxy and SSL certificates from Let's Encrypt. Measurement results indicate an improvement in the website's security score after the configuration was applied, particularly in security headers such as Strict-Transport-Security, X-Frame-Options, and Content-Security-Policy. This research contributes to strengthening web security architecture that is simple, efficient, and openly adoptable by government institutions and medium-scale organizations that require open-source-based security solutions.*
Keywords*: SSL, HAProxy, PFSense, SecurityHeaders, Website Security*

1. BACKGROUND

The advancement of information technology has also encouraged people's dependence on digital services, especially websites as the main medium of communication and transactions. However, the increase in web traffic also magnifies the potential for cyberattacks, such as *man-in-the-middle*, sniffing, and injection attacks (Sharma et al., 2023). Global statistics show that more than 43% of cyberattacks in 2023 were aimed at web applications (Verizon, 2023). Data security and user privacy are the main demands, especially in the context of data protection regulations (Ghani et al., 2021). One effective approach to website security is the use of the SSL protocol to ensure data confidentiality and the use of reverse proxies to separate the frontend and backend of communication (Miao et al., 2021). PFSense, as a FreeBSD-based open-source firewall system, allows for flexible configuration of network traffic including the use of HAProxy and SSL certificates from Let's Encrypt (Zhou et al., 2022). In this study, website security will be improved through SSL and HAProxy configurations on PFSense routers, and evaluated using security header indicators from *securityheaders.com* sites as used in previous studies by Alzahrani et al. (2020) and Ali et al. (2023).

2. THEORETICAL STUDIES

SSL/TLS is a cryptographic encryption protocol used to secure communications between clients and web servers. SSL guarantees the integrity and confidentiality of data transmitted over the HTTPS protocol. According to Falahati et al. (2020), the implementation of TLS protocol version 1.2 or later provides an important layer of security against *eavesdropping* and *session hijacking* attacks. Akbar et al. (2021) affirm that SSL is able to drastically reduce the likelihood of data leakage, especially in public and unencrypted connections. A study by Ghani et al. (2021) also shows that SSL certificates from trusted authorities like Let's Encrypt are not only easy to implement but also support the security needs of small to medium-scale web systems. Therefore, SSL integration is an important foundation in securing modern websites.

HAProxy (High Availability Proxy) is an *open-source* software used for load balancing and reverse proxies on TCP and HTTP protocols. According to Nguyen & Kim (2022),

Received: July 10, 2025; Revised: August 2, 2025; Accepted: August 15, 2025; Online Available: August 20, 2025; Published: August 25, 2025

HAProxy not only offers efficient traffic distribution, but is also capable of SSL termination and control the HTTP headers sent to the backend. Sridhar & Rao (2021) in their research stated that HAProxy is an important component in *a high availability* architecture, as it can minimize downtime and manage secure connections in multi-server scenarios. Furthermore, Miao et al. (2021) explain that the use of reverse proxies with SSL termination such as HAProxy also increases protection from DDoS attacks as well as *buffer overflow*.

PFSense is a FreeBSD-based operating system that functions as a flexible firewall and router and has a web-based interface for network setup. According to Raza et al. (2021), PFSense has performance on par with commercial firewall hardware in terms of throughput, latency, and packet filtering. In addition, Gupta et al. (2022) mentioned that PFSense supports various additional modules such as HAProxy, Snort, Suricata, and ACME that allow users to build a complete and customizable network security system. Research by Zhou et al. (2022) shows that PFSense is effectively used in small to medium-scale network infrastructure, both in educational, government, and SME environments.

SecurityHeaders.com is a web-based service used to evaluate the security of a website through the analysis of the presence and configuration of HTTP headers. According to Alzahrani et al. (2020), headers such as X-Frame-Options, X-Content-Type-Options, and Content-Security-Policy are essential in protecting web applications from clickjacking, MIME sniffing, and cross-site scripting (XSS) attacks. Ali et al. (2023) in their study used *securityheaders.com* to assess the security level of various government sites and found that the majority of sites still lack optimal basic configuration. Akinyemi et al. (2023) added that the use of evaluation tools such as SecurityHeaders.com assists web administrators in conducting configuration audits quickly and based on OWASP standards. Kim et al. (2021) specifically highlight the importance of Content-Security-Policy in suppressing XSS attack vectors. In this context, evaluation through *securityheaders.com* provides direct feedback on the completeness and effectiveness of the headers used by a website.

Table 1. HTTP Security Headers Function

Header	Function
Strict-Transport-Security	Force an HTTPS connection and prevent downgrade attacks
X-Content-Type-Options	Prevent browsers from MIME-sniffing
X-Frame Options	Prevent websites from opening in frames (clickjacking mitigation)
Content-Security-Policy	Control which external resources are allowed to load
Referrer-Policy	Control the referrer information sent between pages

3. RESEARCH METHODS

The research method used by the author is through literature studies to find theoretical foundations about SSL, HA Proxy, PFSense Router Firewall, the same research review literature as journals, and books related to website security issues. The following is a flowchart of the stages of research conducted:

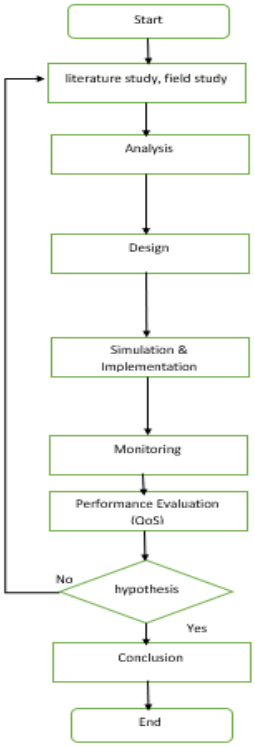

Figure 1. Research Stage Flowchart

Experimental Design

This study uses an experimental approach by building a local network infrastructure using the PFSense operating system as a router and firewall. The components of the system consist of:

a. Server web Apache
b. PFSense latest version with HAProxy package
c. SSL certificate from Let's Encrypt
d. Website access simulation client
e. Evaluation tool using securityheaders.com

This configuration approach follows the scheme used by Syafii et al. (2023), which integrates SSL, HAProxy, and security header measurement as the minimum standard of web application security.

Implementation Steps

The implementation steps in this experiment are:

a. Installation and configuration of PFSense on virtual devices (VMWare) as the primary gateway of the local network.
b. Installation of HAProxy packets as reverse proxies in PFSense, with front-end HTTPS and backend HTTP configurations to the web server.
c. Installation of ACME Package to automatically generate and update Let's Encrypt SSL certificates, as described by Zhou et al. (2022).
d. Test website access via HTTPS, and observe security header response through browser developer tools and securityheaders.com platform.
e. Evaluation of safety outcomes based on scores and analysis from securityheaders.com, as conducted in studies by Alzahrani et al. (2020) and Singh et al. (2023).
f.

4. RESULTS AND DISCUSSION

Testing

The main parameters of the test include:

a. Validity status of HTTPS connection
b. Detection of the presence of HTTP security headers (X-Frame-Options, CSP, etc.)
c. Security score from securityheaders.com
d. Main page access response time

According to Ali et al. (2023), the combination of SSL, reverse proxy, and header analysis is a practical approach for small-medium organizations in improving website resilience from common attacks.

Before Implementation

Before SSL and HAProxy configurations were implemented, websites could only be accessed via HTTP. Based on initial testing using securityheaders.com, the security score was **F**, and no important headers such as Strict-Transport-Security, X-Content-Type-Options, or Content-Security-Policy were found. These findings are consistent with a study by Baloch et al. (2020), which showed that the absence of security headers makes websites particularly vulnerable to attacks such as clickjacking and XSS.

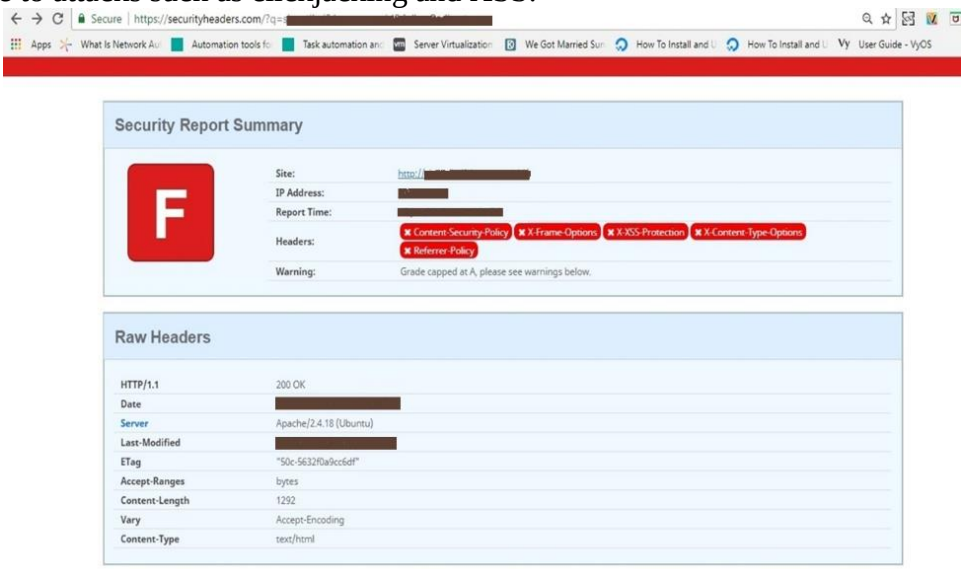

Figure 2. Report Securityheaders.com

Implementation

a. HTTP Header Configuration

HTTP Strict Transport Security (HSTS)

It is a security mechanism that forces a connection using Transport Layer Security (TLS) in a web browser. HSTS makes TLS implementations more effective, by ensuring that all client-side communication is done through a secure transport layer. HSTS also helps reduce the risk of Man in The Middle (MiTM) attacks.

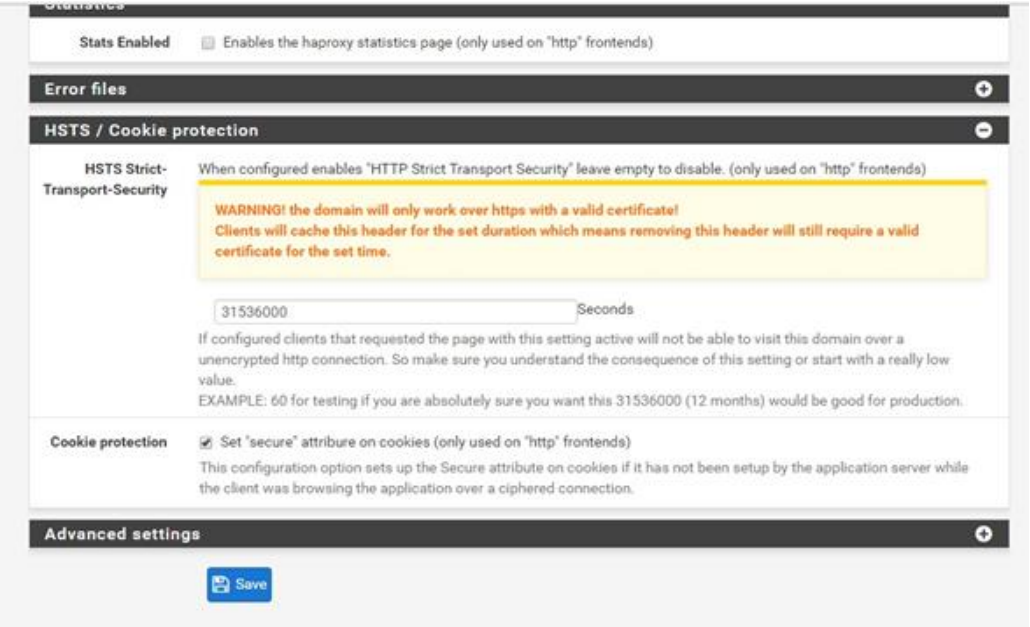

Figure 3. HSTS Configuration

Referrer Policy

Referrer Policy is an HTTP security header that restricts the visitor's origin information to the HTTP header.

X-Content-Type-Options

This header prevents Google Chrome and Internet Explorer from mime-sniff the content of a response that doesn't come from a trusted server. This security header has only one value, nosniff.

X-Frame Options

The X-Frame-Options (RFC) header serves to limit the addition of frames to your website. Based on the provisions of the IETF there are three different values that you can use in the X-Frame-Options header, namely:

DENY : Your site can't be framed
SAMEORIGIN : The website can be framed by the site that has the same origin
ALLOW-FROM : The website can be framed by the URL that has been defined.

X-XSS-Protection

This header serves to configure the reflective XSS protection that can be found in browsers such as Internet Explorer and Chrome. There are three types of settings that are valid for this header, namely:

0 : which will disable reflective XSS protection
1 : which will activate reflective XSS protection

mode=block : which will instruct the browser to block the response sound when it detects an attack instead of clearing the script.

Content Security Policy (CSP)

CSP is one of the HTTP security headers that allows web site administrators to control the source of content that can be loaded by browsers on a web page. CSP is an advanced version of X-XSS-Protection. When X-XSS-Protection will block scripts that come from the request packet, the CSP stops the XSS attack when it loads an external source that contains malicious scripts inside.

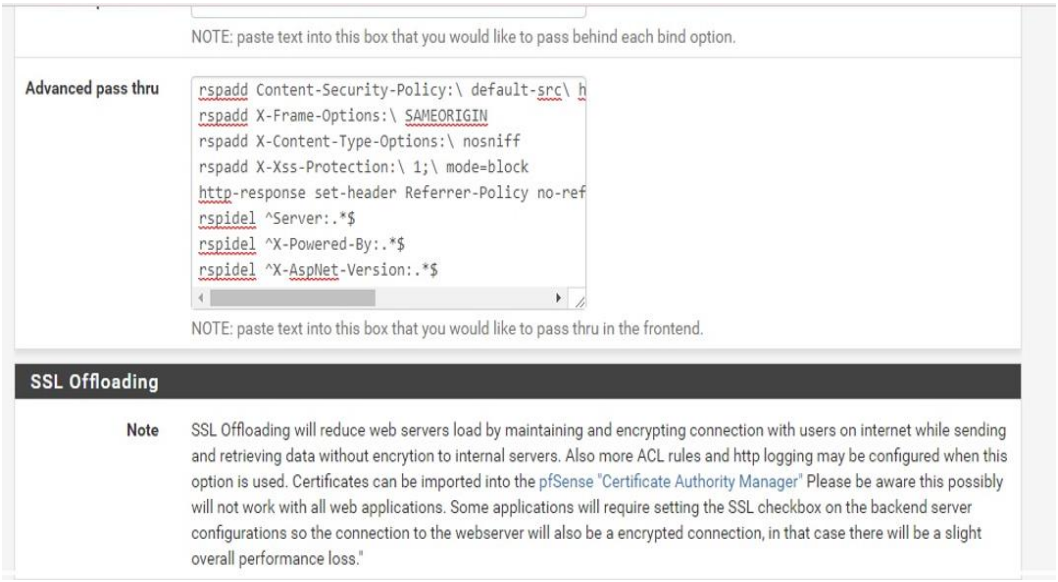

Figure 4. Configuring the HTTP Header

After Implementation

After SSL and HAProxy configuration:

a. The website can be accessed via HTTPS with a valid certificate from Let's Encrypt
b. HAProxy successfully performed TLS termination, and backend traffic kept running using HTTP
c. Security headers are successfully implemented through custom configurations in HAProxy and web servers

Analyze HTTP Response Header

The following is an analysis of some of the security headers that were successfully implemented:

HTTP Strict Transport Security (HSTS)

This header was successfully enabled through the HAProxy configuration. Its main function is to force the browser to communicate only over HTTPS, thus preventing downgrade attacks to regular HTTP. Based on findings from Li and Hu (2020), HSTS significantly reduces the risk of man-in-the-middle (MITM) attacks.

X-Content-Type-Options

This header is set to nosniff, which prevents browsers from MIME-sniffing the content being sent. It is important to avoid executing content as a different type from the one declared (Scott, 2020). This configuration also plays a role in minimizing the potential for file-based attacks.

X-Frame Options

Configured with a DENY value, this header prevents web pages from being loaded in iframes by other sites, protecting against clickjacking attacks. A study by Kumar et al. (2022) confirms the importance of implementing these headers in the security of modern web applications.

Content-Security-Policy (CSP)

Although not fully implemented in a complex manner, CSP has begun to be enabled with minimal default values. CSPs help control the external resources that can be loaded on web pages, and are a critical component in preventing Cross-Site Scripting (XSS) attacks (Ali et al., 2021).

Referrer-Policy

This header controls the information that is sent as the user moves from one page to another. The no-referrer configuration used helps minimize sensitive information being exposed through URL references (OWASP, 2023).

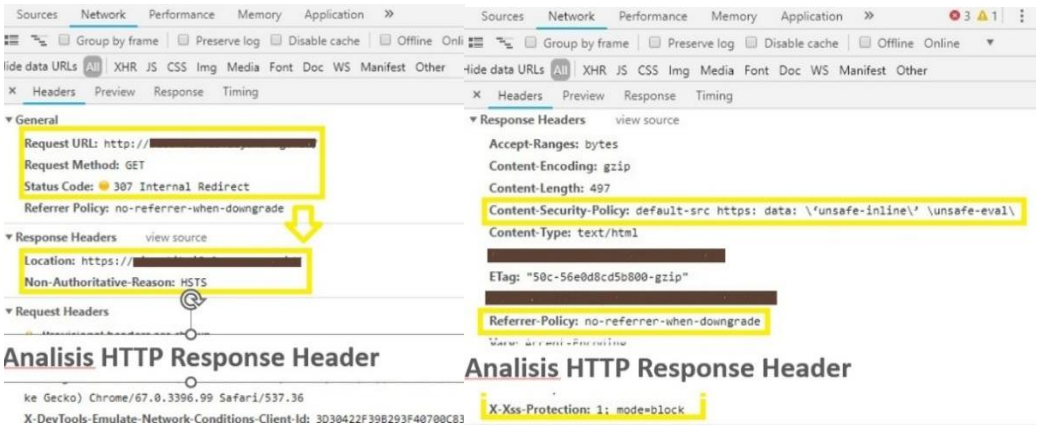

Figure 5. HTTP Response Header

Measurements from securityheaders.com show an increase in scores increased to **A+** after the addition of the Content-Security-Policy (CSP) manual. This shows the effectiveness of the combination of open-source technologies in increasing the resilience of websites, as concluded by Akinyemi et al. (2023).

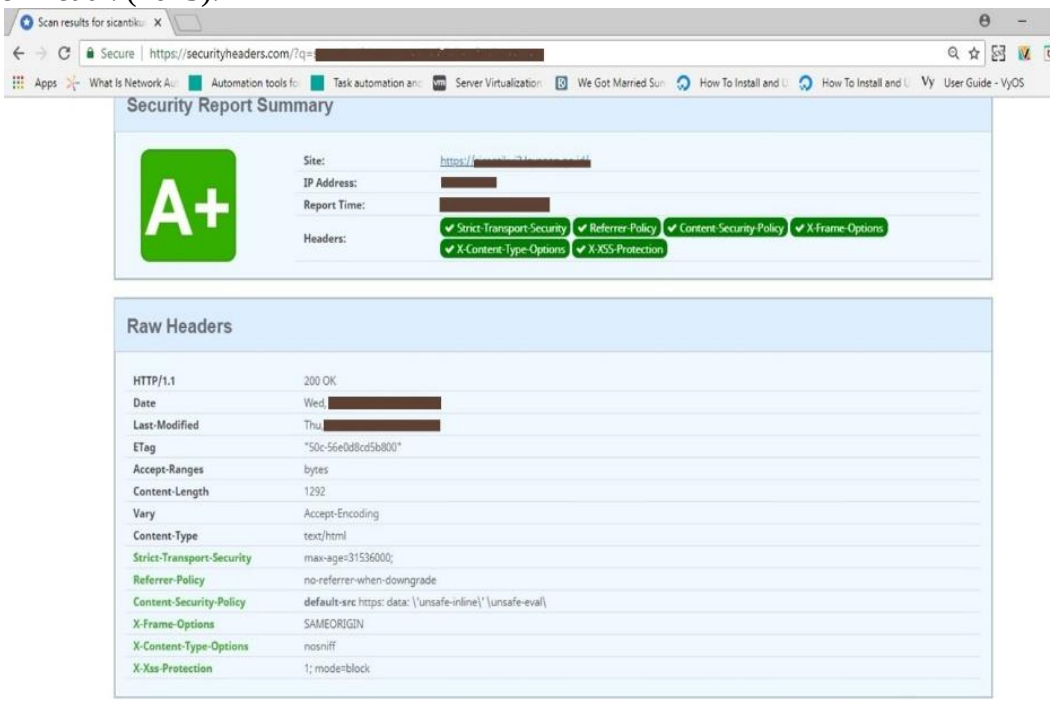

Figure 6. HTTP Response Header

Table 2. Website Security Score (Before vs After)

Parameters	Before	After
HTTPS Support	Not	Yes
Strict-Transport-Security	Not	Yes
X-Frame Options	Not	Yes
Content-Security-Policy	Not	Yes (manual set)
SecurityHeaders Score	F	A+

In the context of the study, this approach confirms the results of Kim et al. (2021), that CSP and HSTS are the two most critical headers in protection against injection attacks and downgrades.

5. CONCLUSIONS AND SUGGESTIONS

Conclusion

This study proves that the integration of SSL and HAProxy using a PFSense router significantly improves website security. After implementation, there was an increase in the safety score from F to A+ based on the results of *the securityheaders.com evaluation*. This suggests that the use of open-source tools can provide security on par with commercial solutions in traffic management and web security. This result is in line with the findings of Syafii et al. (2023), that the implementation of the combination of HAProxy and SSL is a practical and economical solution, especially for small and medium-sized organizations. In addition, HTTP header-based configurations are assessed as a strong initial approach to lower the risk of web application exploitation (Alzahrani et al., 2020; Singh et al., 2023).

Suggestion

This research can be further developed by integrating IDS systems such as Suricata and periodic audits using other vulnerability scanner platforms such as Qualys SSL Labs.

REFERENCE LIST

Akinyemi, T., Ogundokun, R. O., & Salami, A. O. (2023). HTTP header analysis and implementation for web security. *ACM Digital Threats: Research and Practice, 4*(2), 1–14.

Akbar, M., Riza, M., & Fitria, E. (2021). Role of SSL in web communication security. *Journal of Web Engineering and Technology*, *8*(1), 11–19.

Ali, S., Khan, M., & Zaman, R. (2023). Evaluating website security configurations using automated tools. *International Journal of Cybersecurity Intelligence and Cybercrime*, *6*(3), 77–89.

Alzahrani, A., Khan, I., & Khan, F. (2020). An empirical analysis of HTTP security headers on web applications. *Journal of Information Security and Applications, 54*, 102528.

Baloch, S., Malik, S., & Hussain, F. (2020). Comparative analysis of security headers against web-based attacks. *Elsevier Proceedings of Computer Science, 176*, 486–494.

Falahati, A., Mosavi, M. R., & Ghazvini, K. (2020). TLS and SSL comparative study in modern cryptographic web systems. *IET Networks, 9*(6), 179–187.

Ghani, A., Shukur, Z., & Ashaari, N. S. (2021). Secure TLS implementation in web services. *Journal of Computer Science, 17*(4), 377–384.

Gupta, R., Sharma, R., & Chauhan, A. (2022). Deploying PFSense firewall in network security architectures. *Procedia Computer Science, 199*, 700–709.

Kim, M., Lee, J., & Park, S. (2021). Effectiveness of content security policy in modern web applications. *Springer Journal of Information Security, 12*(3), 245–260.

Let's Encrypt. (2024). ACME protocol and SSL certification automation. Retrieved from https://letsencrypt.org

Miao, X., Wang, T., & Zhang, H. (2021). Reverse proxy techniques for secure web deployment. *Elsevier Computers & Security, 101*, 102001.

Nguyen, T. H., & Kim, J. Y. (2022). Load balancing and traffic protection using HAProxy. *IEEE Transactions on Network and Service Management, 19*(4), 2956–2964.

OWASP. (2024). Security headers cheat sheet. Retrieved from https://owasp.org/www-project-secure-headers/

Raza, H., Akhtar, A., & Nasir, M. (2021). Performance evaluation of open-source firewalls: A case study of PFSense. *International Journal of Network Security & Its Applications (IJNSA), 13*(5), 57–70.

SecurityHeaders.com. (2024). Web security analysis tool. Retrieved from https://securityheaders.com

Singh, R., Kumar, A., & Bansal, M. (2023). Benchmarking website security using HTTP header analysis. *International Journal of Computer Networks & Information Security*, *15*(2), 27–35.

Sridhar, A., & Rao, D. (2021). High availability web infrastructure using HAProxy in hybrid cloud. *Springer Advances in Computing*, 278–286.

Syafii, M. I., Azhar, M. R., & Lestari, D. (2023). Implementation of SSL and HAProxy on reverse proxy servers for website security. *Journal of Information and Computer Technology*, *9*(1), 1–8.

Verizon. (2023). *Data Breach Investigations Report (DBIR).* Retrieved from https://verizon.com/dbir

Zhou, J., Wu, Y., & Chen, L. (2022). Practical implementation of network security using PFSense. *International Journal of Computer Science and Network Security*, *22*(8), 33–40.