Journal of Computer Engineering, Electronics and Information Technology (COELITE)

Journal homepage: https://ejournal.upi.edu/index.php/COELITE

Double Proxy Architecture for Data Center DDoS Mitigation

Rahmawan Bagus Trianto[1], Kukuh Muhammad[2], Dwi Novia Prasetyanti[3]*

[1,2,3] Department of Informatics Engineering, Politeknik Negeri Cilacap, Indonesia
Correspondence E-mail: rahmawanbagustrianto@pnc.ac.id

ABSTRACT

The increasing frequency and complexity of Distributed Denial-of-Service (DDoS) attacks present serious challenges to data center availability. Conventional single-layer defense mechanisms are often insufficient to handle high-volume attacks, resulting in excessive resource consumption on critical servers. This study proposes a data center security architecture based on a double proxy system to enhance resilience. The architecture consists of two sequential reverse proxy layers, namely a public-facing proxy and an internal proxy, which collaboratively perform traffic filtering, request validation, and load isolation before forwarding requests to the application servers. Experimental evaluations were conducted under simulated TCP- and UDP-based DDoS attack scenarios to assess system performance. The results indicate that the proposed double proxy architecture significantly reduces CPU utilization by up to 76% and decreases network bandwidth consumption by up to 352 Mbps compared to a traditional single-proxy configuration. Memory and disk usage remain stable during attack conditions, demonstrating minimal performance overhead. Furthermore, the architecture effectively absorbs malicious traffic, prevents direct exposure of backend servers, and preserves data integrity. These findings suggest that the double proxy approach provides a practical, efficient, and scalable solution for mitigating modern DDoS attacks in data center environments and can be adapted to evolving cyber threat landscapes.

ARTICLE INFO

Article History:
Submitted/Received 07 Feb 2026
First Revised 20 Feb 2026
Accepted 17 Mar 2026
First Available online 01 Apr 2026
Publication Date 01 Apr 2026

Keyword:
Data Center Security,
DDoS Mitigation,
Double Proxy,
Network Resilience,
Reverse Proxy.

1. INTRODUCTION

A Distributed Denial of Service (DDoS) attack is a cyberattack designed to make an online service inaccessible by inundating it with a deluge of internet traffic. It's like flooding a highway, causing gridlock and preventing legitimate traffic from passing through [1]–[3]. The increasing sophistication of cyber threats, particularly Distributed Denial of Service (DDoS) attacks, has made data centers vulnerable to unprecedented levels of disruption. DDoS attacks exploit network bandwidth, processing power, or application resources to overwhelm targeted systems and render them unresponsive. The consequences of such attacks can be severe, including financial losses, damage to brand reputation, and loss of customer trust [4]–[6]. As a result, the need for effective DDoS mitigation strategies has become more critical than ever.

Existing single-proxy solutions offer basic DDoS protection but are increasingly ineffective against sophisticated volumetric and application-layer attacks. The need for a more robust, multi-layered defense is critical. This study aims to address this gap by proposing a double proxy architecture that enhances traffic filtering and reduces the burden on backend servers.

The application of proxy servers as a defense mechanism against Distributed Denial of Service (DDoS) attacks. Proxy servers act as intermediaries, filtering and inspecting inbound traffic to identify and block malicious requests, while forwarding legitimate traffic to the backend server. Furthermore, proxy servers can be configured to implement rate-limiting policies, thus preventing excessive traffic from reaching the server [7]–[10].

The research compares the performance of Network Load Balancing (NLB) and High Availability Proxy (HAProxy) in mitigating Distributed Denial of Service (DDoS) attacks, specifically SYN floods. NLB is utilized on Windows Server 2016 with IIS 10.0, while HAProxy is employed on Ubuntu 16.04 Linux with Apache 2. Experiments were conducted in a live network environment to evaluate the efficiency of each load balancer. Average response time and CPU utilization were used as performance metrics. The findings reveal that NLB on Windows platforms exhibits superior performance in mitigating SYN flood DDoS attacks compared to HAProxy on Linux platforms [4][11][12].

The other research at Torabelo Hospital, the current server system faces security and optimization challenges. This research analyzes the impact and recommends solutions to enhance server security and optimization. Research findings indicate that the server system is vulnerable to various types of attacks and performance degradation. This can negatively impact hospital operations and jeopardize patient care. The recommended solution is to implement Squid as a reverse proxy, WAF (Web Application Firewall), and Snort as an IDS (Intrusion Detection System). System testing demonstrates that this solution effectively detects and prevents various common attacks [13]–[15].

Next paper investigates the role of proxy servers in educational settings and proposes innovative enhancements to optimize their performance and functionality. The study explores various cache optimization strategies to accommodate diverse organizational requirements and introduces methods for automatically reducing student bandwidth consumption, thereby enabling more efficient allocation of bandwidth resources for administrative tasks. The paper focuses on improving cache management, bandwidth control, and network traffic management through user and thread management. A practical implementation of a basic proxy server, Peroxy, is presented, demonstrating the potential of integrating the proposed features into an academic proxy server. Peroxy showcases the

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149

feasibility of efficient and effective Internet bandwidth management for institutions, preventing excessive bandwidth consumption by students and employees [8][16].

The study examines DDoS attacks that overload network resources and backend servers. A proxy-based architecture was implemented to manage traffic by dynamically adding servers during traffic spikes. However, attackers can redirect DDoS attacks to the proxy servers once their IP addresses are exposed. To address this issue, the study proposes an authentication mechanism to ensure balanced request distribution and block unauthorized requests. Simulation results show that the proposed approach effectively detects and mitigates DDoS attacks [17].

The other research employing the robust ModSecurity web application firewall (WAF) provides comprehensive protection against a wide range of web application attacks and facilitates HTTP traffic monitoring. ModSecurity has demonstrated its efficacy in mitigating DDoS attacks, achieving a 100% success rate when evaluated against three attack tools: Bash scripts, Golden Eye, and HULK. Furthermore, the implementation of a reverse proxy enhances server security by creating an additional layer of defense. While reverse proxies cannot prevent attacks, they can increase the time it takes for an attacker to reach the target, as evidenced by the average 53ms increase in attack duration using the three aforementioned tools. The combination of ModSecurity and a reverse proxy offers a dual-layered defense mechanism against DDoS attacks [18][19].

Data centers, as the backbone of modern digital infrastructure, are increasingly targeted by DDoS attacks that can disrupt critical services and cause significant operational impacts. While various mitigation techniques have been developed, most existing studies rely on single-layer proxy solutions. To address this limitation, this study proposes a double proxy architecture that combines a public reverse proxy and a local reverse proxy with a custom firewall [20]. This layered approach aims to improve system resilience by filtering malicious traffic before it reaches core systems and providing stronger protection against volumetric and application-layer DDoS attacks.

2. METHODS

This research aims to enhance the security of data centers and networks by proposing a novel architecture. The research process involved the following stages: conducting a literature review, designing the data center and network security architecture, implementing the designed architecture, testing the design, and evaluating the results. Through a comprehensive evaluation, the effectiveness of the proposed architecture in mitigating various cyber threats was assessed. **Figure 1** provides a visual representation of the research process. The steps involved in this research are as follows:

Figure 1. Research workflow.

Figure 1 illustrates the research methodology, which encompasses a literature review, data center and network design, implementation, testing, and evaluation of the data center security architecture's performance against DDoS attacks. A detailed explanation of each stage will be provided in the subsequent sections.

2.1. Literature Review

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149

This research commences with a comprehensive literature review. The review serves to provide a thorough understanding of previous studies, identify knowledge gaps, and lay a solid foundation for the proposed research. The review focuses on the threats posed by DDoS attacks and the critical need for mitigation strategies in data centers to ensure the continued security of systems.

2.2. Data Center and Network Design

This study proceeded to design a secure data center and network architecture, innovating with a double proxy approach. This architecture utilizes both a public reverse proxy and a local reverse proxy, with an additional layer of security provided by a network firewall. **Figures 2 and 3** represent the initial state where a traditional single proxy configuration on the local network was used, prior to the introduction of the double proxy system.

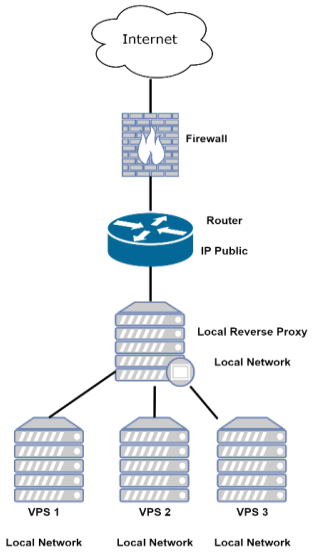

Figure 2. First state design without a public proxy and using the public IP at the router's public interface.

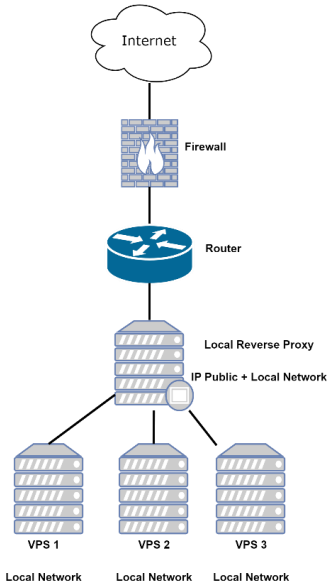

Figure 3. Second state design without public proxy and using IP Public at local proxy.

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149

Figure 2 illustrates the initial design without a public proxy. The public IP was assigned to the router's boundary interface, and a firewall was deployed on the router to provide an additional layer of security against DDoS attacks, in combination with the local proxy. **Figure 3** depicts the second design where the public proxy was omitted. The local proxy was assigned a public IP, allowing it to function on both the local and public networks. The local network serves as a reverse proxy to the VPS, while the public network enables global internet access. A firewall on the router provided an additional layer of security against DDoS attacks, working in tandem with the local proxy. The key difference between the two figures is the location of the public IP, which determines how incoming traffic is initially handled. In **Figure 2**, the router acts as the initial point of contact for public traffic, while in **Figure 3**, the local proxy assumes this role.

Figures 4 and 5 below illustrate the innovation proposed in this research, with a primary focus on the double proxy approach.

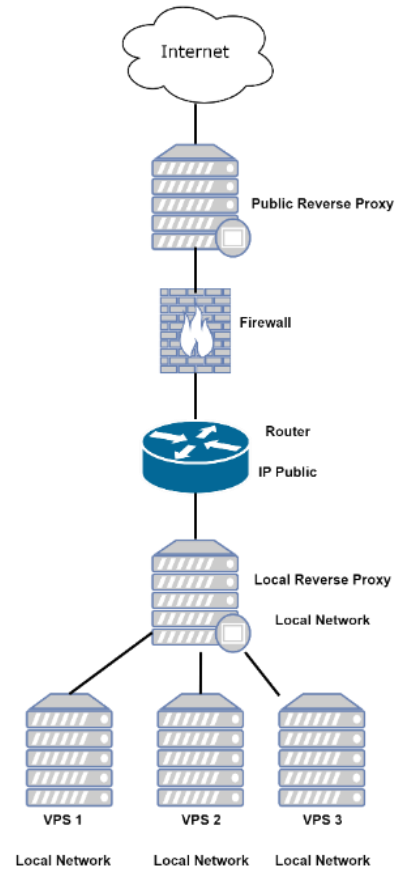

Figure 4. First proposed design with public proxy and using IP Public at router public interface.

Figure 4 presents the first network security architecture design to secure the data center using a double proxy. Both the public and local proxies are reverse proxies. The public proxy is placed above the firewall in the network architecture, aiming to protect against DDoS attacks originating from the internet towards applications on the local VPS. The local proxy is used for load balancing, hardening against local network DDoS attacks, and as a reverse proxy for requests to the local VPS. The router is configured with a public IP on the interface facing the internet.

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149

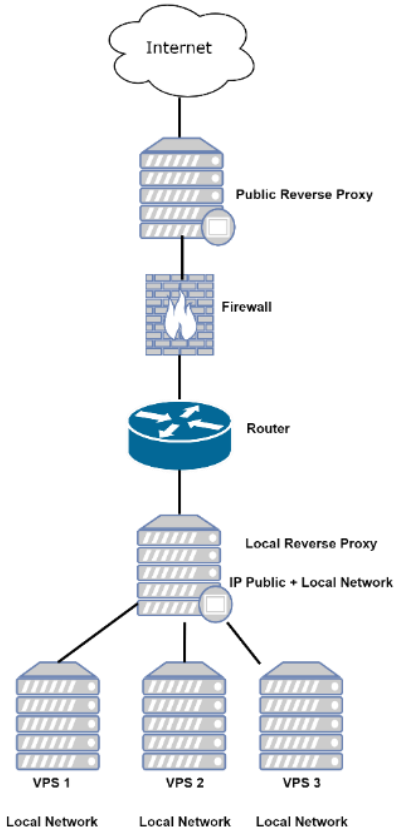

Figure 5. Second proposed design with public proxy and using IP Public at local proxy.

Figure 5 presents the second network security architecture design to secure the data center using a double proxy. Both the public and local proxies are reverse proxies. The public proxy is placed above the firewall in the network architecture, aiming to protect against DDoS attacks originating from the internet towards applications on the local VPS. The local proxy is used for load balancing, hardening against local network DDoS attacks, and as a reverse proxy for requests to the local VPS. The local proxy is configured with a public IP directly connected to the internet. The difference between the two figures lies in the placement of the public IP address. In **Figure 4**, the public IP is located on the router, while in **Figure 4**, it is placed on the local proxy.

2.3. Design Implementation

The previously designed data center and network were implemented on physical servers and virtualized environments. By using server virtualization techniques, it can improve security, reliability, flexible resource usage and better scalability [21][22]. The data center server had the following specifications: 16 x Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz, 32 GB Memory, and 2TB Hard disk. To test the design model, a local reverse proxy with 2 Cores, 1 GB of memory, and 3 GB of hard disk was used. The local reverse proxy utilizes the Nginx web server, which offers better performance compared to Apache [5][23][24]. The local reverse proxy was configured for security by applying hardening techniques. Additionally, a virtual server was used as a local server with 1 Core, 512 MB Memory, and 3 GB Hard disk. All servers, including the data center, local reverse proxy, and virtual server, were connected to the local network. This was also a security measure for the data center, as access to the virtual server from the public network or the internet required going through the local reverse proxy. For the public reverse proxy, the highly reliable third-party platform Cloudflare was employed

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149

[2]. For the network model used in this study, the design of the network topology uses a tree topology, which is widely used and has many advantages, is more efficient, and easier to maintain [25].

2.4. Testing the Design

The fourth phase of this research consists of testing the designed and deployed system. Building upon the second step, four data center and network security architectures will undergo comparative analysis under two attack scenarios, employing both TCP and UDP protocols. The testing will involve subjecting the system to DDoS attacks utilizing both TCP and UDP protocols [4][7][18][17], with each attack lasting for 30 seconds. The study will evaluate the impact of these attacks on the data center, local reverse proxy, local virtual server, and network bandwidth. The performance of the third-party public reverse proxy will not be within the scope of this research. The primary focus will be on assessing the impact on the internal data center. The DDoS attacks will be conducted using the MHDDoS tool, a tool that provides 56 attack methods, including TCP and UDP-based attacks, which will be evaluated in this study. **Figure 6** provides a visual representation of the MHDDoS tool interface and its configuration for the attack testing phase.

Figure 6. MHDDoS tool for attack testing.

MHDDoS is a command-line-driven tool that is developed using the Python programming language. It can attack at layer 4 and layer 7 using various methods. The tool's popularity is evident from its over 13,000 stars on GitHub, attesting to its robust capabilities. Attack scenarios were designed utilizing MHDDoS across two primary protocols (TCP and UDP) with a constant attack intensity (30 seconds, 20 threads). The effectiveness of traffic filtering was evaluated not only based on CPU load and bandwidth utilization but also on the system's success in blocking packets before reaching the VPS. Evaluation criteria included: CPU utilization on the router and proxies, bandwidth usage, latency (not directly tested in this research but proposed as a future metric), and service request success rate.

2.5. Evaluation

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149

The research concludes with an evaluation phase. This phase involves a comprehensive assessment of the experimental results to compare the performance of each designed data center security architecture. The objective is to identify the most suitable design for institutional implementation. However, it is essential to note that ongoing security enhancements will be required to ensure sustained user satisfaction, data integrity, and overall system security.

3. RESULTS AND DISCUSSION

DDoS attack tests were conducted using both TCP and UDP protocols for each of the previously mentioned architecture designs. Each attack was carried out for 30 seconds on each architectural design. The attacks utilized 20 threads, enabling the transmission of numerous large data packets. The DDoS attacks consumed approximately 300-400 Mbps of bandwidth. In the DDoS attack trials using the TCP protocol, packet sizes varied between 11,000 and 44,000 packets per second. From this packet count, the data size sent was equivalent to the attacker's server bandwidth, ranging between 350 Mbps and 400 Mbps, which is equivalent to approximately 45 Mbps. The DDoS attack trials were conducted sequentially under identical environmental conditions on all four architectural designs depicted in **Figures 2, 3, 4, and 5** above using the MHDDoS tool. The research results (the average CPU load was taken, and if it was below 2%, it was written as NaN, as it was not considered a DDoS attack) are detailed in **Table 1** below.

Table 1. Result of DDoS attack testing.

No	Attack	Design	Router			Data Center		Local Proxy		Local VPS	
			CPU	BW-1	BW-2	CPU	BW	CPU	BW	CPU	BW
1	TCP	1	67%	120M	120M	5.7%	42M	5.2%	42M	NaN	< 1K
2	UDP	1	80%	352M	< 1K	NaN	< 1K	NaN	< 1K	NaN	< 1K
3	TCP	2	67%	120M	120M	3.2%	36M	5.4%	120M	NaN	< 1K
4	UDP	2	80%	352M	351M	NaN	346M	4.6%	< 1K	NaN	< 1K
5	TCP	3	4%	< 1K	< 1K	NaN	< 1K	NaN	< 1K	NaN	< 1K
6	UDP	3	4%	< 1K	< 1K	NaN	< 1K	NaN	< 1K	NaN	< 1K
7	TCP	4	4%	< 1K	< 1K	NaN	< 1K	NaN	< 1K	NaN	< 1K
8	UDP	4	4%	< 1K	< 1K	NaN	< 1K	NaN	< 1K	NaN	< 1K

The TCP-based DDoS attack on both architecture designs showed that multi-threaded attacks significantly increased CPU usage on the local proxy while having no impact on the target VPS, as the attacks were successfully terminated at the preconfigured proxy layer. In architecture design 1, the attack generated approximately 120 Mbps of bandwidth on the internet interface, with 42 Mbps reaching the local proxy network and increasing the proxy CPU load to 5.2% from a pre-attack level of around 0.03%. In architecture design 2, similar behavior was observed, with 120 Mbps traffic forwarded to the local proxy network and the proxy CPU load reaching 5.4%. In both cases, the router experienced an increase in CPU utilization to an average of 67%, depending on the number of attack threads and the attacker's bandwidth capacity.

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149

The UDP-based DDoS attack also increased system resource usage but remained contained at the local proxy layer, preventing any impact on the target VPS. In architecture design 1, the attack caused an average bandwidth increase of approximately 352 Mbps on the internet interface, although the traffic was not forwarded to the local proxy network. The maximum CPU load on the local proxy reached 3.7%, slightly higher than the pre-attack level. In architecture design 2, the UDP attack produced similar bandwidth spikes, with traffic forwarded to the local proxy network and proxy CPU usage reaching 4.6%. For both architectures, router CPU utilization increased significantly, averaging around 80%, indicating that the router handled most of the attack traffic load.

DDoS attacks utilizing both TCP and UDP protocols on the proposed data center security architectures (**Figures 4 and 5**) demonstrated significant improvements. The public proxy placed above the firewall exhibited exceptional performance, effectively preventing DDoS attacks from reaching the firewall layer and below. This is evident from the absence of any spikes in CPU load and bandwidth on the router, data center, local proxy, or local VPS. Both proposed data center security architectures demonstrated equally impressive performance. The only difference lies in the placement of the public IP, which is located on the router in the design depicted in **Figure 4** and on the local proxy in the design depicted in **Figure 5**. This provides stakeholders with options that can be implemented in their network security systems.

To further illustrate these improvements, **Figure 7** presents a comparative analysis between design 1 and design 3, highlighting the performance gap under identical attack conditions.

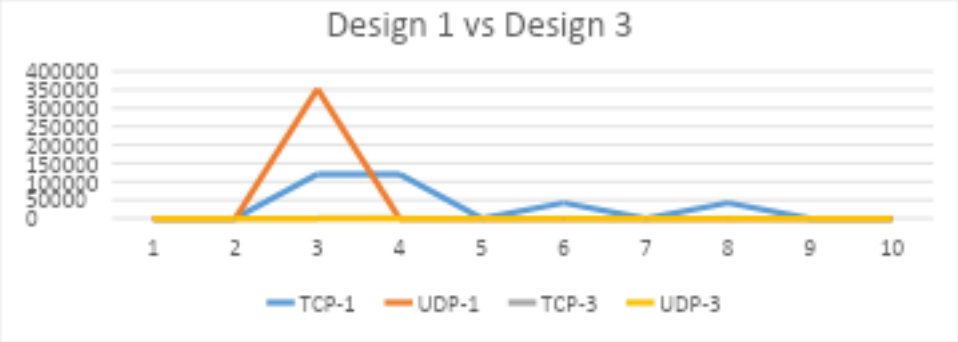

Figure 7. Comparison between design 1 and design 3.

A comparison between design 3 and design 1 (the initial design without a double proxy) in mitigating DDoS attacks using the TCP protocol reveals a significant performance improvement. Specifically, there was a 63% reduction in the router's CPU load, a 120 Mbps bandwidth saving on the internet-connected interface, a 120 Mbps bandwidth saving on the local network, a 5.7% reduction in overall data center CPU load, a 42 Mbps bandwidth saving on the data center, and a 5.2% reduction in local proxy CPU load with a 42 Mbps bandwidth saving. When comparing design 4 and design 2 in mitigating DDoS attacks using the UDP protocol, there was a 76% improvement in the router's CPU load and a 352 Mbps bandwidth saving on the internet-connected interface.

A detailed side-by-side comparison of these performance metrics between design 4 and design 2 is visualized in **Figure 8**.

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149

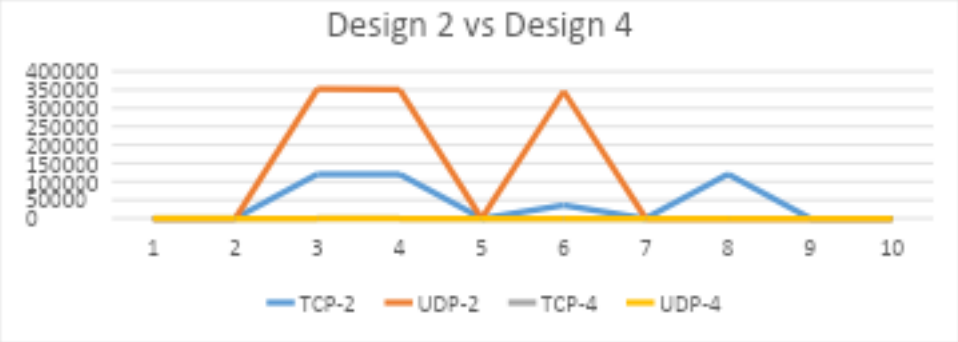

Figure 8. Comparison between design 2 and design 4.

Comparing designs 4 and 2, the new design (5) demonstrated a significant improvement in mitigating DDoS attacks. Using TCP, design 5 reduced router CPU load by 63%, saved 120 Mbps of bandwidth on both the internet and local networks, and decreased data center and local proxy CPU load and bandwidth by 3.2% and 5.4%, respectively, with corresponding bandwidth savings. For UDP attacks, design 5 reduced router CPU load by 76%, saved 352 Mbps of bandwidth on the internet, and significantly reduced bandwidth consumption on the local network and data center. To enhance the clarity and impact of the findings, four bar charts have been added to illustrate the resource usage across the four architectural designs. These graphs allow for direct visual comparison between traditional single-proxy designs and the proposed double-proxy implementations. The comprehensive data for these metrics are visualized in **Figures 9 and 10**, which display the router and data center resource usage, respectively.

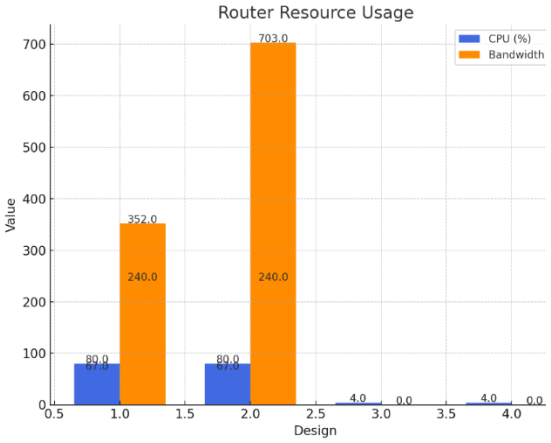

Figure 9. Router Resources Usage.

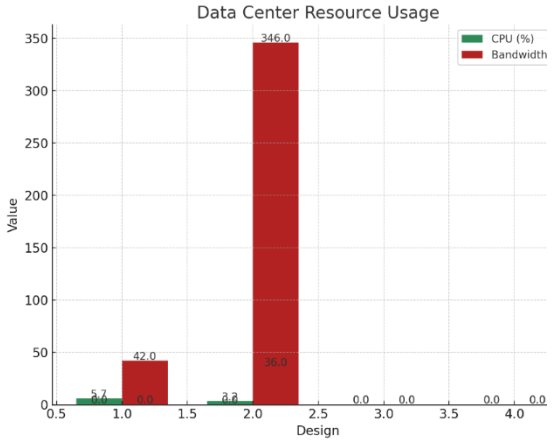

Figure 10. Data Center Resource Usage.

Figure 9 highlights the CPU and bandwidth utilization of the router, designs 1 and 2, CPU usage reaches 67%–80%, and bandwidth spikes to 703 MB. Designs 3 and 4, CPU usage drops to 4%, and bandwidth consumption is 0 MB. In single proxy configurations, the router is the first component to face the full intensity of a DDoS attack. The double proxy setup, especially with a cloud-based public proxy, effectively shields the router, drastically reducing its load.

Figure 10 displays CPU and bandwidth usage at the Data Center server. Designs 1 and 2 show CPU usage of up to 5.7% and bandwidth consumption as high as 346 MB. Designs 3 and 4 show no measurable resource usage, with values at or near zero. Without the double proxy system, malicious traffic is able to reach the data center server, causing performance degradation. The double proxy structure effectively intercepts and blocks harmful traffic before it impacts core infrastructure. Furthermore, the efficiency of this mitigation strategy

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149

across other critical components is detailed in **Figures 11 and 12**, which illustrate the local proxy resource usage and local VPS bandwidth usage, respectively.

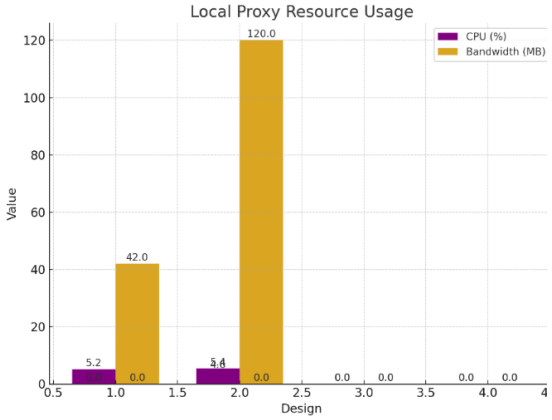

Figure 11. Local Proxy Resource Usage.

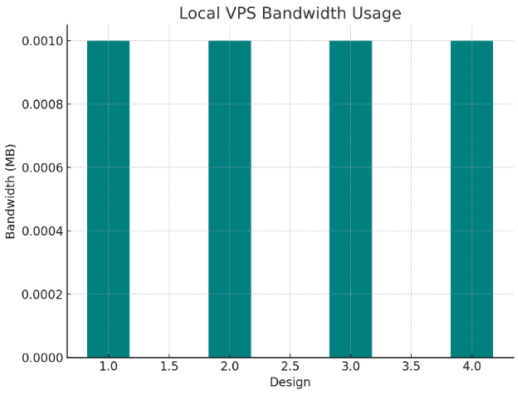

Figure 12. Local VPS Bandwidth Usage.

Figure 11 compares the CPU usage and bandwidth consumption of the Local Proxy. Designs 1 and 2 (without a public proxy) show CPU loads of up to 5.2%–5.4% and bandwidth usage reaching 120 MB. Designs 3 and 4 (with a double proxy setup) show near-zero CPU and bandwidth usage. In single proxy designs, the local proxy directly absorbs the DDoS attack load. The addition of a public reverse proxy in Designs 3 and 4 shifts that burden to the outer layer, allowing the local proxy to focus on legitimate traffic only.

Figure 12 illustrates the bandwidth usage of the Local VPS across the four architectural designs (Designs 1 to 4). The bandwidth recorded for all designs is nearly 0.001 MB, indicating minimal data transmission to the local VPS. All designs successfully prevent DDoS traffic from reaching the Local VPS. However, the double proxy architectures (Designs 3 and 4) enhance traffic isolation and filtering, ensuring that the VPS remains fully protected without experiencing any significant load.

The proposed double proxy design outperforms the single-proxy models due to its layered defense mechanism. The public reverse proxy, hosted via Cloudflare, acts as a buffer zone that absorbs volumetric traffic and blocks known threats even before reaching the institutional network. The local reverse proxy further inspects, filters, and balances the incoming requests within a secure internal boundary. This two-tier separation significantly reduces load on downstream components (router, local proxy, data center), as malicious traffic is halted in earlier stages. In contrast, single-proxy systems expose internal resources more directly to attack loads. Implementing the double proxy architecture provides substantial long-term benefits. It improves operational stability by reducing hardware stress, which helps extend the lifespan of network equipment. In addition, the architecture enables resource savings by lowering CPU and bandwidth demands, thereby reducing energy consumption and infrastructure costs. The modular design also enhances scalability, allowing the system to be adapted to various network sizes and environments. Furthermore, the implementation strengthens the overall security posture by establishing a multi-layer protection mechanism that aligns with zero-trust principles and supports compliance with cybersecurity standards.

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149

4. CONCLUSION

This study proposed and evaluated a double proxy architecture to enhance data center security against DDoS attacks. The results show that the proposed approach improves system performance compared to traditional single-proxy designs, achieving CPU usage reductions of up to 76% and bandwidth savings of up to 352 MB. By combining a public reverse proxy and a local reverse proxy, the architecture effectively filters malicious traffic and maintains system stability during volumetric attacks.

However, this study has several limitations. The evaluation focused only on TCP and UDP-based DDoS attacks (Layer 3 and 4), while the effectiveness against more complex Layer 7 attacks was not examined. In addition, scalability in large-scale enterprise or multi-site data center environments has not yet been evaluated. Future research should explore the integration of machine learning-based traffic classification and anomaly detection, as well as real-world deployment to further assess system performance, reliability, and operational feasibility.

5. ACKNOWLEDGMENT

This study is supported by the Department of Computer and Business, Politeknik Negeri Cilacap, in providing the necessary network equipment.

6. AUTHORS' NOTE

The authors declare that there is no conflict of interest regarding the publication of this article. The authors confirmed that the paper was free of plagiarism.

7. REFERENCES

[1] Wikrama, K. S. M., Firdaus, R., Mendrofa, L. Z. M., Saskara, G. A. J., & Listartha, I. M. E. (2023). DDoS Attack Using GoldenEye, DAVOSET, and PyLoris Tools. Jurnal CoreIT: Jurnal Hasil Penelitian Ilmu Komputer dan Teknologi Informasi, 9(2), 43–52. https://doi.org/10.24014/coreit.v9i2.20020.

[2] Imthiyas, M., Wani, S., Abdulkhaleq, R., Abdulghafor, A., Ibrahim, A. A., & Hafeez, A. (2020). DDoS Mitigation: A review of Content Delivery Network and its DDoS Defence techniques. International Journal of Perceptive and Cognitive Computing, 6(2), 67–76. https://journals.iium.edu.my/kict/index.php/IJPCC/article/view/160.

[3] Wani, S., Imthiyas, M., Almohamedh, H., Alhamed, K. M., Almotairi, S., & Gulzar, Y. (2021). Distributed Denial of Service (DDoS) Mitigation Using Blockchain—A Comprehensive Insight. Symmetry, 13(2), 1–21. https://doi.org/10.3390/sym13020227.

[4] Zebari, R. R., Zeebaree, S. R. M., Sallow, A. B., Shukur, H. M., Ahmad, O. M., & Jacksi, K. (2020). Distributed Denial of Service Attack Mitigation using High Availability Proxy and Network Load Balancing. 3rd International Conference on Advanced Science and Engineering (ICOASE), 174–179. https://doi.org/10.1109/ICOASE51841.2020.9436545.

[5] Saputra, P. S., Pratama, P. A., & Tjahyanti, L. P. A. S. (2023). Perancangan Dan Komparasi Web Server Nginx Dengan Web Server Apache Serta Pemanfaatan Reverse Proxy Server Pada Nginx. Jurnal Komputer dan Teknologi Sains, 2(1), 16–21.

[6] Singh, A., & Gupta, B. B. (2022). Distributed Denial-of-Service (DDoS) Attacks and Defense Mechanisms in Various Web-Enabled Computing Platforms. International Journal on Semantic Web and Information Systems, 18(1), 1–43. https://doi.org/10.4018/IJSWIS.297143.

[7] Bamane, A. (2023). Detecting and Mitigating Network Attacks Using Network Simulator 3. International Journal of Innovative Research in Technology, 9(12), 1285–1292.

[8] Dangi, R., Jha, V. K., Choudhary, M., & Bhavsar, P. D. (2019). An Enhanced Proxy Server for Better Performance & Security of the Network for Academics. International Journal for Research in Applied Science and Engineering Technology, 7(6), 278–281. https://doi.org/10.32622/ijrat.76S201955.

[9] Shah, Z., Ullah, I., Li, H., Levula, A., & Khurshid, K. (2022). Blockchain-Based Solutions to Mitigate Distributed Denial of Service (DDoS) Attacks in the Internet of Things (IoT): A Survey. Sensors, 22(3), 1–26. https://doi.org/10.3390/s22031094.

[10] Wright, M., Venkatesan, S., Albanese, M., & Wellman, M. P. (2016). Moving Target Defense against DDoS Attacks. Proceedings of the 2016 ACM Workshop on Moving Target Defense, (October), 93–104. https://doi.org/10.1145/2995272.2995279.

[11] Karnani, S., & Shakya, H. K. (2023). Mitigation strategies for distributed denial of service (DDoS) in SDN: A survey and taxonomy. Information Security Journal: A Global Perspective, 32(6), 444–468. https://doi.org/10.1080/19393555.2022.2111004.

[12] Wang, J., Wang, L., & Wang, R. (2023). A Method of DDoS Attack Detection and Mitigation for the Comprehensive Coordinated Protection of SDN Controllers. Entropy, 25(8), 1–26. https://doi.org/10.3390/e25081210.

[13] Rifai, M. F., Hendra, S., Ngemba, H. R., Azhar, R., & Laila, R. (2024). Enhancing Web Server Security against Layered Cyber Threats in Healthcare. Advances in Sustainable Science and Engineering Technology, 6(2), 1–9. https://doi.org/10.26877/asset.v6i2.18307 .

[14] Orosz, P., Nagy, B., & Varga, P. (2025). Real-Time Detection and Mitigation Strategies Newly Appearing for DDoS Profiles. Future Internet, 17(9), 1–32. https://doi.org/10.3390/fi17090400.

[15] Khan, F. F., Hossain, N. M., Shanto, M. N. H., Anwar, S. B., & Noor, J. (2022). Mitigating DDoS Attacks Using a Resource Sharing Network. Proceedings of the 9th International Conference on Networking, Systems and Security, (December), 1–11. https://doi.org/10.1145/3569551.3569560.

[16] Ouhssini, M., Afdel, K., Akouhar, M., Agherrabi, E., & Abarda, A. (2024). Advancements in detecting, preventing, and mitigating DDoS attacks in cloud environments: A comprehensive systematic review of state-of-the-art approaches. Egyptian Informatics Journal, 27(100517), 1–37. https://doi.org/10.1016/j.eij.2024.100517.

[17] Dharam, P., & Musarrat, J. (2020). An Authentication Technique to Handle DDoS Attacks in Proxy-Based Architecture. ICNS 2020: The Sixteenth International Conference on Networking and Services, (16), 49–54.

[18] Zain, A. R., Muhamad, I., Matin, M., & Kautsar, D. K. (2023). Analisis Implementasi Modsecurity dan Reverse Proxy Untuk Pencegahan Serangan Keamanan DDoS pada Web Server. SNIV Seminar Nasional Inovasi Vokasi, 2(1), 118–127.

[19] Mahjabin, T., Xiao, Y., Sun, G., & Jiang, W. (2017). A survey of distributed denial-of-service attack, prevention, and mitigation techniques. International Journal of Distributed Sensor Networks, 13(12), 1–33. https://doi.org/10.1177/1550147717741463.

[20] Amalia, E. R., Nurheki, Saputra, R., Ramadhana, C., & Yossy, E. H. (2022). Computer network design and implementation using a load balancing technique with per connection classifier (PCC) method based on a MikroTik router. Procedia Computer Science, 216, 103–111. https://doi.org/10.1016/j.procs.2022.12.116.

[21] Satwika, I. K. S., & Andika, I. G. (2024). Performance Analysis of an E-commerce Website

Using Distributed Servers (Case Study: E-commerce Bumdes Sarining Kukuh Winangun). Journal of Computer Networks, Architecture and High Performance Computing, 6(3), 1390–1398. https://doi.org/10.47709/cnahpc.v6i3.4383.

[22] Manalu, A. S., & Sitanggang, S. S. (2019). Perancangan Dan Implementasi Private Cloud Storage Dengan Owncloud Pada Jaringan Lokal Menggunakan Virtualbox. Journal of Computer Networks, Architecture and High Performance Computing, 1(2), 60–71. https://doi.org/10.47709/cnahpc.v1i2.244.

[23] Apriliansyah, F., Fitri, I., & Iskandar, A. (2020). Implementasi Load Balancing Pada Web Server Menggunakan Nginx. Jurnal Teknologi dan Manajemen Informatika, 6(1), 18–26. https://doi.org/10.3997/2214-4609.201801770.

[24] Wijaya, G., & Franklyn, F. V. (2021). Perancangan Dan Implementasi Desain Arsitektur Cloud Yang Aman Untuk Sistem Stok Pada Halutime Thrift Shop. Conference on Business, Social Sciences and Technology (CoSST), 1(1), 39–46. https://journal.uib.ac.id/index.php/conescintech/article/view/5828.

[25] Fa'iz, N. F., Irfawan, D. R., Putri, A. W., & Hidayatullah, S. (2024). Arsitektur Jaringan Menggunakan Topologi Tree. Jurnal Riset Sistem Informasi dan Teknik Informatika (Merkurius), 2(3), 98–104. https://doi.org/10.61132/merkurius.v2i3.112.

DOI: https://doi.org/10.17509/coelite.v5i1.98157
p- ISSN 2829-4157 e- ISSN 2829-4149