IJITCE

ISSN 2347- 3657

International Journal of Information Technology & Computer Engineering

Email : ijitce.editor@gmail.com or editor@ijitce.com

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 1, 2025

Fortifying Cyber Defenses: Empowering Web Application Firewalls Through Threat Intelligence

Dr.A. Krishna Chaitanya[1], Gopalam Jignyasa[2], Sundarapalli Keerthi[3] , Gurram Himesh Reddy[4]

Asst. Professor Computer Science and Engineering Institute of Aeronautical Engineering Dundigal, Hyderabad

a.krishnachaitanya@iare.ac.in[1], 21951A6213@iare.ac.in[2], 21951A6215@iare.ac.in[3], 21951A6211@iare.ac.in[4]

***Abstract*— In today's cyber landscape, the proliferation of sophisticated threats necessitates a proactive approach to safeguarding web applications. The proposed method aims to integratethreat intelligence feeds, specifically focusing on AlienVault OTX, with the robust capabilities of ModSecurity firewall software. By leveraging real-time threat data and implementing it within the firewall infrastructure, organizations can significantly enhance their defense mechanisms against evolving cyber threats. This integration will enable the firewall to dynamically analyze incoming traffic patterns, detect possible threats, and implement preventive actions to reduce risks in real-time. Techniques for efficiently processing and prioritizing threat intelligence data will be explored, ensuring optimal performance andminimal latency in threat detection and response. The ultimate goal is to fortify cyber defenses by empowering web application firewalls through enhanced threat intelligence integration.**

***Index Terms*— Web Application Security, Web Application Firewall, Threat Intelligence Feeds, Cyber threats, Threat Intelligence Integration.**

I.INTRODUCTION

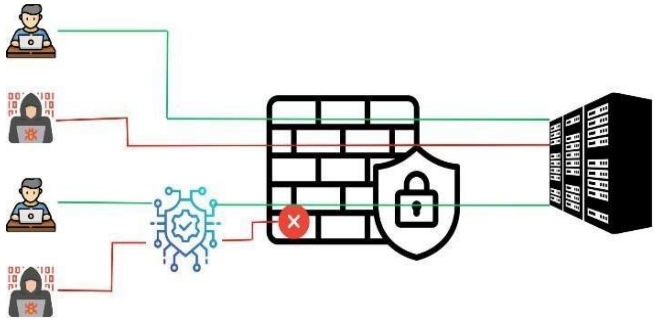
The rapid expansion of web applications across industries like e-commerce, banking, and healthcare has greatly broadened the range of potential targets for cyber-attacks. Since web applications are easily accessible online and often built with intricate, sometimes error-prone code, they are especially at risk. Traditional network firewalls, designed to defend against network-specific threats, fall short when it comes to safeguarding these applications from attacks that leverage vulnerabilities in web protocols like HTTP and similar technologies.

Web Application Firewalls (WAFs) have become essential tools for protecting web applications. Designed specifically to monitor and filter HTTP traffic, WAFs help defend against frequent webbased attacks, such as cross-site scripting(XSS), remote file inclusion and SQL injection. Despite their effectiveness, the fast-paced evolution of cyber threats means that WAFs need regular updates with the latest threat intelligence to maintain their protective capabilities.

ModSecurity, a widely-used open-source WAF, is recognized for its strong defense against known vulnerabilities, relying on a signature-based approach. While powerful, ModSecurity and similar WAFs face limitations due to their static rule sets, which lack integration with real-time threat intelligence, potentially limiting their adaptability to new threats.

Threat intelligence refers to the gathering and examination of information on potential security threats. This data is usually sourced from global threat databases, security research publications, and live data feeds. When integrated with WAFs, threat intelligence enhances the firewall's capacity toidentify and counter emerging, complex threats by supplyingtimely insights on the latest attack methods and techniques.

Integrating real-time threat intelligence feeds, like AlienVault OTX, into the ModSecurity firewall can strengthen the defenses of web applications. This setup allows ModSecurity to update its rule sets dynamically based on current threat data, boosting its effectiveness against new and evolving threats. The proposed system will monitor incoming traffic patterns, detect potential threats in real-time, and take proactive steps to reduce risks.



ISSN 2347–3657
Volume 13, Issue 1, 2025

Figure 1: Network Security Architecture

Figure 1 illustrates a network security setup in which a firewall is integrated with a threat intelligence feed to strengthen traffic filtering and protect a web server. Incoming traffic, whether from legitimate or malicious users, is first processed by the threat intelligence system, which detects potential threats. Traffic deemed legitimate, shown by green lines, is allowed to pass through the firewall and reach the server. On the other hand, malicious traffic, represented by red lines, is blocked by the firewall before it can reach the server. This integration enhances the firewall's ability to detect and prevent harmful traffic, ensuring that only safe requests are allowed, thereby boosting the overall security of the network.

By utilizing real-time threat intelligence, organizations can greatly strengthen their cybersecurity defenses, ensuring that their web applications are better shielded from the constantly changing threat landscape. Exploring the architecture, methodology, implementation, and outcomes of this integration highlights how combining threat intelligence with Web Application Firewall (WAF) capabilities can create more resilient and adaptable cyber defenses.

II.LITERATURE REVIEW

[1] In Adem Tekerek study titled "Development of a Hybrid Web Application Firewall to Prevent Web Based Attacks", the author highlights the dual purpose of firewall and intrusion detection systems: preventing information loss and vulnerabilities on the internet, while securing web applications. However, traditional security tools focused on the network layer prove inadequate against HTTP attacks targeting web applications. To address this gap, the study introduces a hybrid firewall system employing signature-based and anomaly detection methods tailored for HTTP detection. The findings reveal that while signature-based detection offers speed, it falls short against zero-day attacks, whereas anomaly detection proves effective against such threats.

[2] Firkhan Ali Bin Hamid Ali's (2011) study, "A Study of Technology in Firewall System" explores the significance of logging and reporting within firewall systems, emphasizing their pivotal role in enhancing effectiveness alongside technological advancement. While delving into the historical progressions within firewall technology, the paper falls short in explicitly addressing contemporary trends, potentially leading to incomplete evaluations of firewall efficacy amid the dynamic landscape of cybersecurity threats.

[3] Lin Zhang (2015) presents "A Firewall Rules Optimized Model Based On Service-Grouping," which introduces a model for merging firewall policy rules through rule-service grouping, aiming to diminish the number of rules and filtering times without compromising firewall performance. However, the model operates under the assumption that rule sets are devoid of anomalies or inconsistencies. In practical settings, firewall rule sets often harbor errors, conflicts, or ambiguities, potentially impeding the merging process and diminishing the overall efficacy of the firewall policy.

[4] In their 2015 study, Z. Ghanbari examines a "Comparative approach to Web Application Firewalls (WAFs)", underscoring their significance in mitigating the detrimental impacts of cyber- attacks, including financial losses, credibility damage, and risks to organizational and national interests. However, it is highlighted that WAFs have the propensity to erroneously block valid traffic if not configured correctly. This occurs when they mistakenly categorize legitimate requests as malicious, often as a result of deviations from standard HTML/HTTP protocols or unconventional content formats.

III.WEB APPLICATION SECURITY

Web application security is essential for safeguarding web applications against various cyber threats. A Web Application Firewall (WAF) is often used as a protective tool, but traditional WAFs struggle to keep up with rapidly evolving threats due to their reliance on static rule sets. This limitation makes them less effective at defending against zero-day vulnerabilities and new attack methods. Additionally, false positives, where legitimate user actions are incorrectly flagged as threats, can disrupt operations and negatively impact the user experience.

To improve web application security, a WAF can be integrated with a threat intelligence feed. This integration allows the firewall to dynamically match incoming traffic with known malicious indicators, enhancing the accuracy of threat detection. By minimizing false positives, legitimate user activity is less likely to be blocked, ensuring the WAF offers more dependable protection against both known and emerging threats. Real-time updates from threat intelligence keep the firewall informed about the latest malicious activities targeting web applications, strengthening the defense against evolving cyber threats.

Threat intelligence feeds offer valuable insights into malicious actors, attack signatures, indicators of compromise (IoCs), and emerging cyber threats. To integrate this data effectively, the firewall uses event-driven triggers or scheduled tasks to

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 1, 2025

automatically update its rule sets based on the latest threat intelligence. By automating these updates, the firewall reduces the need for manual intervention, enabling it to quickly respond to new threats and enhance the overall security of web applications.

The objectives of integrating the firewall include:
The integration of the firewall aims to enhance the capabilities of the web application firewall (WAF) by enabling it to proactively defend against evolving cyber threats, including zero-day vulnerabilities and emerging attack methods. This involves the development of advanced threat detection mechanisms and the continuous refinement of rule sets, which are updated using real-time global threat intelligence. Such updates help maintain the firewall's agility and responsiveness to new and evolving threats.

Another key objective is to minimize false positives by employing advanced traffic analysis techniques and intelligent algorithms. These methods allow the WAF to more accurately distinguish between legitimate user activities and potential threats, significantly reducing disruptions caused by false alarms. Additionally, enhanced behavioral analysis and contextual awareness further strengthen the WAF's ability to adapt its security measures. This ensures that the rule sets are finely tuned, enabling the firewall to filter out malicious traffic effectively while still allowing legitimate traffic to pass smoothly, thereby maintaining a seamless user experience.

IV. DYNAMIC THREAT INTELLIGENCE INTEGRATION

The methodology combines ModSecurity with a powerful threat intelligence feed to enhance web application security. ModSecurity is initially deployed on a high-performance server running Parrot OS, where it intercepts and inspects HTTP requests handled by Nginx. The integration of real-time updates from the threat intelligence feed allows for dynamic rule adjustments, enabling the system to adaptively detect and respond to emerging threats.

A Security Information and Event Management (SIEM) system plays a crucial role in comprehensive logging and monitoring, correlating security events to deliver actionable insights. Extensive testing in controlled environments fine-tunes configurations, ensuring optimal performance. Once deployed in production, the system is continuously monitored and updated to maintain strong security measures against evolving threats. This approach allows the system to respond swiftly to new threats, further strengthening the security posture of web applications.

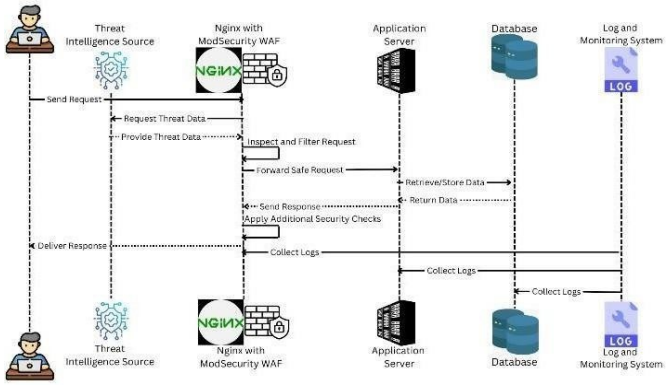


Figure 2: System Architecture

Figure 2 depicts a comprehensive web application architecture designed to ensure security, efficiency, and continuous monitoring. At the top, the Threat Intelligence Source provides real-time data on emerging threats, which is essential for keeping the system updated against new vulnerabilities.

This data is fed into Nginx, which is paired with ModSecurity to function as a Web Application Firewall (WAF). The WAF inspects and filters incoming traffic, blocking malicious requests based on the most current threat intelligence. Once traffic is deemed safe, it is forwarded to the Application Server, which processes user requests, handles backend logic, and interacts with the Database to store or retrieve data.

The Log and Monitoring System continuously collects and analyzes logs from all components, ensuring that the system's performance and security are consistently maintained. It also enables rapid responses to any detected anomalies. This integrated approach guarantees that users accessing the application via their browsers experience a secure, seamless interaction, protected by up-to-date security measures and supported by robust backend processing and monitoring systems.

ISSN 2347–3657
Volume 13, Issue 1, 2025

Rule-Based Detection

Rule-Based Detection strengthens web security by assessing network traffic using predefined rules instead of relying on specific attack signatures. These rules determine actions such as allowing, blocking, or flagging traffic based on factors like IP addresses, ports, HTTP methods, and payload content. Unlike signature-based methods, which focus solely on known attack patterns, rule-based systems within Intrusion Detection and Prevention Systems (IDPS) provide dynamic security responses to new and emerging threats. By integrating threat intelligence, organizations can continuously update rules to address evolving

vectors effectively. This approach enhances the agility of security systems, enabling rapid responses to both new and unknown threats that traditional methods may overlook. Rulebased detection facilitates precise traffic filtering, reducing risks from sophisticated cyber threats. Its flexibility also allows for customization according to operational needs and regulatory requirements, ensuring strong protection against a wide range of vulnerabilities and attack scenarios.

V.IMPLEMENTATION

The implementation of the Dynamic Threat Intelligence Integration (DTII) methodology involves several key steps to ensure the smooth integration of the ModSecurity Web Application Firewall (WAF) with the AlienVault OTX threat intelligence feed. This process is designed to improve the WAF's ability to detect and mitigate emerging cyber threats by dynamically updating its rule sets in real time.

A. ModSecurity Installation and Configuration

The initial step in the implementation involves installing and configuring ModSecurity on the web server. As an open-source Web Application Firewall (WAF), ModSecurity is integrated with the web application server, such as Apache or Nginx, to monitor and filter HTTP traffic. This process includes downloading the ModSecurity module, compiling it with the web server, and configuring it to begin filtering traffic. Basic security rules are then implemented to ensure that ModSecurity is operating properly.

B. AlienVault OTX Integration

To integrate threat intelligence feeds, the first step is to create an AlienVault OTX account. This involves registering on the AlienVault OTX platform and obtaining the necessary API keys to access the threat intelligence feeds. These API keys enable the system to retrieve real-time data on a variety of cyber threats, such as indicators of compromise (IOCs), malicious IP addresses, URLs, and attack patterns.

C. Data Aggregation and Normalization

Once the API keys are configured, the system begins collecting threat data from AlienVault OTX. The data is aggregated from multiple sources within the platform to provide acomprehensive view of the threat landscape. This collected data is then normalized, which involves converting it into a consistent format suitable for developing ModSecurity rules. Normalization ensures that the security rules generated from the threat intelligence data are both compatible and effective in protecting against emerging threats.

D. Custom Rule Set Development

Using the normalized threat intelligence data, custom Mod Security rules are created to address specific threats identified during the data collection phase. These rules are designed to detect and mitigate the threats based on the latest intelligence, enhancing the firewall's ability to block malicious traffic effectively.The rules are written in the Mod Security rule syntax andare designed to detect and block malicious traffic patterns, such as Referer header manipulation attacks and remote file inclusions. The rules extracted from AlienVault OTX threat intelligence feeds are converted into Mod Security Web Application Firewall (WAF) rules using the following algorithm:

OTX to ModSecurity Rules Generation Algorithm:

Step-1: Set API Key and OTX URL:
Define the OTX_API_KEY with the provided API key and set the OTX_URL to the AlienVault OTX API endpoint for exporting threat intelligence indicators (e.g., IPv4).
Step-2: Prepare Headers for API Request:
Create a headers dictionary that includes the API key with the key 'X-OTX-API-KEY'. **Step-3: Send API Request:**
Send an HTTP GET request to the OTX_URL using the requests.get method, passing the prepared headers.
Step-4: Check the Response Status:

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 1, 2025

Print the response status code to check whether the request was successful (200 for success).

Step-5: Handle Successful Response:

If the status code is 200, extract the JSON data from the response and print it to verify its structure.

Step-6: Verify Data Structure:

Check if the returned data is a dictionary and contains the key 'results'. If 'results' exists, assign its value to the indicators list. If the 'results' key is absent, assign the entire data to indicators.

Step-7: Open File to Write Rules:

Open the file /etc/nginx/modsec/otx_rules.conf in write mode for writing the ModSecurity rules.

Step-8: Iterate Through Indicators:

For each indicator in the indicators list:

IPv4 Indicator: If the indicator type is 'IPv4', extract the IP address and generate a ModSecurity rule to block it and write the rule to the file.

Domain Indicator: If the indicator type is 'domain', extract the domain name and generate a ModSecurity rule for domain- based blocking and write the rule to the file.

URL Indicator: If the indicator type is 'URL', extract the URL and generate a ModSecurity rule for URL-based blocking and write the rule to the file.

Step-9: Handle Failed Response:

If the status code is not 200, print an error message indicating the failure, along with the status code.

E.Dyamic Rule Update Mechanism

To enable Mod Security to respond to new threats in real-time, a dynamic rule update mechanism is implemented. This involves developing scripts or utilizing existing plugins that periodically fetch updated threat intelligence data from Alien Vault OTX. The retrieved data is then automatically converted into Mod Security rule format and applied to the WAF configuration. This automated process ensures that Mod Security continuously updates its rule sets without the need for manual intervention, providing ongoing protection against the latest threats.

F. System Integration and Traffic Filtering

The dynamic rule update mechanism is integrated into the ModSecurity configuration, allowing the system to analyze incoming HTTP traffic in real-time and apply the most up-to-date rules to detect malicious requests. This integration is thoroughly tested in a controlled environment to ensure its proper functionality and efficiency under real-world conditions, guaranteeing that ModSecurity can effectively handle live traffic.

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 1, 2025

G. Logging and Monitoring

Logging and monitoring are essential components of the implementation to ensure both the operational health and security effectiveness of the integrated system. Logging mechanisms are set up to capture all ModSecurity activities, including detected threats, blocked requests, and rule updates. This detailed logging provides an audit trail that supports forensic analysis and ensures compliance with security standards and regulations.

VI. RESULTS

Initially, a header manipulation attack was conducted to evaluate the security posture of the web application before integrating the firewall with the threat intelligence feed. In this attack, the Host header of the HTTP request was deliberately altered to spoof the domain "biketools.ru," even though the request originated from a different IP address. Figure 3 depicts that the web server accepted the forged request and allowed access to the website. This behavior highlights a significant vulnerability, as it demonstrates that the server was unable to validate the authenticity of the Host header, potentially exposing the application to various threats, such as phishing, cache poisoning, or unauthorized access.

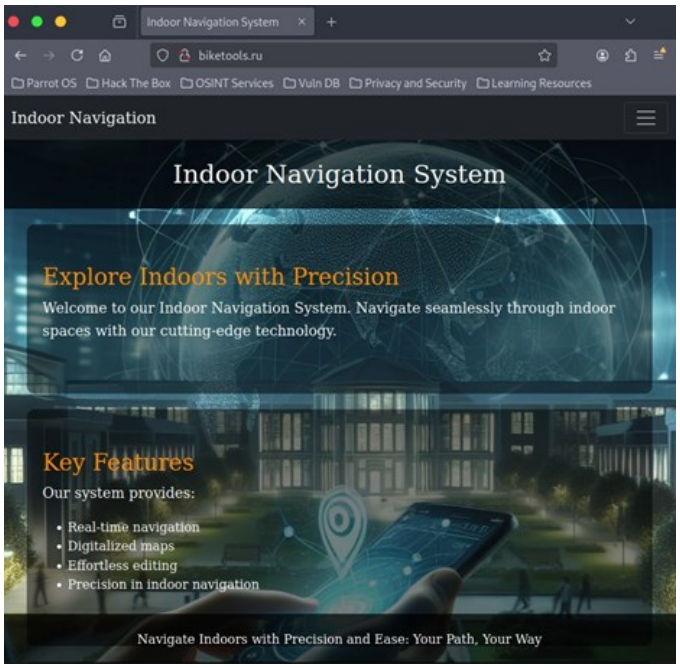


Figure 3: Website before integrating the firewall

After integrating the firewall with the threat intelligence feed (AlienVault OTX), thesame header manipulation attack was

Figure 4: Website after integrating the firewall

repeated to evaluate the effectiveness of the enhanced security configuration. As illustrated in Figure 4, the firewall successfully identified and blocked the malicious request, preventing unauthorized access to the website. The response returned a 403 Forbidden error, demonstrating that the firewall effectively mitigated the attack. This integration significantly enhances the website's security by leveraging threat intelligence to identify and neutralize potential vulnerabilities proactively.

VII. CONCLUSION

Integrating the ModSecurity Web Application Firewall (WAF) with the Emerging Threats threat intelligence feed marks a major advancement in enhancing the security framework of a web application. Through careful configuration of ModSecurity and the integration of threat intelligence rules, the defenses against a broad spectrum of cyber threats have been considerably improved. The thorough testing and deployment phases have confirmed the effectiveness of these security measures, ensuring robust protection for the application.

Continuous monitoring of WAF logs, along with extensive testing of simulated attack scenarios, has refined the WAF's ability to detect and swiftly mitigate potential threats. These efforts have strengthened the system's defenses while ensuring the reliability and resilience of the overall security infrastructure. Furthermore, the integration has facilitated proactive identification and response to

emerging threats, resulting in a robust and dynamic security posture. This comprehensive approach to security has instilled a high level of confidence in the effectiveness of the protective measures in place.

VIII. FUTURE SCOPE

Looking ahead, several strategic avenues have been identified for further advancement and expansion. One key area is the integration of advanced threat detection technologies, such as machine learning and artificial intelligence, to improve the WAF's ability to detect and respond to sophisticated cyber threats. Enhancing the integration of threat intelligence by expanding beyond the Emerging Threats feed and continuously refining rule sets is also a priority. In addition, implementing comprehensive cybersecurity training programs to educate users on secure behavior practices is crucial. Furthermore, integrating complementary security technologies, such as Intrusion Detection Systems (IDS), Endpoint Protection Platforms (EPP), and Security Information and Event Management (SIEM) systems, will strengthen layered defenses and provide comprehensive security visibility.

Collaboration with industry peers and security communities plays a vital role in exchanging insights and staying informed about emerging threats, which in turn strengthens proactive defense strategies. Regular adherence to regulatory compliance and governance standards through audits ensures that security measures remain aligned with legal frameworks. Embracing emerging technologies, fostering a strong cybersecurity culture, and leveraging industry partnerships are key to maintaining robust protection, adapting to evolving threats, and effectively safeguarding digital assets.

ACKNOWLEDGMENT

The resources required for this research study and related work were provided by the Department of CSE (Computer Science Engineering), Institute of Aeronautical Engineering, Hyderabad, India.

REFERENCES

[1] A. Tekerek, C. Gemci and O. F. Bay, [2014], "Development of a hybrid web application firewall to prevent web based attacks", IEEE 8th International Conference on Application of Information and Communication Technologies (AICT), Astana, Kazakhstan, pp. 1-4, doi:10.1109/ICAICT.2014.7035910.

[2] H. K. J and G. P. J[2023], Securing Web Application using Web Application Firewall (WAF) and Machine Learning. First International Conference on Advances in Electrical, Electronics and Computational Intelligence (ICAEECI), Tiruchengode, India, pp. 1-8.

[3] T. Rahmawati, R. W. Shiddiq, M. R. Sumpena, S. Setiawan, N. Karna and S. N. Hertiana, [2023], "Web Application Firewall Using Proxy and Security Information and Event Management (SIEM) for OWASP Cyber Attack Detection", IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS), Bali, Indonesia, 2023, pp. 280-285, doi: 10.1109/IoTaIS60147.2023.10346051.

[4] K. R. Khamdamovich and I. Aziz, 2021, "Web application firewall method for detecting network attacks," International Conference on Information Science and Communications Technologies (ICISCT), Tashkent, Uzbekistan, 2021, pp. 01-03, doi: 10.1109/ICISCT 52966.2021.9670285.

[5] R. A. Muzaki, O. C. Briliyant, M. A. Hasditama and H. Ritchi, 2020, "Improving Security of Web-Based Application Using ModSecurity and Reverse Proxy in Web Application Firewall," 2020 International Workshop on Big Data and Information Security (IWBIS), Depok, Indonesia, 2020, pp. 85-90, doi: 10.1109/IWBIS50925.2020.9255601.

[6] S. Wang, R. Liu, X. Guo and G. Wei, 2022, "Design of Web Application Firewall System through Convolutional Neural Network and Deep Learning," International Conference on Computers, Information Processing and Advanced Education (CIPAE), Ottawa, ON, Canada, 2022, pp. 454-457, doi: 10.1109/CIPAE55637.2022.00101.

[7] V. Clincy and H. Shahriar, 2018, "Web Application Firewall: Network Security Models and Configuration," IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), Tokyo, Japan, 2018, pp. 835-836, doi: 10.1109/COMPSAC.2018.00144.

[8] M. Srokosz, D. Rusinek and B. Ksiezopolski, [2018], "A New WAF-Based Architecture for Protecting Web Applications Against CSRF Attacks in Malicious Environment," 2018 Federated Conference on Computer Science and Information Systems(FedCSIS), Poznan, Poland, pp. 391-395.

[9] L. Zhang and M. Huang, 2015, "A Firewall Rules Optimized Model Based on Service Grouping", 12th Web Information System and Application Conference (WISA), Jinan, China, pp. 142-146, doi: 10.1109/WISA.2015.47.

[10] Z. Ghanbari, Y. Rahmani, H. Ghaffarian and M. H. Ahmadzadegan, 2015, "Comparative approach to web application firewalls", 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI), Tehran, Iran, pp. 808-812, doi: 10.1109/KBEI.2015.7436148.

[11] Firkhan Ali Bin Hamid Ali, 2011, "A study of technology in firewall system", IEEE Symposium on Business, Engineering and Industrial Applications (ISBEIA), Langkawi, Malaysia, pp. 232-236, doi: 10.1109/ISBEIA.2011.6088813.

[12] H. Takahashi, H. F. Ahmad and K. Mori, 2011, "Application for Autonomous Decentralized Multi Layer Cache System to Web Application Firewall," Tenth International Symposium on Autonomous Decentralized Systems, Tokyo, Japan, pp. 113-120, doi: 10.1109/ISADS.2011.20. [13] A. El-Atawy, E. Al-Shaer, T. Tran and R. Boutaba, 200.

[13] A. El-Atawy, E. Al-Shaer, T. Tran and R. Boutaba, 2009, "Adaptive Early Packet Filtering for Defending Firewalls Against DoS Attacks", IEEE INFOCOM 2009, Rio de Janeiro, Brazil, 2009, pp. 2437-2445, doi: 10.1109/INFCOM.2009.5062171.

[14] F. Zhao, X. Peng and W. Zhao, "Multi-Tier Security Feature Modeling for ServiceOriented Application Integration," 2009 Eighth IEEE/ACIS International Conference on Computer and Information Science, Shanghai, China, 2009, pp. 1178-1183, doi: 10.1109/ICIS.2009.80.

[15] A. El-Atawy, E. Al-Shaer, T. Tran and R. Boutaba, "Adaptive Early Packet Filtering for Defending Firewalls Against DoS Attacks," IEEE INFOCOM 2009, Rio de Janeiro, Brazil, 2009, pp. 2437-2445, doi: 10.1109/INFCOM.2009.5062171.

[16] G. Namit, S. Abakash, S. Dheeraj “Web Application Firewall”. CS499: B. Tech Project Final Report, 2008.

[17] W. Wang and J. Li, "An XML Firewall on Embedded Network Processor," Fourth International Conference on Networking and Services (icns 2008), Gosier, France, 2008, pp. 1-6, doi: 10.1109/ICNS.2008.15.

[18] Hamed H, Al-shaer E. Dynamic rule-ordering optimization for high-speed firewall filtering[J]. In Asiaccs 06: Acm Symposium on Information, 2006:332--342.

[19] Firkhan Ali, H. A., “ An Analysis of Possible Exploits in the Computer Network’s Security“ in ISC 2005 : Proceedings of the International Science Congress 2005. PWTC, Kuala Lumpur, 2005. pp.338.

[20] Yuan L, Chen H. FIREMAN: a toolkit for firewall modeling and analysis[C]. //IEEE Symposium on Security & Privacy. IEEE, 2006:15 pp-2.