UDK: 004.056

VEB-HUJUMLARDAN TRAFIKNI VEB-FILTRLASH ARXITEKTURASI

G‘ulomov Sh.R
PhD
Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti
Kiberxavfsizlik fakulteti dekani, dotsent
sherhisor30@gmail.com,

Annotatsiya. *Ushbu maqolada trafikni filtirlash orqali veb hujumlarning oldini olish arhitekturasi ishlab chiqilgan va tasniflangan. WFT larni ishlash mexanizimlari va uning imkoniyatlari tavsifi keltirilgan. Taklif qilinyotgan WFT arxitekturasi qanday xizmatlarni bajarishi va natija bo‘yicha qanday qarorlar qabul qilishi ko‘rsatilgan.*

Tadqiqot natijalariga ko‘ra trafik harakatini kuzatuvchi va zaifliklar bo‘yicha qanday choralar ko‘rishni hal qiladigan WFT xavfsizlik devorlarini belgilangan. Sakkizta asosiy blokdan iborat so‘rovni qayta ishlash arxitekturasi taklif etilgan va har bir bloklarning tarkibiy qismlari keltirilgan.

Ushbu maqola tadqiqotchilar va kiberxavfsizlik mutaxassislari uchun zamonaviy veb hujumlarni oldini olishni amalga oshirishda foydali bo‘lishi mumkin.

Kalit so‘zlar: *Veb hujumlar, WTF, zararli traffik, veb traffik, veb server, WAF, XSS, SQL inektsiya, dekoder, veb filter, IP filterlash.*

АРХИТЕКТУРА WEB-ФИЛЬТРАЦИИ ТРАФИКА ОТ WEB-АТАК

Гуломов Ш.Р.
PhD
sherhisor30@gmail.com,
Ташкентский университет информационных технологий имени Мухаммада ал-Хоразмий
Декан факультета кибербезопасности, доцент

Аннотация. *В этой статье разработана и классифицирована архитектура для предотвращения веб-атак путем фильтрации трафика. Представлено описание механизмов работы WFT и его возможностей.*

Предлагаемая архитектура WFT показывает какие услуги выполнены и какие решения приняты на основе результатов.

По результатам исследования определены межсетевые экраны WFT, которые отслеживают трафик и решают, какие действия предпринимать над уязвимостями. Предложена архитектура обработки запросов, состоящая из восьми основных блоков, и перечислены компоненты каждого блока.

Эта статья может быть полезна исследователям и специалистам по кибербезопасности при предотвращении современных веб-атак.

Ключевые слова: *Веб-атаки, WTF, вредоносный трафик, веб-трафик, веб-сервер, WAF, XSS, SQL-инъекция, декодер, веб-фильтр, IP-фильтрация.*

ARCHITECTURE OF WEB-FILTERING TRAFFIC FROM WEB-ATTACKS

Gulomov Sh.R.
PhD
sherhisor30@gmail.com,
Tashkent University of Information Technologies named after Muhammad al-Khorazmi
Dean of the Faculty of Cyber Security, associate professor

Annotation.

In this article, an architecture is developed and classified to prevent web attacks by filtering traffic. A description of the mechanisms of WFT operation and its capabilities is presented. The proposed WFT architecture shows what services are performed and what decisions are made based on the results.

Based on the results of the study, WFT firewalls were identified that monitor traffic and decide what actions to take on vulnerabilities. A request processing architecture is proposed, consisting of eight main blocks, and the components of each block are listed.

This article may be useful to researchers and cybersecurity professionals in preventing modern web attacks.

Keywords: *Web attacks, WTF, malicious traffic, web traffic, web server, WAF, XSS, SQL-injection, decoder, web-filter, IP-filtering.*

Introduction

One of the common challenges in various computer science disciplines is protecting computers and networks from intrusion, theft and interference. The importance of security increases as the number of Internet users increases. Web traffic filtering (WFT) acts as a barrier between a web application and a client on the Internet when it is deployed in front of the web application. WFT is a type of reverse proxy that protects the web server from being exposed to the client by detecting anomalous traffic, while the proxy server acts as an intermediary to protect the identity of the client computer [1].

WFT is governed by a set of rules known as policies and a pre-trained module to predict new incoming requests. By filtering malicious messages, these policies try to protect applications from vulnerabilities. The usefulness of WFT is partly determined by the speed and ease with which policy modifications can be deployed, allowing for faster response to different attack vectors. Figure 1 shows the structure of the WFT[2].

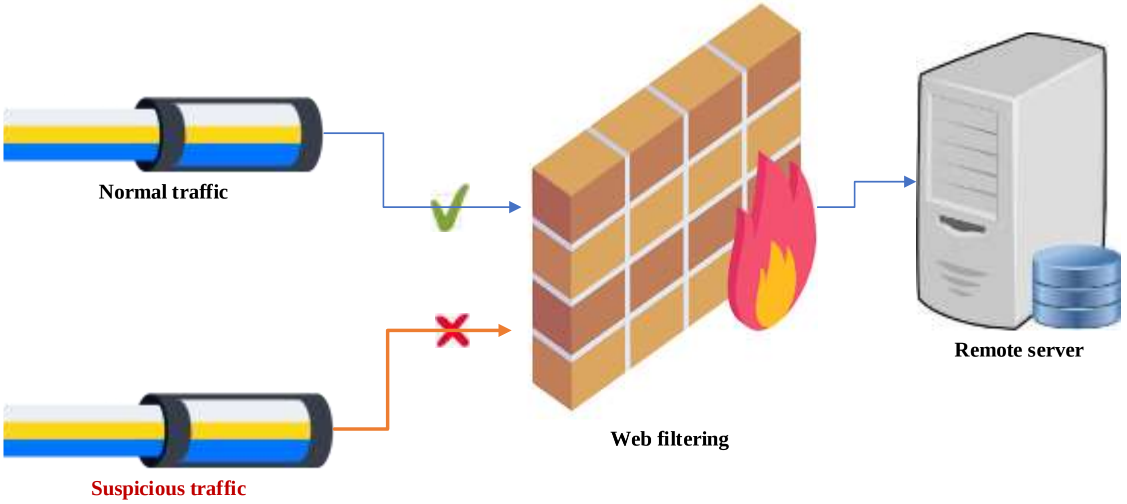

Figure 1. Structure of Web traffic filtering.

Many attempts have been made to create various security solutions such as intrusion detection systems (IDS) and firewalls. In most of these cases , network layer firewalls and IDSs do not inspect HTTP and HTTPS packets at the application layer. As a result, they cannot fully protect web servers. Web applications, especially in the cloud, are one of the most attractive targets for attackers seeking to infiltrate an organization’s information infrastructure. Internal data leaks, financial losses, and website manipulation can result from an organization’s failure to implement web security[3].

The WFT mechanism consists of two modules:

- configuration module;
- packet analysis module.

When packets are received from the Internet, the rule files filter them from the configuration module and pass the traffic to the packet sniffing module. The packet sniffer module analyzes packets and extracts characteristics from them. Using pre-trained data, it checks and identifies the nature/nature of this packet. Therefore, only parsed and allowed packets pass through the packet sniffing module to the web application server. WFT can be deployed as hardware devices in virtual appliances or as software running on the same web server as the web application or via the cloud. It works using a specific set of rules called policies. In each of these deployment models, WFT is always placed in front of the web application, intercepting all traffic between the application and the Internet. Thus, these policies define WFT firewalls that monitor traffic behavior and decide what action to take on vulnerabilities. WFT will continue to scan web applications and receive GET and POST requests to detect and filter HTTP/ HTTPS requests with malicious activity. In addition, the intelligent WFT can even ask if the participant is a human or a bot. When vulnerabilities are found in an application, WFT immediately patches them to automatically block intruders and intruders, such as bots and attacked IP addresses[4]. The most effective and efficient solutions provide the following W FT capabilities:

1. Input protection provides a comprehensive application filter that accepts only valid user input.
2. HTTPS Inspection detects HTTP/ HTTPS vulnerabilities and prevents attacks by configuring inspection rules.
3. Policies designed for widely used applications are customized according to specific requirements and needs. Thus, it protects applications from vulnerabilities and also provides real-time traffic information.
4. Data Leak Prevention provides alerts and prevents any unusual traffic or data leakage by identifying, filtering and protecting personal data.
5. Automatic Attack Blocking provides automation to block attacks by preventing malicious traffic from entering the network[5].

Web application security is essential to protect information, customers, and organizations from information theft, trade interference, or other destructive activities associated with cybercrime[6]. Approaches to web application security and protection seek to ensure application security through measures such as WAF, multi-factor validation for clients, leveraged security, and threat approval to preserve client state. Every website on the Internet is vulnerable to cyber attacks[7]. The dangers range from human error to sophisticated cyberattacks carried out by an organized group of criminals[8].

The main types of known web attacks are shown in Figure 2.

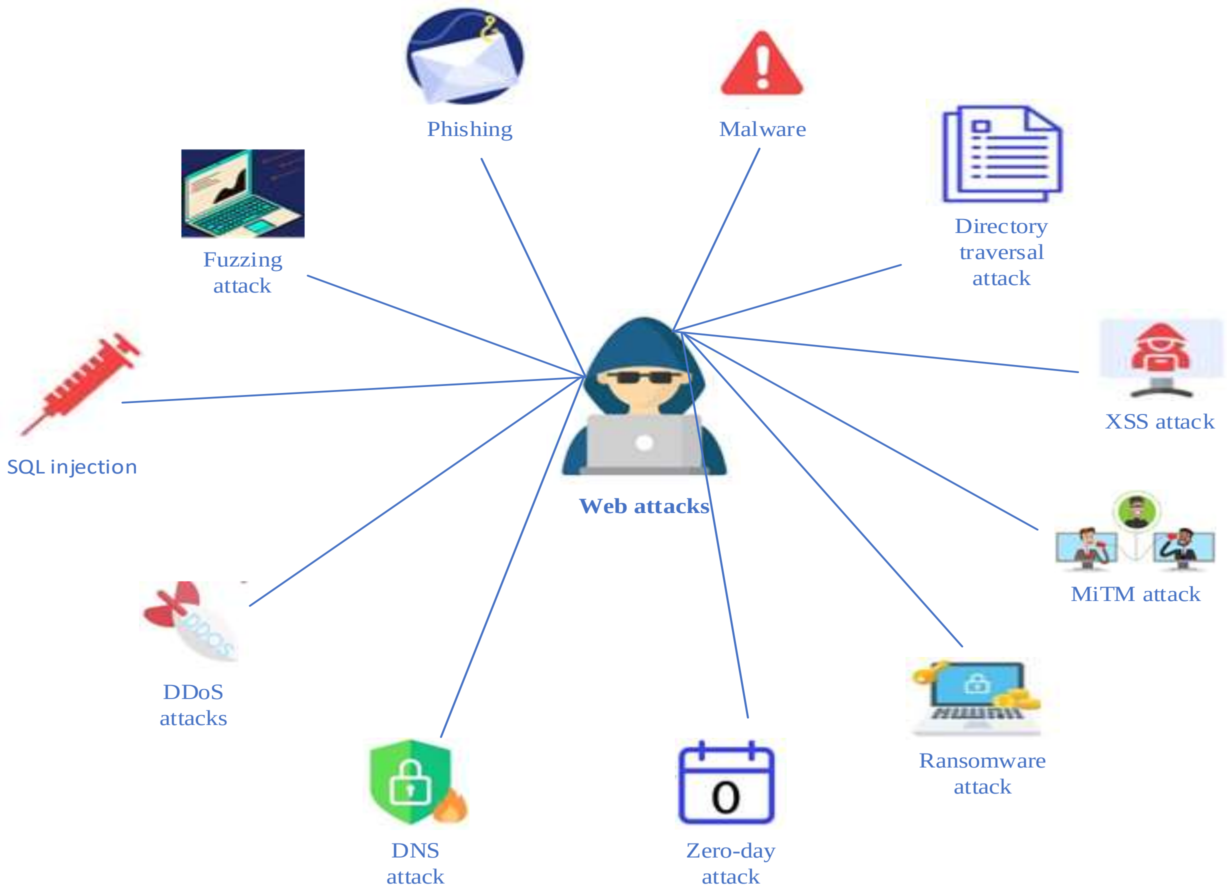

Figure 2. Main Types of Known Web Attacks

XSS and SQL injections are among the most popular web attacks.

Web traffic filtering architecture to protect against web attacks

The WFT architecture we propose runs as an operating system service that acts as an intermediary between the web server and clients. This service receives the request, parses it, extracts features, classifies them, and makes decisions based on the result of the classification [9].

Request Processing Architecture

The proposed request processing architecture consists of the following eight main blocks:

1. Study block;
2. Parsing block;
3. Threat Detection Block;
4. Audit block;
5. configuration block;
6. Interactive block;
7. Unit of classification;
8. Decision Block.

Study block

After running WFT, the dataset name and classification algorithm will be retrieved from the databases for model training, WFT is now ready to accept queries. Popular classification algorithms are used here, any algorithm can be added by inserting its name into the database [10].

Parser block

With the WFT enabled and the training model enabled, the WAF is now ready to accept requests. When an HTTP/ HTTPS request arrives at WFT, the parser breaks the request into feature extractions. The parser creates the final feature vector and passes this vector to the classifier [11].

Threat detection block

The threat detection block consists of three modules. Web requests are sent to the data filtering module. Figure 3 shows a diagram of the threat detection block.

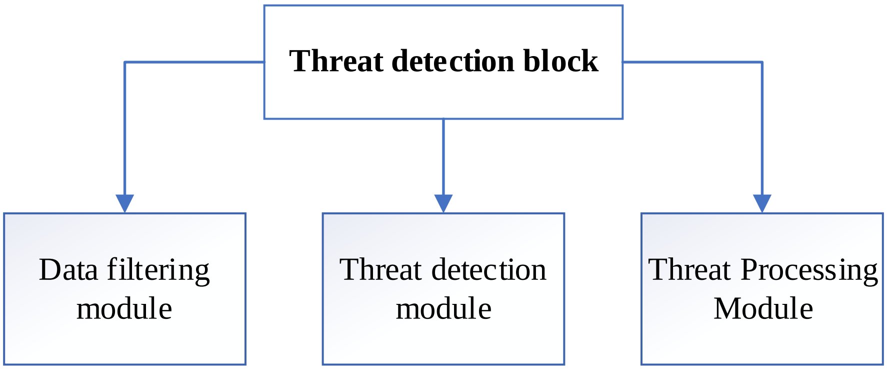

Figure 3. Diagram of the threat detection block.

The data filtering module filters web requests with IP filter, SQL injection filter, XSS filter, ID filter, File upload filter, Dictionary attack filter. Figure 4 shows the components of the data filtering module [12].

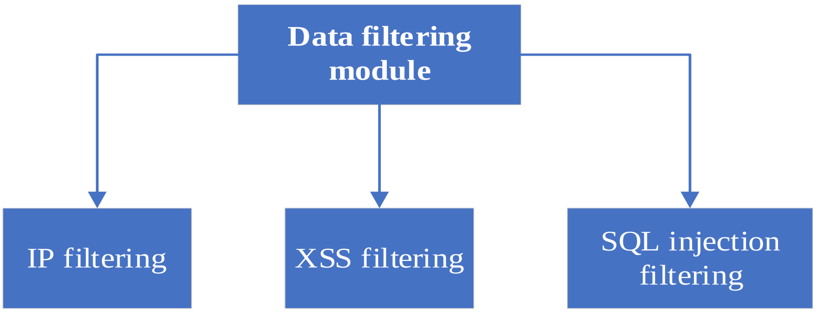

Figure 4. Components of data filtering modules.

The threat detection module consists of two elements:

Decoder: The captured data is in raw form and must be decoded to a standard format. The decoded data set consists of various features and parameters, so for training the module, it is necessary to select the appropriate parameters/features. The DDoS detection features were selected based on the analysis of the standard data set and the correlation analysis in the captured data. For SQL injection and XSS detection, a standard set of data is analyzed to compare normal and attacked traffic, and appropriate parameters are selected [13].

Numbering: GET and POST request methods are encoded as 1 and 2 respectively. Similarly, flag values in text forms will be converted to 1 and 0 respectively. Figure 5 shows the elements of the threat detection module.

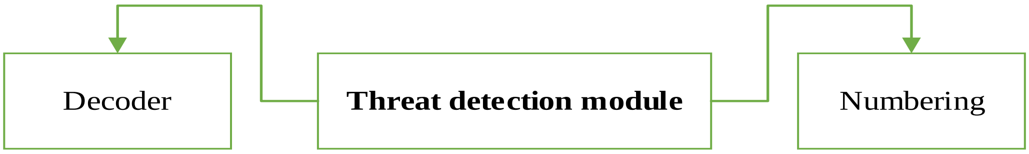

Figure 5. Elements of the Threat Detection Module.

The threat processing module evaluates the threat level of the detected threat, compares the score value and the set threshold according to the result, to block the request, release request, or registration request. In Figure 6 represents the threat level of the detected threat. The Threat Engine can be configured to set a threat threshold for registering and denying a web request.

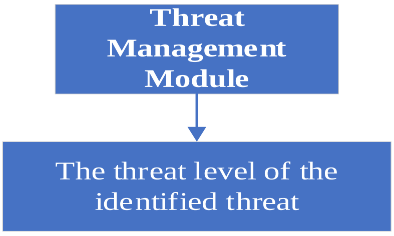

Figure 6. Threat level of detected threat.

Audit block

The audit block not only provides a framework for auditing security events, but can also help administrators analyze system security risks. The audit system logs contain detailed information about web requests, such as: request URL, port number, request IP address, time, date, and description of the exception. The audit subsystem contains the database log and mail notification. The database log records all information about a web request whose threat level exceeds the log threshold. It can clearly reflect the security status of web application access through charts and data. The email notification system provides instant notification when a web application is under attack. Figure 7 shows the security audit block.

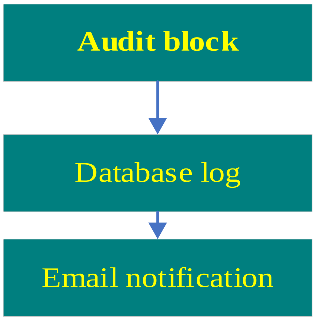

Figure 7. Security Audit Block.

Configuration block

The administrator can configure the firewall in the configuration subsystem. Regular users can only view configuration information. The filter module can be configured to set the filtering order of the filter, set the blacklist of IP addresses, and enable and disable various filters. The database log can be configured to enable or disable the log function and the email notification function. The self-management module can reset the firewall. If the configuration information is changed, the firewall must be reinitialized to read the new configuration information. Figure 8 shows the configuration block.

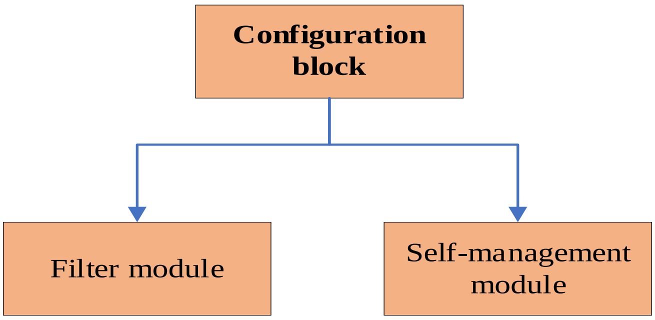

Figure 8. Configuration block.

interactive block

The interactive block is mainly used to interact with the firewall. The block includes two functions: log viewing and account management. Once authenticated, the administrator can view firewall logs, reconfigure the firewall, change administrator account information, add users, and change user information and browsing permissions. Regular users can view the firewall log and change account information. In addition to functional integrity and ease of use, the interactive block uses a large number of charts and statistical tables that present the firewall log to the user as a web page to improve the user experience. Figure 9 shows the functions of the interactive block.

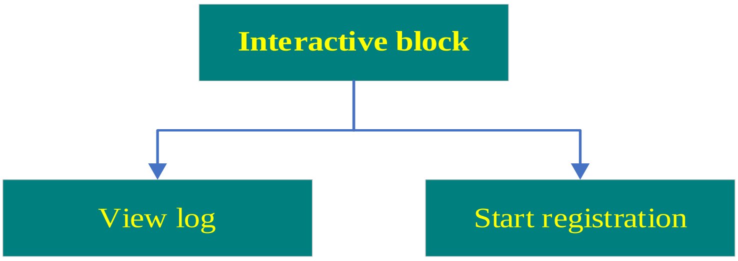

Figure 9. Interactive Block Functions.

Classification unit block

The classification unit block receives the final vector from the parser and classifies the request depending on it. The classification block sends the classification results to the decision block.

Decision group block

This block receives the classification results from the classification module and redirects the request to the web server if the request is normal and discards the request if it is abnormal.

And so, in Figure 10, the architecture of web-filtering traffic from web - attacks is proposed.

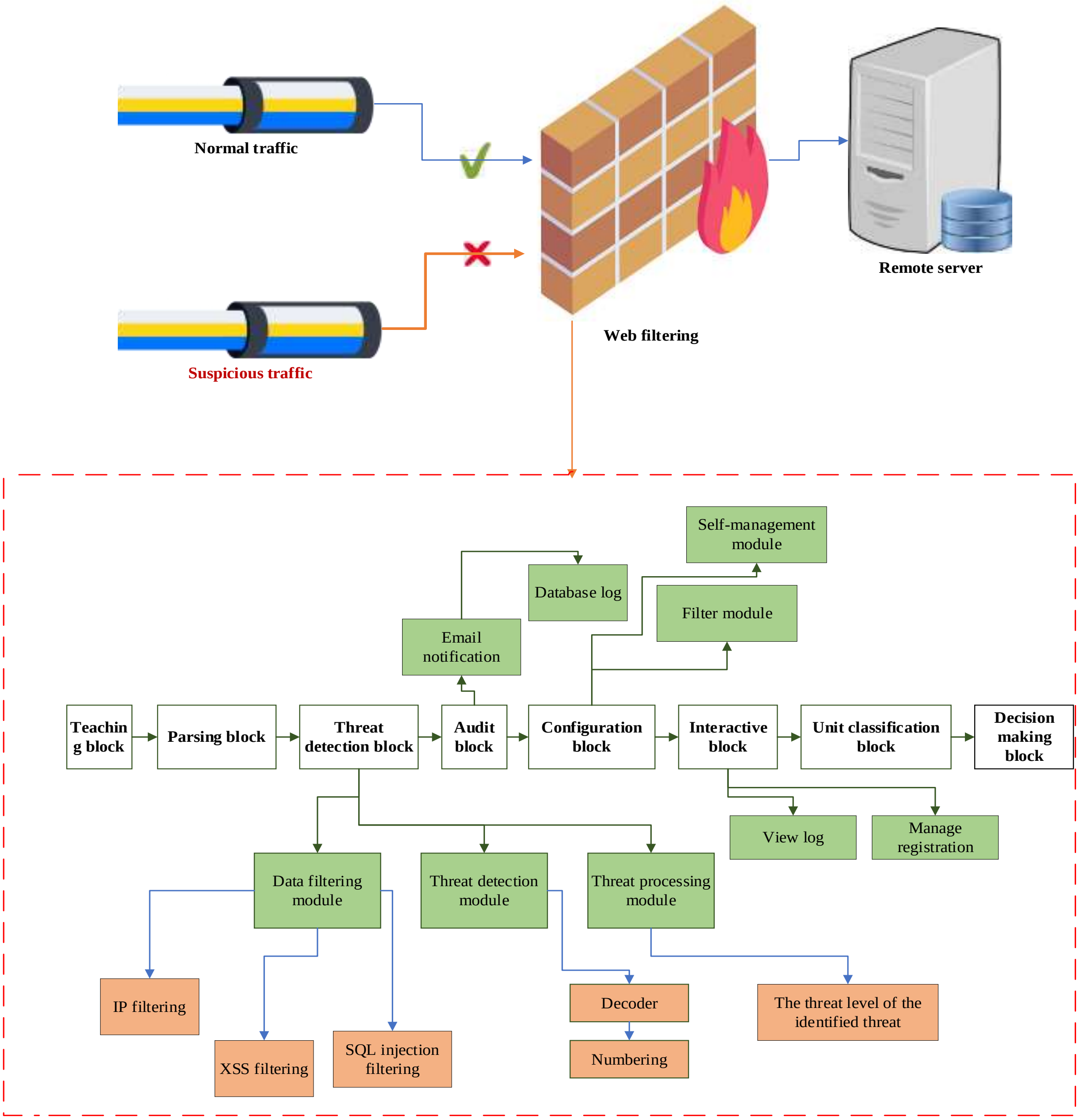

Figure 10. Architecture of web-filtering traffic from web-attacks

The proposed traffic web filtering architecture makes it possible to protect web resources from malicious bots and suspicious traffic, and allows minimizing and preventing existing and new web attacks based on multifactorial traffic analysis.

Conclusion

Every website on the Internet is always vulnerable to cyber attacks to some degree. Threats range from human error to sophisticated cyber attacks by organized crime groups. Web Traffic Filtering (WFT) acts as a barrier between the web application and the client on the Internet when the web application is deployed. WFT is driven by a set of rules known as a pre-trained module and policies to predict new incoming requests. By filtering out malicious messages, these policies try to protect applications from vulnerabilities.

In the article, an architecture for detecting and eliminating attacks directed at web applications was developed, and its components, organization, and advantages were presented. This study can be useful for organizations' cyber security response professionals and network administrators in configuring the security of information communication systems.

REFERENCES

1. A. Osincev and O. R. Laponina, "Vulnerability testing in web applications external entities XML," International Journal of Open Information Technologies, vol.7, no.10, pp.71–79, 2019.
2. P.P.MukkamalaandS .Rajendran,"A survey on the different firewall technologies," International Journal of Engineering Applied Sciences and Technology, vol. 5, no. 1, pp. 363–365, 2020.
3. W. Wang and K. Siau, "Artificial intelligence, machine learning, automation, robotics, future of work and future of humanity," Journal of Database Management, vol. 30, pp. 61–79, 2019.
4. J.Doshiand T. Bhushan, " Sensitive data exposure prevention using dynamic database security policy," International Journal of Computer Application, vol. 106, no. 15, pp. 18600–19869, 2014.
5. M.-H. Huang and R. T. Rust, "Artificial intelligence in service," Journal of Service Research, vol. 21, no. 2, pp.155–172, 2018.
6. J.H.Li,"Cyber security meets artificial intelligence: asurvey," Frontiers of Information Technology & Electronic Engineering, vol. 19, no. 12, pp. 1462–1474, 2018.
7. P.P.MukkamalaandS. Rajendran, "A survey on the different firewall technologies," International Journal of Engineering Applied Sciences and Technology, vol. 5, no. 1, pp. 363–365, 2020

8. Akbar Memen, Ridha Muhammad Arif Fadhly, et al., SQL injection and cross site scripting prevention using OWASP ModSecurity WebApplication firewall, Int. J. Inf. Visualization, 2018, vol. 2, no. 4. pp. 286–292.

9. Yuan, H. et al., Research and implementation of WEB application firewall based on feature matching, Proc.Int. Conf. on Application of Intelligent Systems in Multi-modal Information Analytics, Springer, 2019, pp. 1223–1231.

10. Domingues Junior, M. and Ebecken, N.F.F. (2021) 'A new WAF architecture with machine learning for Resource-efficient use', Computers &amp;amp; Security, 106, p. 102290. doi:10.1016/j.cose.2021.102290.

11. D. Wichers and J. Williams, "Owasp Top Ten," 9e open web application security project, vol. 3, 2017

12. K. Dalai and S. Kumar Jena, "Neutralizing SQL Injection Attack Using Server Side Code Modification in Web applications," Security and Communication Networks, vol. 2017, Article ID 3825373, 2017.

13. D. Mitropoulos, V. Karakoidas, P. Louridas, and D. Spinellis, "Countering Code Injection Attacks: A Unified Approach," Information Management& Computer Security, vol.19, no.3, 2011.