International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 04 | Apr 2019 www.irjet.net p-ISSN: 2395-0072

Web Application Firewall: Artificial Intelligence Arc

Parikshit Prabhudesai[1], Aniket A. Bhalerao[2], Rahul Prabhudesai[3]

[1]Director, Pitambari Products Pvt. Ltd., Maharashtra, India
[2]Deputy General Manager, IT & System Department, Pitambari Products Pvt. Ltd., Maharashtra, India
[3]Assistant General Manager, IT & System Department, Pitambari Products Pvt. Ltd., Maharashtra, India

---

Abstract – *Nowadays every business and individual, are using online platform to do business and to promote themselves by performing financial transactions as well as handling user confidential information transactions. Every website holder needs security against all known and unknown threats; hence, we are developing a web application firewall using artificial intelligence architecture to recognize attacks and existing vulnerabilities by experiencing the behavior of attacker and user in a unique way.*

1. INTRODUCTION

Web application firewall is necessary for all static and dynamic website holders to maintain & enhance security of information, which is available on website or on server. We all know that attackers are finding various vulnerabilities daily. We need to update our security system by giving it self-intelligence by changing our approach towards protection by applying self-created knowledgebase.

1.1 A. I. Architecture Engine

Artificial intelligence architecture engine has a base and builds on the integrity parameters defined by OWASP[2] & ITProPortol[3]. It has automated prevention and mitigation system which is able to recognize attack pattern behavior and impact on the information system to identify attack pattern by building own knowledgebase and mitigation category. Its algorithm has the capability to skip false positive attack pattern by building own testing environment lab for all new packets to the software. The software identifies whether it's a false positive or a positive impact.

Artificial Intelligence architecture engine has OWASP[2] integration module which will help to build predefined vulnerability database as well as help to build a knowledgebase for particular attack type.

1.2 OWASP[2] Integration

OWASP[2] (open web application security projects) is a web portal, which keeps track on all vulnerabilities from the globe and categorises it by its severity and impact on information system, hence it is the leading open web vulnerability database. OWASP has developed an API which is able to provide data access for third party queries and to get predefined database. In order to stay relevant with the time, we have integrated OWASP API, so that we can provide cutting edge security.

Table -1: OWASP TOP VULNERABILITIES CHART

Vulnerability	Severity	Type
DDoS	High	Web Threat
Spamming	Medium	Mail Threat
SQL Injection	High	Database Threat
Proxy	High	Identity Threat

2. Detection Method

2.1 Modules:

Detection method contains two uniquely designed modules for threat detection and mitigation. In first module, when a WAN packet approaches the DNS, it is automatically diverted to the WAF. WAF then separates its segments depending upon meta-data. The AI engine will check the source code and threat segments depending upon the defined database and behavioral based AI engine knowledgebase. In second module, mitigation is applied by using OWASP engine or by using own created mitigation algorithm to treat packets properly by removing false positive.

2.2 Algorithm:

In first phase, all packets are filtered through the main web application firewall engine, which is integrated with OWASP for filtering predefined vulnerabilities as well as identifying and eliminating globally defined threats.

If OWASP definition matches to the input packets, then the packet will be dropped immediately by WAF. If input packet definition does not match with OWASP definition, then first phase will mark those packets partially cleaned. Here, the first phase will end and the packet will be transferred to the second phase.

In second phase, AI engine will receive those partially cleaned packets as an input. First activity from AI engine will be to record packets' behavioral pattern and if behavioral pattern matches to the existing knowledgebase then AI engine will mark those packets as malicious and in another case, if packets' behavioral pattern does not match with existing knowledgebase then AI engine will inspect the packets' behavior by giving it virtual environment to detect whether it

is harmful for website or not. If packets behavioral pattern is found harmful then AI engine will record packets behavioral pattern and insert it to knowledgebase and if the packets behavioral pattern is not found harmful to the virtual web application then the AI engine will mark it to cleaned and pass those cleaned packets to third phase, here second phase ends.

In third phase WAF targets IP section and to apply global filter on the IP's, IP sanitization section includes IP ban system which is regularly updated by taking updates from Virustotal[6] API. We have integrated IP sanitization section with virustotal API which fortifies the database by giving the information about the globally banned IPs to the system.

Finally, cleaned packet with cleaned IP will go to the main web application and get response from web application but before getting a response, each transaction of packets will be recorded in a log retention system.

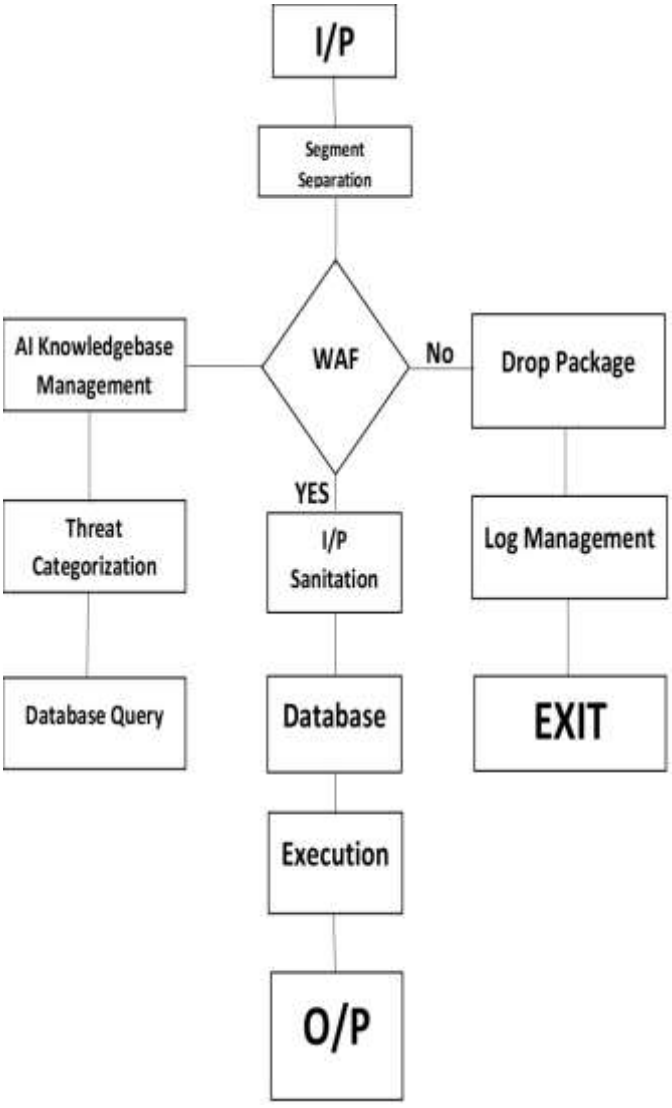

Chart -1: Flow Chart

2.3 Affected Area:

Fig -2: WAF Working

Basic working of the web application firewall is to prevent malicious packets from reaching to the main web application. It is not concerned about the vulnerabilities of the source code and hosting vulnerabilities as this WAF has its own independent detection system. The firewall will not be affected even in case of multiple external vulnerabilities.

3. ADVANTAGES

1. Intrusion prevention system and intrusion detection system will get its own artificial intelligence as a backup layer, which will provide an advance layer to the threat protection system by giving transaction wise experience to the system, which will handle threats more carefully than ever before to eliminate false positive results.
2. While analyzing and preventing threat or malicious packets, normal firewall needs to scan each packet separately and it takes more time comparatively. By providing artificial intelligence to web application firewall, which makes segment scanning on each packet, it takes very less time to handle threat as well as very low bandwidth consumption.
3. Artificial intelligence will reduce manpower and human interaction as well as human error by giving experience to the system to handle each threat and to reduce false positive response.
4. Threat log retention helps by maintaining knowledgebase and taking actions actively by learning from the knowledgebase.

4. APPLICATION

By doing existing market survey, we found that among all CMS's major parts are using PHP language as a web application platform. So, based on this information we implemented above algorithm into user friendly application by using PHP language, which is open source. According to this study, while developing user end application, we created a process flow in three phases. In first phase, we create neural network at the application layer for routing packets through threat detection engine by making independent

proxy server for a particular layer after which segmentation is applied on the packets to split into layers.

In second phase, we send those segmented packets to artificial intelligence arc engine to detect behavioral pattern from inbuilt knowledgebase and to save these behavioral events in event log section and to follow algorithm so on.
In third phase, we integrate global threat detection system with the main firewall engine to detect and drop globally declared threat definitions; after which it passes cleared and clean packets to IP sanitization module to detect whether the IP is banned. If the IP is not banned it reaches to the web application.

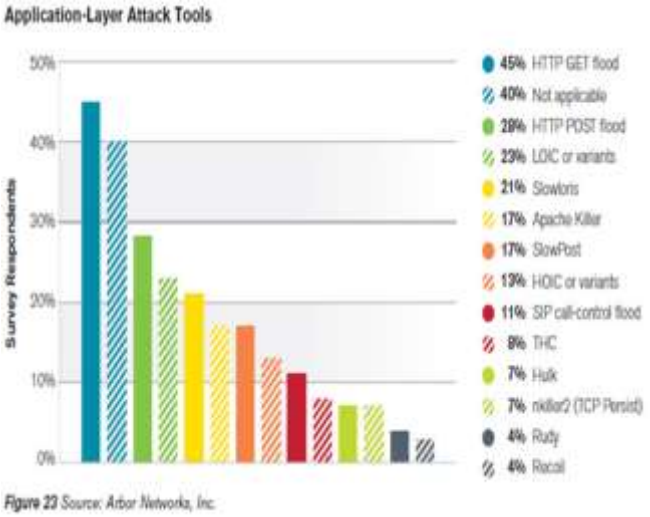

Fig -3: Application layer attack tool analysis

Above figure shows the statistics of attacking methods which has an impact on the application layer by using various destructive hacking tools. Therefore, our main target is to protect the application layer.

5. CONCLUSION

Hence, we conclude that among all existing web application firewalls, knowledgebase system with artificial intelligence is not implemented yet. It is imperative that the, updated security system must be armed with AI to recognize attack pattern and behavior by creating own knowledgebase and mitigation system by eliminating false positive results separately and by treating each packet independently. We have developed a neural network based AI engine for web application firewall which is able to mitigate all the loopholes by using artificial intelligence.

REFERENCES

[1] Web Application Firewall Market Worth $5.48 Billion by 2022. CISO Magazine. 5 October 2017. Retrieved 10 April 2018.

[2] "Web Parameter Tampering - OWASP". www.owasp.org.

[3] Svartman, Daniel (12 March 2018). "The OWASP Top Ten and Today's Threat Landscape". ITProPortol. Retrieved 10 April 2018.

[4] K. Elissa, “Title of paper if known,” unpublished. Jason Pubal (March 13, 2015). "Web Application Firewalls - Enterprise Techniques" (PDF). SANS Institute. SANS Institute InfoSec Reading Room.

[5] "TEST METHODOLOGY Web Application Firewall 6.2". NSS Labs. NSS Labs. Retrieved 2018-05-03.

[6] Lardinois, Frederic. "Google Acquires Online Virus, Malware and URL Scanner VirusTotal". TechCrunch. Retrieved 12 April 2013.

[7] Anderson, James P., "Computer Security Threat Monitoring and Surveillance," Washing, PA, James P. Anderson Co., 1980.

[8] David M. Chess; Steve R. White (2000). "An Undetectable Computer Virus". Proceedings of Virus Bulletin Conference. CiteSeerX 10.1.1.25.1508.

[9] Denning, Dorothy E., "An Intrusion Detection Model," Proceedings of the Seventh IEEE Symposium on Security and Privacy, May 1986, pages 119–131

[10] Lunt, Teresa F., "IDES: An Intelligent System for Detecting Intruders," Proceedings of the Symposium on Computer Security; Threats, and Countermeasures; Rome, Italy, November 22–23, 1990, pages 110–121.

[11] "Comparison operators". PHP.net.

[12] Pawel Krawczyk (2013). "Most common attacks on web applications". IPSec.pl. Retrieved 2015-04-15.

[13] Pawel Krawczyk (2013). "So what are the "most critical" application flaws? On new OWASP Top 10". IPSec.pl. Retrieved 2015-04-15.

ABBREVIATIONS

[1] OWASP – Open Web Application Security Projects
[2] WAF – Web Application Firewall
[3] WAN – Wide Area Network (i.e. Internet)
[4] DNS – Domain Name Server
[5] AI – Artificial Intelligence
[6] IP – Internet Protocol
[7] API – Application Program Interface
[8] CMS – Content Management System
[9] PHP – Personal Home Page