www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)

Vol. 03, Issue 01, January 2023, pp : 145-147

e-ISSN : 2583-1062

Impact Factor : 2.625

MODEL BASED SOLUTION FOR PREVENTING MODERN WEB-BASED CYBERATTACKS USING EDGE COMPUTING, ARTIFICIAL INTELLIGENCE AND CLOUD COMPUTING

Avinash Sawade[1], Mohammad Umar[2], Rohit Kawle[3], Rahul Gurjar[4], Dr.B.J.Mohite[5]

[1,2,3,4,5]Zeal Institute of Business Administration, Computer Application and Research, Pune, Maharashtra, India.

ABSTRACT

Businesses are increasingly adopting web-based applications, which is leading to a rise in cyberattacks this leads to weaknesses in these applications. As threat actors continue to develop more creative attacks, it is harder for cyber defence technologies to keep pace. Attackers are always ready with creative attacks with the help of encoding schemes, obfuscated payloads, new 0-day exploits, etc. To prevent such web-based attacks the major defence mechanism implemented is the Web Application Firewall(WAF) but still, it gets bypassed with various tactics of payload and exploit development. To reduce the gap between attack strategies and defensive mechanisms there is a need of model based solution which will be in sync with the creative art of bypassing mechanisms adopted by threat actors. Along with the need for such a learning model, there is also a need for data pre-processing, analysis, and prevention of malicious inputs as near as possible to the client side. This model will be an extension to the existing Web Application firewall.

Keywords: Edge Computing, Edge Intelligence, Web Application Firewall, Threat.

1. INTRODUCTION

The model must be based on advantage of cloud-based technologies, Edge computing, Artificial Intelligence, Machine learning, and Edge intelligence. The adoption of Edge computing and Edge Intelligence will help to avoid conflict between legit request and malicious request and will reduce the unnecessary data passing through and will work on the requests only which seems malicious. The request which seems malicious but not having rules and fingerprints to get detected by the firewall will be sent to the learning model situated in the cloud to realize the intent of the request and if deemed malicious will create a new rule to drop the request otherwise the request will be flagged as safe and allowed to pass through. The newly created rule will help to drop such malicious requests if received in the future.

Statement of Problem:

The Threats for web applications are on rise day by day. New tactics, techniques, bypasses are being used by attackers to evade the implemented protection. The firewall can protect the application server against the existing attacks it is familiar with but the firewall can't detect new bypass techniques and eventually threat actor can evade the protection implemented.

2. OBJECTIVE OF STUDY

The solutions which are generally implemented traditionally are good for traditional client server architecture and somewhat helpful for servers with multiple hosts. The researchers suggested a model which will extend the use of traditional firewall by integrating edge computing and artificial intelligence technologies for protect the underlying architecture as well as to make firewall learn about new bypasses and protect against them.

3. PROPOSED MODEL

To reduce the gap between attack strategies and defensive mechanisms, there is a need for a model-based solution that can adapt to the ever-changing tactics of threat actors. This model should be able to learn from incoming requests in real-time and update its security rules accordingly. In addition, this model should also incorporate data pre-processing and analysis, as well as prevention of malicious inputs as close to the client side as possible.

The proposed model will be an extension to the existing Web Application Firewall. By incorporating cloud-based technologies, Edge computing, Artificial Intelligence, Machine learning, and Edge intelligence, the model will be able to detect malicious requests in real-time. The adoption of Edge computing and Edge Intelligence will help to avoid conflicts between legitimate requests and malicious requests and will reduce the unnecessary data passing through. The model will work on requests only which seem malicious. The requests which seem malicious but do not have rules and fingerprints to get detected by the firewall will be sent to the learning model situated in the cloud to realize the intent of the request. If deemed malicious, the model will create a new rule to drop the request; otherwise, the

request will be flagged as safe and allowed to pass through. The newly created rule will help to drop such malicious requests if received in the future.

Diagram of Proposed System:

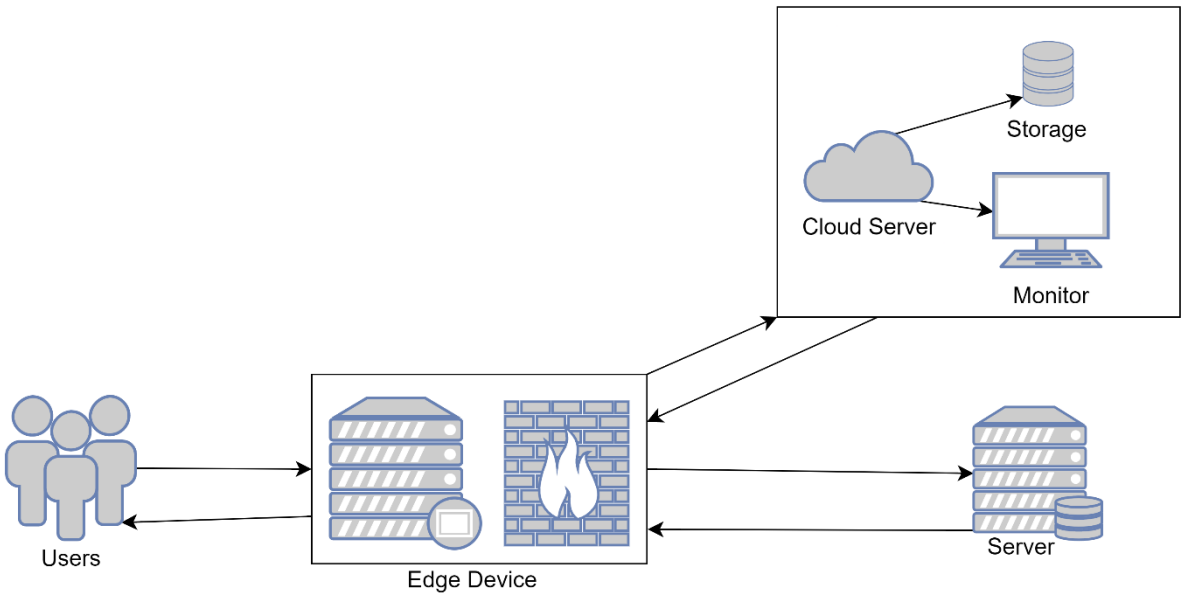

Diagram: Architecture of Edge Computing Security implementation

4. COMPONENTS OF SYSTEM

Network Level Components:

At the network level, the proposed model would likely include the following components:

1. Edge devices: These would be responsible for collecting and processing data from client-side requests, and for implementing the first line of defence against malicious inputs.

   1.1 Client-side monitoring: This component would be responsible for monitoring client requests and analysing them for potential malicious inputs.

   1.2 Firewall: This component would be responsible for implementing the first line of defence against malicious inputs by filtering out any requests that match known attack patterns.

   1.3 Data collection: This component would be responsible for collecting data from client requests for analysis and rule generation.

2. Cloud infrastructure: This would provide the necessary resources for data analysis, machine learning, and rule generation.

   2.1 Data analysis: This component would be responsible for analysing data from edge devices and identifying patterns that indicate malicious inputs.

   2.2 Machine learning: This component would be responsible for training models based on the data analysis results, and for generating new rules to detect and block malicious inputs.

   2.3 Cloud-based management: This component would be responsible for managing and updating the edge devices and machine learning models.

Software Level Components:

At the software level, the proposed model would likely include the following components:

1. Edge intelligence: This would be responsible for analysing and classifying client-side requests, and for implementing the first line of defence against malicious inputs.

   1.1 Client-side monitoring: This component would be responsible for monitoring client requests and analysing them for potential malicious inputs.

   1.2 Firewall: This component would be responsible for implementing the first line of defence against malicious inputs by filtering out any requests that match known attack patterns.

1.3 Data collection: This component would be responsible for collecting data from client requests for analysis and rule generation.

2. Machine learning: This would be responsible for analysing data from edge devices, identifying patterns, and generating new rules to detect and block malicious inputs.

   2.1 Data analysis: This component would be responsible for analysing data from edge devices and identifying patterns that indicate malicious inputs.

   2.2 Model training: This component would be responsible for training models based on the data analysis results, and for generating new rules to detect and block malicious inputs.

3. Cloud-based management: This would provide the necessary resources for managing and updating the edge devices and machine learning models.

   3.1 Remote management: This component would be responsible for remotely managing and updating the edge devices from the cloud.

   3.2 Rule management: This component would be responsible for managing and updating the security rules generated by the machine learning component.

4. Data analysis: This would be responsible for analysing data from edge devices and machine learning models to identify potential threats and improve the overall effectiveness of the model.
5. Security rules: This would be responsible for implementing the rules generated by the machine learning component to detect and block malicious inputs
6. Client: This component would be responsible for sending requests to the server and receiving responses.
7. Server: This component would be responsible for processing the requests received from the client, and for providing the appropriate responses.

8. LIMITATIONS OF PROPOSED SYSTEM

1. The system will not be able to provide protection against the business logic vulnerabilities.
2. The response time will be reduced due to interference of edge device and extra request and responses from cloud device.

9. CONCLUSION

As cyber threats continue to evolve and become more sophisticated, it becomes increasingly difficult for current defence technologies to keep up. To bridge this gap between attack strategies and defensive mechanisms, there is a need for a model-based solution that can adapt to the ever-changing tactics of threat actors. The proposed model, based on cloud-based technologies, Edge computing, Artificial Intelligence, Machine learning, and Edge intelligence, will be able to detect malicious requests in real-time and update its security rules accordingly. This model-based solution will reduce the gap between attack strategies and defensive mechanisms by providing a real-time, adaptive approach to web-based attacks. It will also improve the effectiveness of existing Web Application Firewalls by incorporating data pre-processing, analysis, and prevention of malicious inputs as close to the client side as possible.

10. REFERENCES

[1] Dr Mohite B.J titled “Study on different data security measures preferred in Warana Udyog Samuh”, Symphony, ZIMCA’s Management Journal [2017].

[2] Yu, Wei and Liang, Fan and He, Xiaofei and Hatcher, William Grant and Lu, Chao and Lin, Jie and Yang, Xinyu titled “A survey on the edge computing for the Internet of Things”, IEEE access, (6):6900–6919, [2017], 10.1109/ACCESS.2017.2778504.

[3] James Kettle titled “Server-Side Template Injection”, PortSwigger Research [2015].

[4] Gareth Heyes titled “XSS in hidden input fields”, PortSwigger Research [2015].

[5] Dr. K K Paliwal, Arun Kumar Rana, Sandeep Arora, Rohit Sharma titled “A Review on Web Application Security: A research plan”, IJCRT ISSN: 2320-2882 [2018].