Centralized Web Application Firewall Security System

Saher Manaseer[1] & Ahmad K. Al Hwaitat[1]

[1] Department of Computer Science, King Abdullah the II IT School, The University of Jordan, Amman, Jordan.

Correspondence: Department of Computer Science, King Abdullah the II IT School, The University of Jordan, Amman, Jordan. E-mail: saher@ju.edu.jo

Received: May 31, 2018 Accepted: June 6, 2018 Online Published: September 29, 2018

doi:10.5539/mas.v12n10p164 URL: https://doi.org/10.5539/mas.v12n10p164

Abstract

In this paper we propose a centralized web firewall system for web application security which will provide a new type of synchronized system, which has the ability to detect and prevent a variety of web application attacks for a wide range of hosts at the same time , using an centralized command and control system, the attacked client then sends the information to a centralized command and control server which will distribute the attack information to all of the integrated clients connected to it. The distributed information contains all of the attack information including the type of attack, the IP address of the attacker, and the time of attack. The process of receiving the attacker's information and distributing it through the centralized web firewall is done automatically and immediately at the time of the attack. And all of the receiving clients will take actions against the threat depending on the distributed information such as banning the IP address of the attacker. The main process aims to protect multiple clients from any possible attack from the same attacker or the same type of attack. The system has been implemented to protect a real web application. Experiments showed that the attacks has been successfully prevented on multiple hosts at the time.

This paper came to provide a centralized web firewall system that connect different web firewalls in order to detect and prevent different types of web attacks and work as a fully integrated system with the different clients.

Keywords: Distributing web-firewall, SQL Injection, XSS, DDoS Attack, Suspicious User Behavior, Web Applications

1. Introduction

Recently, the revolutionary growth of web application usage started to increase in a large way which made the possibility of more web application weaknesses and vulnerabilities to appear and more infiltration attempts to happen on daily basis .the development of web applications witnessed a huge change along with the event of the Internet appearance . Most companies and individuals have started to use web applications on daily basis (Rababah O. et al. 2016). The web became the main link that connects all users all over the world where private information about the web users is stored in databases. Some of these activities contains sensitive data about the users such as e-banking. Social security, passwords, and money authorization transactions information (Peotta L. et al., 2011). The security of the user's information is a major concern for all e-business owners and administrators because of the existence of successful infiltration attempts against web applications across the history. Many attackers may be able to compromise some web applications and get access to private data across the globe by exploiting several web application vulnerabilities (Chavan S., & Meshram B., 2015). Such cyber threats can cause financial loss for many parties including private companies and any other type of infrastructure.

Different types of web security mechanisms have been developed to detect and prevent multiple types of web threats.

The idea of detecting web threats and passing down the information's to other web hosts plays a major role in preventing the occurrence of possible attacks performed by the same attacker or using the same technique of attack (Chou T., 2013).

There is a demand to develop a better and fast methods and systems that work as a fully integrated system with different web firewalls to detect, warn, prevent and take actions against the attacker.

2. Related Works

(Shadi Aljawarneh et al., 2013) have focused on one issue, namely the integrity of web content. It has been shown that given the limitations of SSL, a loss of web content integrity is possible because of the statelessness of HTTP. In an attempt to overcome this problem, we have formulated a systematic web security framework that could provide continued reliable and correct services to external users, even though a web data manipulation problem may have occurred. It was suggested that such a framework will offer an increased level of user confidence, since the framework provide a greater protection against web server subversion.

An approach for (Raikar D., 2012), it is based on identification of hotspot from the application and data sources which are trusted and highlighting data that come from these sources as trusted, with the notice that only trusted data can form the parts of queries which are semantically relevant such as SQL keywords and generators.

A mechanism was developed by (Hidhaya S. et al., 2012), for the detection of SQL injection. By employing a Reverse proxy and MD5 algorithm to watch SQL injection in input. Using rules of grammar expressions for checking SQL injection in URL's. No changes are done in the application's source code by their method. Investigating and decreasing the attack is automatically done. The increasing in the number of proxy servers makes web applications able to handle any number of requests with no delay of time, and makes it able to protect the application from SQL injection attack.

A prevention technique against DDoS attack on REST based web service, was presented by (Lad and Baria, 2014), in this technique, resources were represented by a special URL that was generated by a part of the core set of HTTP orders: Put, Post, Get and Delete. Web services which are REST based perform DDoS easily. It'll monitor the behavior of the IP address, by employing a number of requested URL and time Interval Analysis which is based on threshold.

A technique for automated detecting weaknesses in web application and preventing many attacks on web application was proposed by (Dr. Meshram, 2012). Its function is monitoring all data incoming or outgoing in the application and blocking attacked related to web like SQL injection attacks, Directory traversal attacks, Cookie poisoning, Buffer Overflow attacks, Forceful browsing and Cross Site Scripting attacks. An application firewall tool was presented for protecting applications from being hacked.Proposed a technique for prevention and detection of intrusion by the improvement of reverse proxy .

Double Guard is an application developed by (Reddy et al., 2015), used for checking the intrusions in multi tier application. This application is used for back-end and front-end and its independent, it is also operated in dynamic and static servers in the web, these servers provide better protection for the application and information.

An IDS system which has the function of predicting the actions related with user across front-end web server and back-end repository, was proposed by (Namratha et al., 2013), by keeping in touch with www and going after what the database asks for, what it does is ferreting out attacks whose independent IDS won't be able to identify. The restrictions of virtually any multitier IDS related to training consultations and covering features are quantified.

An approach whose base is learning was presented by (Laranjeiro et al, 2010), for securing web services against SQL and XPath Injection attacks. Valid requests patterns were learnt by the approach, and that is called the learning phase, then it was able of detecting and aborting requests which might harm the server, which is called the protecting phase. Some heuristics might be used to deal with suspicious cases when there isn't a possibility to have a finished learning phase. The technique was executed to keep TPC-App services safe, and for opening source service effect.

3. Proposed System

The design of the proposed system aims to provide the best warning and preventing methodology with the maximum security measures for all the integrated clients. The main idea is based on detecting the attack on one of the integrated hosts and passing down the information to the centralized command and control center which will send the information to the other integrated hosts, in order to take actions against the same attacker and the same type of attack as one fully integrated system. So basically the system role begins whenever a host detects an attack and send the needed information to begin sending the information to other clients and take the needed countermeasure in order to prevent such events. Moreover the system has a backup measures to prevent one point failure in the whole system such as having another centralized command and control center in-case of crash or shutdown or any-type of system failure in the main command and control center, the backup command and control center goes up and takes control of the integrated clients to make sure that is the system performance doesn't undergo any changes and to make sure that the system keeps functioning normally. The main goal is achieved by sharing the attack information with other clients through a new genre of modified web application firewall which has the ability to send and receive the threat information also take actions depending on these provided information.

The flow chart of the proposed system is illustrated in figure 1.

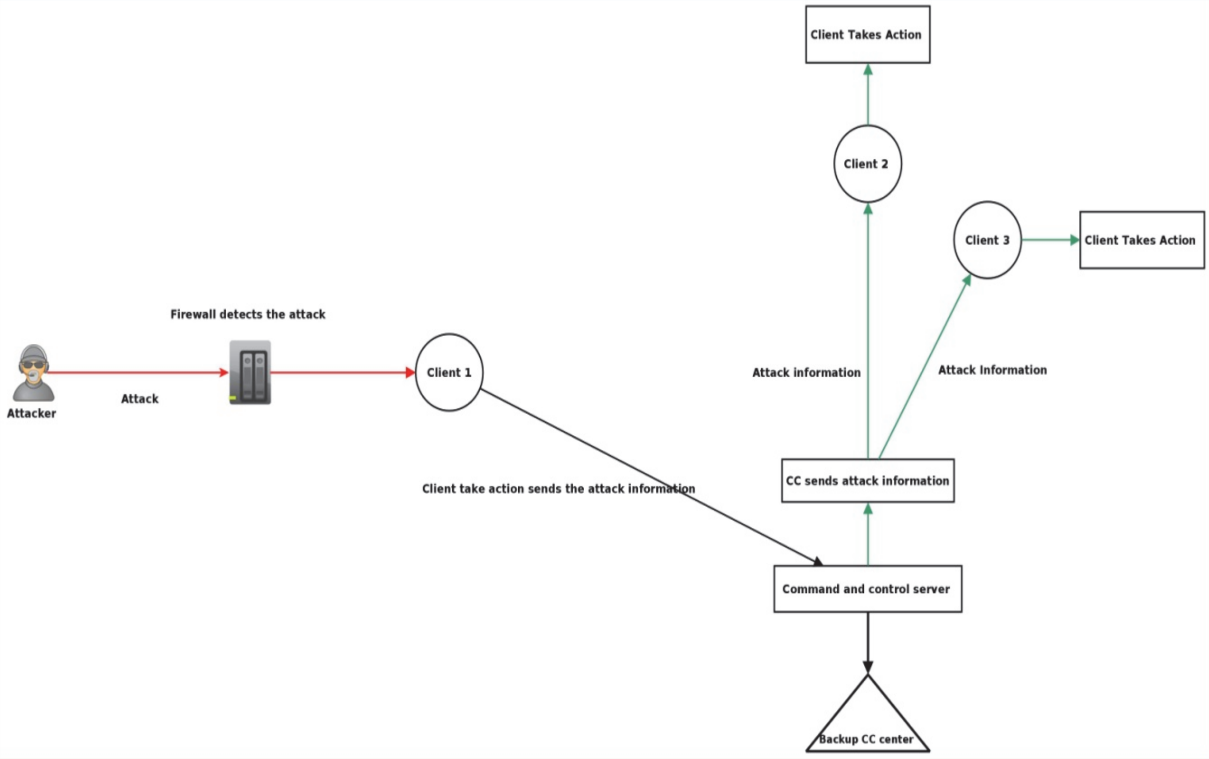

Figure 1. Flow chart of the web firewall mechanism

The system will act as soon as an attack is detected and take action against the threat also notify other firewalls after sharing the information attack through the command and control center, in order to take actions against the threat. The first step to trigger the system is to detect the attack and simply after that sending the detected log information to the command and control center which will pass the information to all of the connected clients to take the known precocious steps against the threat depending on the type of the firewall genre and rules assigned to it, The centralized web firewall system is simply an enhancement that can apply on all of the web firewall technology that has added a new approach and made it different from the standalone web firewall services to enhance web firewall detecting and control process capabilities. The integrated clients will automatically update their information about the new detected threat and take actions according to the received information inside the attack log file which contains; type of attack, date and time and The IP-Address of the attacker. The following figure illustrate the process:

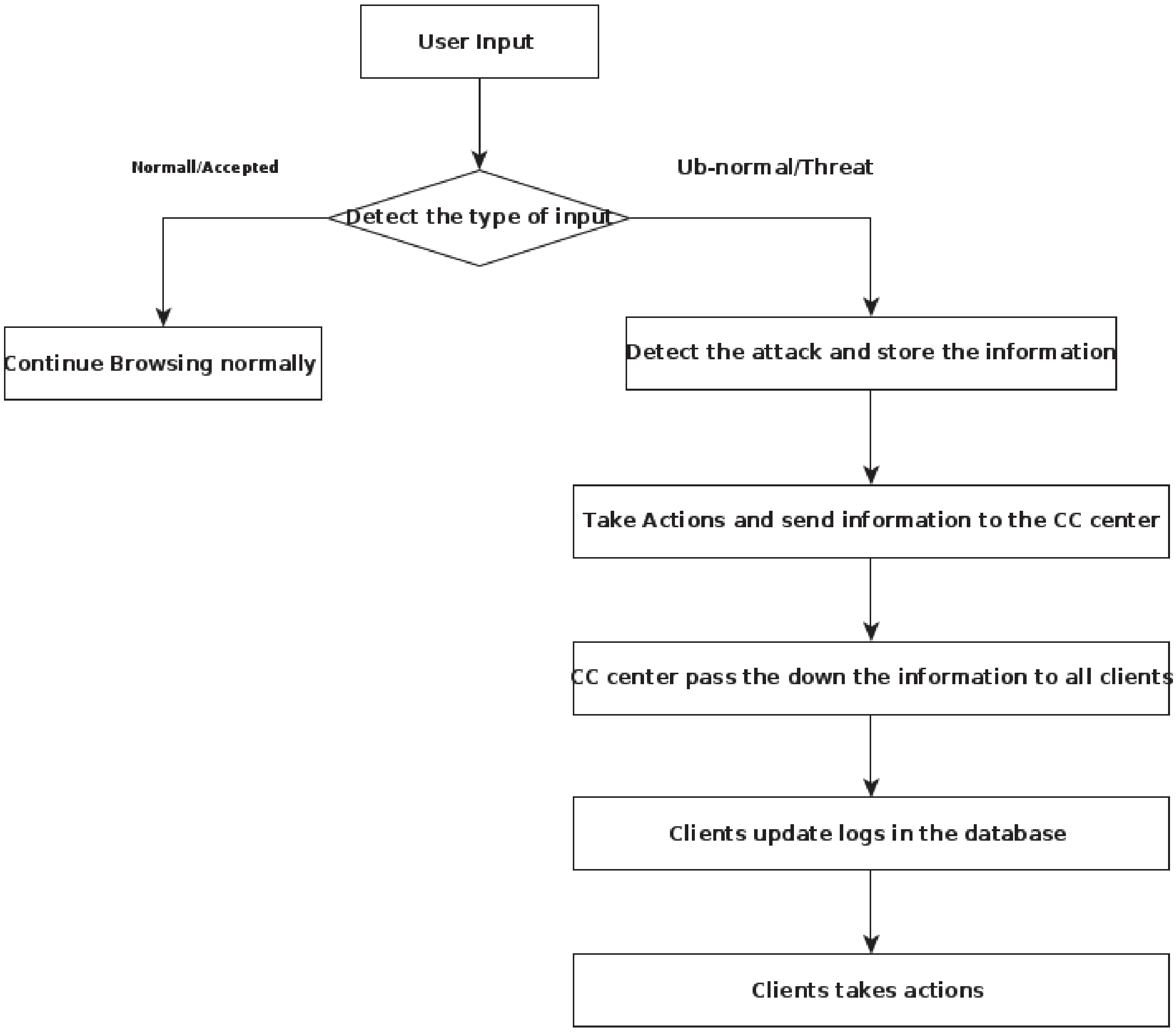
User Input
Normall/Accepted
Ub-normal/Threat
Detect the type of input
Continue Browsing normally
Detect the attack and store the information
Take Actions and send information to the CC center
CC center pass the down the information to all clients
Clients update logs in the database
Clients takes actions

Figure 2. Centralized web firewall system

3. Results

To test the proposed system a web firewall has been designed and implemented using all the previous steps and mechanisms shown in figure 1 and figure 2, the implementation was done using php language, the system has been implemented on 50 hosting server each of them integrated with each other, several attacks has been performed on different clients. The system was success in 100% to prevent the attack on different clients after it has been detected and distributed to all of the connected clients in the system, also the system was able to detect attacks such as (xss,Ddos,sql injection). And it has prevented all the threats mentioned before. Any further attack for different clients by the same attacker was prevented after banning the IP-Address of the attacker. All of the clients' firewalls took the same action before they get the attack and by this process of synchronization and distributing the log attack we were able to enhance the security of different hosts on different networks and web applications hosted on different servers and different platforms using this easy to use solution.

Table 1. Comparisons between researches Web detection and prevention methods

Author name	Method used	Year	aim	Implementation environment	result
NAVALE et al.	The method use counter base algorithm and Access Pattern algorithm that was executed in	2014	Predicting the attacker future behavior	A Hadoop framework	technique for DDos detection

	Hadoop framework for predicting the action that the attacker will do next				
Neha Lad	It'll monitor the behavior of the IP address, by employing a number of requested URL and time Interval Analysis which is based on threshold.	2014	A prevention technique against DDoS attack on REST based web service. by a special URL that was generated by a part of the core set of HTTP orders: Put, Post, Get and Delete	Microsoft .Net and its technology ASP .Net MVC 4 Web Application and Microsoft SQL server.	A prevention technique against DDoS attack on REST based web service
Ashwini R and Pawar, S.	self-governing IDSs, DoubleGuard form's container-based IDS with manifold input streams to create alerts.	2014	intrusion detection system which is built the models of normal behavior for multitier web applications from both front-end web (HTTP) requests and back-end database (SQL) queries	of lightweight virtualization and an Apache web server with MySQL	An intrusion detector system
ShikhaGarg and PoojaNarula,	using unclear parameters and some rules. The technique they use consist of the steps mentioned next, collection data set which is meant for training, and also a cheat sheet for the analysis, then data set trained for the interrogation, generation number 3 of the patterns and keys , fourth is that each parameter gets a full analysis. the final step is fetching the results and data interpretation	2014	Early detection and mechanism for the avoidance of the SQL Injections Attacks.	implemented the mechanism with the use of Fuzzy Parameters and set of rules	to detect SQL injection attacks and avoid them early
VIJENDER PUPPALA R.	web services moved to a multi-tiered design where in the application front-end logic is run by the web server, and wherein we outsource data to database or file server	2013	IDS system that is monitoring the network actions of the sessions by users across both front-end web server and back-end database	Double Guard	IDS system that is monitoring the network actions of the sessions by users across both front-end web server and back-end database
Shenb agalakshmi	models the network behavior of user sessions across both the front-end web server and the back-end database	2013	Intrusion Detection Systems endeavor at detecting attacks against computer systems and networks that offer techniques for modeling and distinguish normal and abusive system behavior	Apache web server with MySQL and lightweight virtualization.	enhances the security

Table 2. Comparisons between Web detection and prevention methods

Firewall name Feature	The ESAPI	Web Castellum	Open WAF	Mod Security	Base Firewall	Ninja Firewall	Centralized Firewall
Type of attack that is detected and prevented	Web Application based Attacks	Web Application based Attacks	Web Application based Attacks	Web Application based Attacks	Web Application based Attacks	Web Application based Attacks	Web Application based Attacks
Runtime detection and	yes	yes	yes	yes	yes	yes	yes

prevention							
Implementation language	Java	Java	Java/Html	Java	PHP	PHP	PHP
Action and response type	Automatically	Automatically	No action	Automatically	Automatically	Automatic	Manual /Automatic
Message Alert speed	No alert Message	Instant	No alert Message	No alert Message	No alert Message	Instant	No Alert Message
Methods of alerting admins	Web based	Web based	Web based	Web based	Web based	Web based	Web based /SMS / Email
Blocking of the attacker	yes	yes	yes	yes	yes	No	yes
Saving log file and history of attack	yes	yes	Yes	yes	yes	yes	Yes
Logged Data Management interface	yes	yes	yes	yes	yes	no	yes
Centralized Logged Firwalls	No	Yes	No	No	No	No	Yes

4. Conclusion

The proposed system for the centralization process of web firewalls, enhances the process of detecting and preventing the web application based attacks through making the standard standalone web firewalls work together as one fully integrated system, simply by updating and distributing the attack log to all firewalls connected in the system. Although every Web Firewall contain its own log attack and can work independently which will enhance the functionality and reduce the possibility of attack in all of the integrated system. And increased the possibility of reducing and preventing of many types of attacks.

References

Aljawarneh, S., Laing, C., & Paul, V. (2013). *Verification of Web Content Integrity: A new approach to protecting servers against tampering.* School of Computing, Engineering & Information Sciences Northumbria University, Newcastle upon Tyne, 1-6. https://doi.org/10.13140/2.1.4225.6648.

Ashwini, R., et al. (November 2014). Enhanced Security Approach for Detecting Intrusions in Multitier Web Applications. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 3*(11), 3850 -3854.

Chavan, S., & Meshram, B. (March2015). Classification of Web Application Vulnerabilities. *International Journal of Engineering Science and Innovative Technology (IJESIT), 2*(2), 226-234.

Chou, T. (June 2013). security threats on cloud computing vulnerabilities. *International Journal of Computer Science & Information Technology (IJCSIT), 5*(3), 79-88. https://doi.org/10.5121/ijcsit.2013.5306

Garg, S., & Narula, P. (2014). A novel Approach and Implemention Of Secured Algorithm Against SQL Injections. *International Journal of Enterprise Computing and Business Systems, 4*(1), 1-7.

Hidhaya, S., & Angelina, G. (2012). Intrusion Protection against SQL Injection Attacks Using a Reverse Proxy. Recent Trends in Computer Networks and Distributed Systems Security Communications. *Computer and Information Science, 335*, 252-263. https://doi.org/10.5121/csit.2012.2314.

Laranjeiro, N., et al. (2010). A Learning-Based Approach to Secure Web Services from SQL/XPath Injection Attacks. Pacific Rim International Symposium on Dependable Computing IEEE, 191-198. https://doi.org/10.1109/PRDC.2010.24

M. Rababah, O., K. Al Hwaitat, A., Al Manaseer, S., Fakhouri, H. N., & Halaseh, R. (August2016). Web Threats Detection and Prevention Framework. *Communications and Network, 8*, 170-178. https://doi.org/10.4236/cn.2016.83017

Meshram B. and Patil J. (September 2014). Web Gladiator a Web Application Firewall. *International Journal of Emerging Technology and Advanced Engineering, 4*(9), 336 -345.

Namratha, T., & Venkatramulu, S. (2013). Identifying Distributed Dos Attacks in Multitier Web Applications. *The International Journal Of Engineering And Science (IJES), 2*(10), 17-22.

Navale G., et al. (Feb,2014). Detecting and Analyzing DdoS Attack Using Map Reduce in Hadoop. *International Journal of Industrial Electronics and Electrical Engineering, ISSN: 2347-6982, 2*(2), 56-88.

Neha, L., & Baria, J., (2014). DDos Prevention on Rest Based Web Services. *(IJCSIT) International Journal of Computer Science and Information Technologies, 5*, 7314-7317.

Peotta, L., Holtz, M., Deus, F., & Sousa, J. R. (Feb 2011). A formal classification of internet banking attacks and vulnerabilities. *International Journal of Computer Science & Information Technology (IJCSIT), 3*(1), 186-197. https://doi.org/10.5121/ijcsit.2011.3113 186.

Puppala, V., et al. (2013). Detecting Intrusions in N-Tier Web Applications by using Double Guard Approach. *The International Journal of Engineering and Science (IJES), 2*(2), 2014-2018.

Raikar, D., et al. (October 2012). Preventing SQL Injection Attacks Using Combinatorial Approach. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 1*(8), 46- 52.

Reddy, D., et al. (January 2015). Detecting Attacks and Protecting From single To Multi Level application. *International Journal of Advanced Technology in Engineering and Science, 3*(1), 478-484.

Shenbagalakshmi, G., & Muneeswaran, K. (January 2013). Intrusion Detection in Multitier Web Applications. *International Journal of Emerging Technology and Advanced Engineering, 3*(1), 346 -351.

Copyrights