Intrusion Detection and Prevention by using Light Weight Virtualization in Web Applications

Prof. Dipashree Sonavale[1] Nehal Agarwal[2] Aniket Yadav[3] Vardhan Pathare[4] Amar Mhatre[5]
[1,2,3,4,5]Department of Information Technology
[1,2,3,4,5]MGMCET, India

Abstract— Web services and applications is an important part of our daily life providing communication and management of information from anywhere. With the increase in web services, the applications of web moved to a multitier design. These services have web server at front end that runs the applications and data server at back end. Due to their abundant use in managing information, the web services have always been the target of attacks such as future session attack, sql injection attack, cross site scripting attack etc. To prevent such attacks in web services, IDS based on anomaly detection that depends on honey pot can be used to detect the unknown attacks by identifying the anomalous behavior that differs from the behavior of legitimate user.

Key words: Multitier, Web Services, Honey Pot, IDS

I. INTRODUCTION

The usage of web services and applications has increased ranging from individual to large organizations. It allows the users to interact remotely with the information using the web browser. Our day to day tasks such as banking, travel, social networking etc are all done through web. These applications have always been the target of attacks that exploit the confidentiality and integrity of services delivered to the users. These attacks use the web request to destroy the back end data. To protect the web services several security mechanisms have been developed. Intrusion detection system can be used to detect the unknown attacks by analysing the traffic patterns. It can be used to detect the abnormal behavior that deviates from the normal network behavior. If an attacker log into the web server using the normal user access can find a way to issue privileged database query. In this paper, we present an approach that create models of isolated user sessions that includes front end(http) and back end(sql) network transactions. Each of the user session is assigned to a dedicated container. We can use the ID of the container to match the web request with the data base queries. The isolated user sessions prevent the attacker from compromising the session.

II. INTRUSION DETECTION SYSTEM

Intrusion detection system monitors network traffic and suspicious activity and also responds to malicious traffic by blocking the user or IP address from accessing the network. Intrusion detection assumes that the behavior of the intruder differs from that of a legitimate user. The classes of intruder are Masquerader: An individual who does not have authorized to use computer penetrates system's access control to exploit legitimate user account. Misfeasor: A legitimate user accesses the data, programs and resources of the authorized user's privileges. There are two types of intrusion detection system Anomaly detection and misuse detection. Anomaly detection involves collection of data relating to the behavior of legitimate user over a period of time. The statistical tests are applied to the observed behavior to determine with a high level of confidence whether the behavior is not legitimate user behavior. Threshold detection: This approach involves defining thresholds, independent of user. Profile based: A profile of activity of each user is developed and used to detect changes in the behavior of individual accounts. Our approach belongs to anomaly detection depends on training phase to build correct model. Misuse detection: It is the reverse approach of the anomaly detection by defining the abnormal behavior first and then the normal behavior uses the attack signatures. The previous approaches have detected intrusions by analysing the source code. Our approach does not require knowing the application logic or analysing the source code.

III. TYPES OF ATTACKS

A. Privilege escalation attack

Privilege means what is allowed to do. Common privileges include viewing, modifying and editing the files of the system. Privilege escalation is the process of exploiting a bug in an operating system or software applications to gain access to resources that protected from an application or user and perform unauthorized actions. The attacker logs into the web server as the normal user and upgrades the privileges, triggers the queries to obtain the data of the administrator.

B. Hijack future session attack

This type of attack uses the http request to take over the web server. This is also known as cookie hijacking, exploiting the valid computer session to gain authorized access in a computer system and hijack all the legitimate user session to launch the attacks. The attacker can also eavesdrop or drop the user requests. Session hijacking attack can also be categorized as spoofing, man in the middle attack, denial of service attack and replay attack.

C. Injection attack

This type of attack takes the advantage of the improper coding to execute attacker commands. Using sql the web applications interact with back end data base to retrieve the user's data. An attacker delivers malicious sql query segments in order to alter the query sent to the back end data base. This is a technique to attack the websites.

IV. RELATED WORK

Intrusion detection systems work in isolation from access control for the applications to protect the system. The lack of coordination between the components prevents detecting and responding to ongoing attacks in real-time before they cause damage. To overcome this problem in detection dynamic authorization technique used to support fine-grained access control and application level intrusion detection and response capabilities, experience with integration of the Generic

Authorization and Access Control API to provide dynamic intrusion detection and response for the Apache Web server. This [1] paper describes the implementation of the Vulnerability &Attack Injector Tool that allows the automation of the entire process. This tool is used to run a set of experiments that describes the feasibility and the effectiveness of the proposed method. The experiments include the evaluation of coverage and false positives of an Intrusion Detection System for SQL Injection attacks and the assessment of the effectiveness of two top commercial web application vulnerability scanners. This [5] paper proposes a system composed of a web-based anomaly detection system that is a reverse HTTP proxy and a database anomaly detection system. The process of combining web-based anomaly detector and SQL query anomaly detector increases the detection rate of our system. To address the increase in the false positive rate, an anomaly-driven reverse HTTP proxy to serve anomalous requests that does not require access to sensitive information. We developed a prototype and evaluated its applicability with respect to several existing web-based applications. Results show that our approach is feasible and effective in reducing both false positives and false negatives.

V. SYSTEM ARCHITECTURE

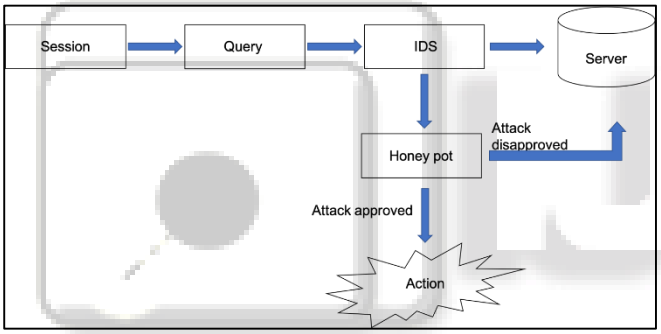

Fig. 1: Shows the block diagram of intrusion detection system with honey pot. The user enters into the login or a particular session. The http request of user is analysed by the IDS before sent to the server. IDS is based on the assumption that the behavior of the intruder deviates from the normal behavior. If no suspicious activity is detected the query for database will be sent directly to the server. When the IDS detect the malicious activity, it will be sent to honey pot for approval. If the attack is approved, then the honey pot tracks the activity of the intruder and responds to it. It will block the intruder from accessing the privileges.

A. Traffic Collection

In this module each user session is assigned to a different container and this was a design decision. For instance, a new container per each new ip address of the client is assigned. In this container, the recycled is based on events or when sessions time out.

B. Building Honey Pot

This container based and session-separated web server architecture not only enhances the security performances but also provides us with the isolated information flows. It allows us to identify the mapping between the web server requests and the following DB queries and use such a mapping model to detect abnormal behaviors on a session.

C. Attack Scenarios

At the web server, Honey pot is deployed on the host system and cannot be attacked directly. These Honey pots will not be attacked at the database server either that the attacker cannot completely take control of the database server. The IDS analyzes the information it gathers and compares it to normal users behaviour essentially, the IDS looks for specific attack that has already been documented like an SQL injection attack, cross scripting attack. Then, it classifies normal user and abnormal user by behavior and detects the intruder and also takes a necessary action based on intrusion.

VI. CONCLUSION

We proposed an intrusion detection system that depends on honey pot builds the models of normal behavior for multitier web applications considering both front-end requests and backend db queries. It provides a container based IDS with multiple input streams to produce the alerts and can identify a large number of attacks with the minimal false positive rate. This achieves better characterization of the system for anomaly detection and it is more effective for both the static and dynamic web service.

REFERENCES

[1] Jose Fonseca, Marco Vieira, Henrique Maderia, "Evaluation of Web Security Mechanisms using Vulnerability & Attack Injection", IEEE transactions on dependable and security computing, 2013.
[2] William G.J. Halford, Alesaandro Orso, "WASP: Protecting Web Applications Using Positive Tainting and Syntax-Aware Evaluation", IEEE transactions on software engineering, January 2008.
[3] Tatyana, Clifford Neuman, Dongho Kim, "Integrated Access Control and Intrusion Detection for Web Servers", IEEE transactions on parallel and distributed systems, vol 14, September 2003.
[4] Correlation", IEEE transactions on dependable and security computing, vol 1 September 2004.
[5] William Robertson, Christopher Kreugal,"Reducing errors in the anomaly-based detection of web-based attacks through the combined analysis of web requests and SQL queries", Journal of computer society, 2009.
[6] T.Venkat, Narayana Rao, V.Tejaswini. "Defending against vulnerabilities and cross site scripting", Journal of Global Research in Computer Science, vol 3, may 2012.
[7] Peyman Kabiri, Ali A.Ghorbani,"Research on Intrusion Detection and Response: A survey", International Journal of Network Security, vol.1, 2005.
[8] C.Anley, Advanced SQL injection in SQL server applications, Next generation security software Ltd, 2002
[9] Paul K Harmer, Paul D.Williams,"An artificial immune system for computer security applications", IEEE transactions on evolutionary computation, vol 6, 2002.

[10] Jose Fonseca, Marco Viera, Henrique Maderia,"Analysis of field data on web security vulnerabilities", IEEE transactions on dependable and secure computing, 2013.

