Intrusion Detection in Multitier Web Applications Using DoubleGuard

Neha A. Mohadikar
Department of Information Technology,
Sinhgad College of Engineering, Pune
University of Pune, India.
nehamohadikar103@gmail.com

Prof. M. V. Nimbalkar
Department of Information Technology,
Sinhgad College of Engineering, Pune
University of Pune, India.

***Abstract*— Nowadays, use of various internet services and internet applications are increases in huge amount which results the increase in data complexity. Hence web service focuses on multi-tiered design where web server serve as front end and database server serve as back end. Various attacks target directly the back end database and try to accomplish the personal information. Hence, there is need to provide more security to both web server and database server. This paper proposes an efficient intrusion detection and prevention system, called doubleguard system, which is used to detect attacks in multitier web applications. This IDS system models network performance of user sessions crosswise both front-end web server and back end database server. To achieve this, a Lightweight virtualization technique is used, which tracks the information flow from the web server to the database server for each session. It allocates a dedicated container to each user's web session. Each container is associated with an independent container ID and hence it enhances more security. Doubleguard is implemented by using Apache webserver with MySQL. The system built a well-correlated model for websites and detects and prevents different types of attacks.**

Keywords- Multitier web application, Intrusion detection and prevention system, Lightweight virtualization, Dedicated container, Attacks.

I. INTRODUCTION

As we know today's world is the technology world and therefore usage of internet as well as different web services increases tremendously. A web service as well as their different applications provides great usability to user. So, their popularity, availability and usage increased day by day. As this is technology era, we have to face web services and their applications in various fields like banking, shopping, financial sectors, travelling. While using of that services larger amounts of data are storing and retrieving from database of web services. Many web applications are most of the time open to use and therefore it is easy to find out security drawbacks to attacker, [1] so it is then turned as potential victims of insecure web applications by using various security attacks. Nowadays because of increasing use web and their services as well as their usage complexity, web services required shifting to multi-tiered design approach where the web server runs web application at front end and data is deploy to database server. To provide more security to multitier web services intrusion detection systems (IDS) are largely used and it detects various attacks by using matching misuse traffic patterns or signatures. Intrusion Detection System made up of set of tools that they can be useful to prevent and detect attacks of intrusion.

There are two IDS that is web IDS and database IDS, most of times used to detect unusual network traffic one at a time sent to any of them. But these IDS can't identify cases in which traffic will be used to attack web server and database server. Sadly, in the present multithreaded web server architectures, it is not likely to detect or figure out such unusual mappings between traffic of web server and database server because traffic can't be crystal clearly identified to user sessions. So, in this paper Doubleguard approach is described. Doubleguard is the system which is used to detect attacks in multitier web services. This technique tracing out the network behavior of user sessions between front end web server and back end database server. It administers web requests and figure out attacks which independent IDS would unable to detect. In this approach it uses lightweight virtualization phenomenon. Lightweight virtualization assigns a dedicated container to every user's session. It is being at virtual computing environment and might be isolated. The web container Id will be used to perfectly assist the web request with different database queries. Hence, doubleguard system develops causal mapping correlation by taking into account both database traffic and web server.

II. RELATED WORK

A. Introduction of IDS System

An essential part of our day-to-day life is Information Technology. Various web services and applications work on front end and back end server. Front end consist of application user interface logic and back end server consist of database for particular user data [1]. All the vital information is stored on database server so attacker shifted their focus from front end to back end. IDS system is a device or software application that monitors network or system activities for malicious activities or policy violations and produces alerts [2]. A class of IDS detects unknown attacks by identifying the abnormal behavior of the network traffic action from previous behavior of IDS training phase. The attackers abnormal network traffic can

be detected by database and web IDS. It stops the attacker to enter within the server. But when attacker used the normal traffic to attack on the web server and data server then this type of attack is unable to detect by IDS [2].

B. Multitier Web Application

IDS systems have been widely used in order to protect multi-tier web services. Multi-tier web architecture often referred to as n-tier architecture. In figure 1, at the database side, we cannot able to tell which transaction corresponds to which client request. Also, the communication between the web server and the database server is not separated and we cannot understand the relationships among them.

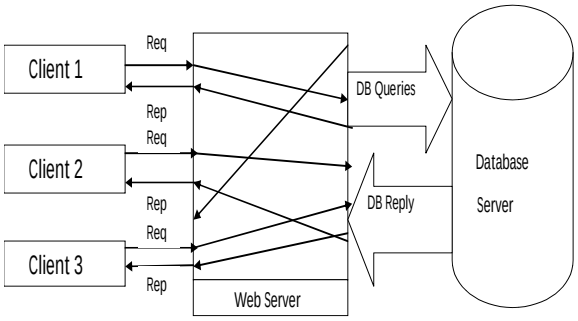

Figure 1. Classic Three-tier Architecture

In multitier web architecture, firewall is used to protect the back-end database at behind and web application made it possible for user to access set of services from web servers which are remotely accessible over the Internet .The current IDS system installed at web server and at database server are unable to detect intrusions [3] where a normal traffic is used for attacking back end database. It is also found that these IDS cannot detect cases wherein normal traffic is used to attack the web server and the database server. IDS are protected from direct remote attacks; but then also the back-end systems are vulnerable to attacks that use web requests as a means to exploit the back-end.

But, the doubleguard system uses a new container-based web server architecture that enables us to separate different information flows by each session. It track the information flows to database server. The main purpose of double guard system is to model the mapping patterns between database queries and http requests to detect malicious user sessions.

C. Existing Methodologies

1) SWADDLER

SWADDLER [4] [7] is an approach which consists of different web application state. Web application state is the information related with single user session. This system operates in two phases, training and detection. Swaddler includes main components as sensor and analyzer. Sensor collects the data of web application state, i.e. it collects values of state variables, encapsulates these variables in an event and sent it to analyzer. Events are generated by sensor defines mapping between variable names and their current values. In training phase, profiles for application blocks are established using the events generated by the sensor, and in detection and prevention phase, these profiles are used to identify anomalous application states [4]. If an anomalous state is detected, the analyzer raises an alert message and it can immediately stop the execution of the application. This technique is vulnerable to mimicry attacks. Figure 2 shows Swaddler.

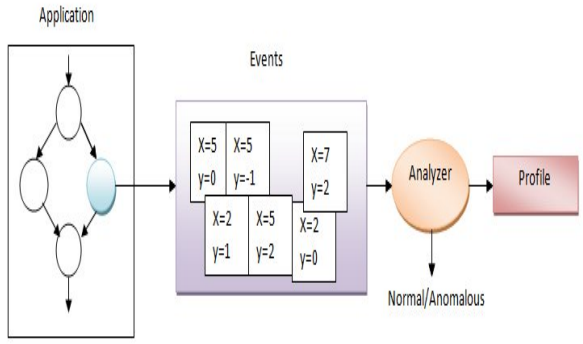

Figure 2. SWADDLER

2) Combined Approach for Analysis of Web Request and Database Request

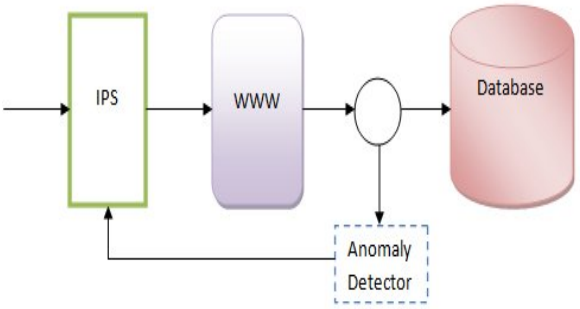

Figure 3. Combined approach for analysis of web request and database request

Figure 3 illustrates the combined approach for analysis of web request and database request which is an intrusion detection system that is implemented at both levels, at webserver and database server. In this approach, it includes SQL anomaly detector. This Intrusion prevention system blocks the requests that are found to be anomalous and SQL anomaly detector detects the normal looking malicious request that generate anomalous database queries [5]. Anomaly detector sends the feedback about detected attacks back to IPS. IPS receives feedback information and updates its configuration to improve its capability of detecting attacks. This approach includes the

addition of detection system at database level which detects malicious web request [7] that are mistakenly considered as normal. Hence the system is able to prevent future attack.

3) Histogram-Based Traffic Anomaly Detection

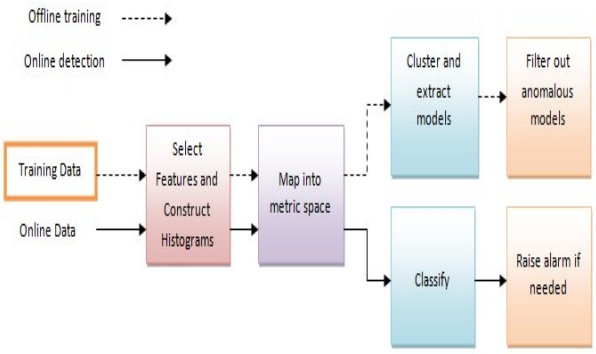

Figure 4. Histogram Based Approach

Figure 4 shows the histogram based traffic anomaly detection. This approach simulates different traffic features with the help of histogram [6]. It models the normal behavior of a network and follows the regular patterns and identifies variation from established model. Following are the steps included in this approach:

a. Select feature of training data such as IP address, source and destination address, port number etc. and construct histogram based on that feature.
b. Map the histogram into metric space, i.e. two similar histograms are close together and two dissimilar histograms are far away.
c. Cluster and extract models using clustering algorithm. Clustering is required to analyze the pattern of normal behavior. When we completed with clustering we need to categorize the clusters that are correspond to normal and abnormal behavior. The set of clusters that model the normal behavior of a network, we keep it as baseline.
d. Classify. In Detection phase, we compare observed network behavior with baseline. It computes a vector that encodes the network behavior for each feature. The network behavior is normal only if the vector falls within the scope of baseline cluster; otherwise network behavior is examined as abnormal.

In the existing system, both the web and the database servers are vulnerable. Attacks are come from the web clients. They launch application layer attacks to compromise the web servers they connect into. The attackers can bypass the web server to directly attack the database server. Attackers may take over the web server after the attack, and that afterwards they can obtain full control of the web server to launch subsequent attacks. Attackers could modify the application logic of the web applications, eavesdrop or hijack other user's web requests, or intercept and modify the database queries to steal sensitive data beyond their privileges.

III. PROPOSED SYSTEM

A. Problem Definition

Internet is an area that provides the information to everyone connected to it. But its advantage comes with various flaws like trust and security. The existing system also has various drawbacks. Hence, in order to protect multitier web service, an efficient system is needed.

The proposed Doubleguard system built a normality model that provides an effective mechanism to detect the different types of attacks. It forms a container-based ID with multiple input streams to produce alerts. This system achieved by isolating the flow of information from each web server session with a lightweight virtualization and detects a wider range of attack. Hence, proposed system provides an implementation of efficient IDS, Doubleguard, that detect different attacks by taking the structure of the web request and database queries. The system is modeled for websites.

B. Purpose

To study a system that will help to provide more security to multitier web applications with the help of normality model that provides effective mechanism which detects and prevents various types of attacks.

C. Objective

The main objective of the system is to provide more security to both front end and back end. Also the system detects and prevents various attacks by building a normality model and serves more security to multitier web applications.

D. System Architecture

Figure 5 illustrates the doubleguard system architecture.

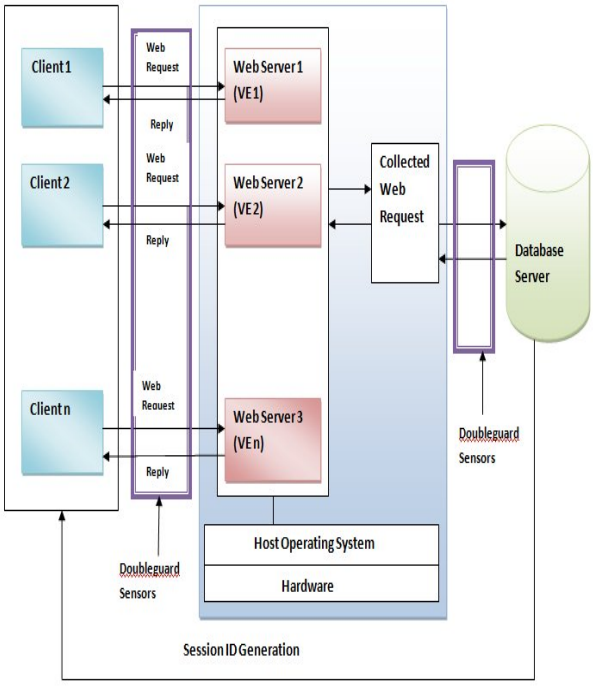

Figure 5. System Architecture

DoubleGuard is an Intrusion Prevention System (IPS) that models the network behavior of user sessions across both the front-end web server and the back-end database. Doubleguard sensors are used at both front end and back end detection process.

The new Session based web server architecture enables us to separate the different information flows by using light weight virtualization. Within a light weight virtualization [1] environment we ran many copies of web server instances so that each one isolated from the rest and separate the information flow. This provides a means of tracking the information flow from the web server to the database server for each session. It detects SQL injection attacks by taking the structure of the web request and database queries without looking into the values of input parameter. It utilizes the ID to separate session traffic as a way of extracting and identifying causal relationship between web server request and database query event. This approach dynamically generates new session. This system chooses to separate communications at the session level so that a single user always deals with the same web server instances. Sessions represent different users and the communication of a single user goes through to the same dedicated web session, thereby allowing us to identify suspect behavior by both session and user.

It detect abnormal behavior in a session, will treat all traffic within this session as tainted and the session will be destroyed immediately [8]. It is possible to initialize thousands of sessions on a single physical machine, and these virtualized sessions can be discarded, quickly reinitialized to serve new sessions.

E. Algorithm

1. User registers to system
2. User performs login by entering user_id and password
   If (authenticated) then
   Send web request
   Else
   Invalid User
3. Assign separate container to user's web request
4. Check web request
   If (legitimate request) then
   Forward request to web server
   Else
   Block the request and add the user in block list
5. Web server generates query
6. Evaluate query
   If (unauthorized) then
   Generate alert and reject
   Else
   Forward query to database server
7. Perform query processing
8. Output

State Chart Diagram:

Figure 6 shows state chart diagram in which following six states are used.

T0= Request manager
T1= Authenticator
T2= Container manager
T3= Query analyzer
T4= Alert Generator
T5= Database
T6= Blocker

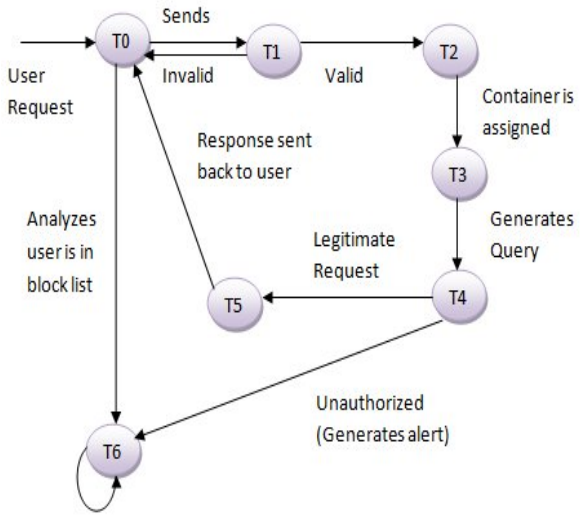

Figure 6. State Chart Diagram

IV. TYPES OF ATTACKS OVER WEB

A. Privilege Escalation Attack

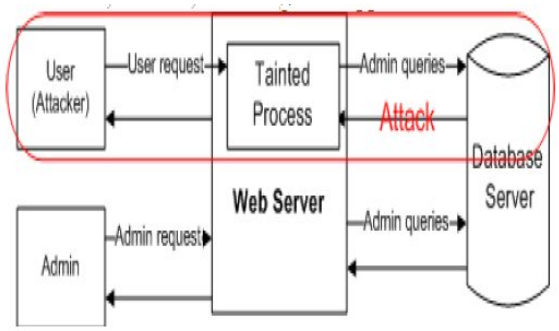

Figure 7. Privilege Escalation Attack

Figure 7 illustrates the privilege escalation attack. For a normal user, the web request r_u will prompt the set of SQL queries Q_u; for an administrator, the request r_a will activate the set of admin level queries Q_a. Assume that an attacker logs into the web server as a normal user, improves their privileges, and triggers admin queries as a result to obtain an administrator's data [9]. This attack cannot be detected by either the web server IDS or the database IDS because both r_u and Q_a are authentic requests and queries. This approach, can detect this type of attack in view of the fact that the DB query Q_a does not match the request r_u, according to the mapping pattern.

B. Hijack Future Session Attack

This attack is mainly aimed at the web server side. An attacker will usually grab the web server and then hijacks all subsequent valid user sessions to initiate attacks. For example, by capturing other user sessions, the attacker can snoop, send spoofed replies, and/or drop user requests [8] [9]. A session-hijacking attack can be further categorized as a Spoofing/Man-in-the-Middle attack, a Denial-of-Service or a Replay attack or a Packet Drop attack. Figure 8 illustrates the hijack future session attack.

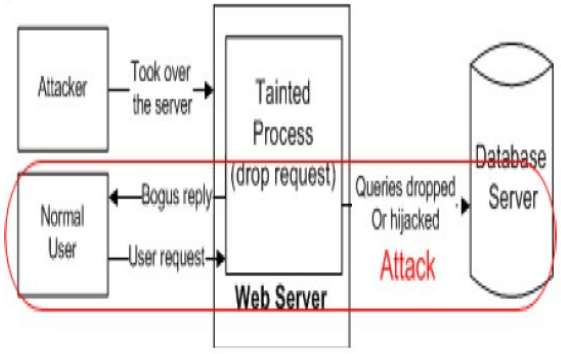

Figure 8. Hijack Future Session Attack

C. Injection Attack

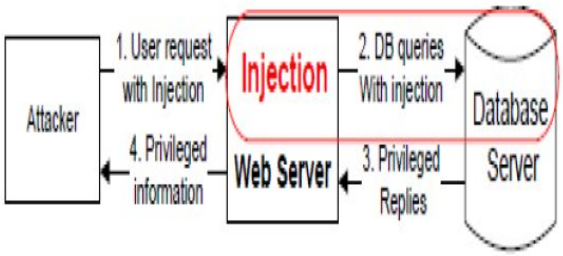

Figure 9. SQL Injection Attack

Figure 9 shows Injection attack. Attackers can use accessible vulnerabilities in the web server logic to infuse the data or string that contains the abuse and then use the web server to relay these exploits to attack the back-end database. Attacker enter particular line (' OR 1 = 1; --) into user name and enter any password he/she login into account and access system as authorized user. Since the SQL injection attack modifies the SQL queries structure, it would generate SQL queries in a structure that could be noticed as a deviation from the SQL query structure [9].

D. Direct DB Attack

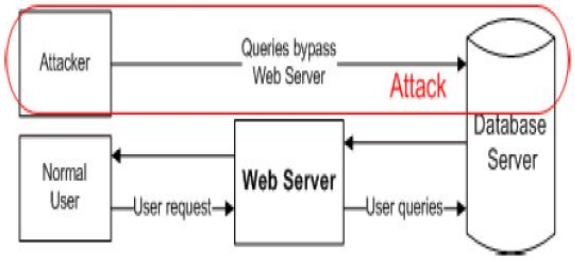

Figure 10. Direct DB Attack

Figure 10 describes direct DB attack. It is possible for an attacker to bypass the web server and connect directly to the database. An attacker takes over the web server [8] and submitting queries from the web server without sending web requests.

V. CONCLUSION

DoubleGuard is used to prevent the intrusions in multi tier web application. It builds normality model for multitier web applications. Unlike previous techniques this technique forms container-based IDS with multiple input streams to produce alerts. There will be a technique which uses lightweight virtualization to assign session ID to a committed container which is nothing but isolated virtual computing environment. Furthermore, there will precise detection of attacks such as Hijack Future Session Attack, Privilege Escalation Attack, DB Attack and SQL Injection Attack. Also the requests which breach the normality model that will be treat as an intruder. It is an application independent system and used for both front-end as well as back-end. It is also used for static and dynamic web server which provides better security for data and web application.

REFERENCES

[1] Meixing Le, Angelos Stavrou, Brent ByungHoon Kang,"DoubleGuard: Detecting Intrusions in Multitier Web Applications", IEEE Transactions On Dependable And Secure Computing, Vol. 9, No. 4, March 2014.

[2] M.Sujitha, P.Suganya, T.Shampavi, S.Anjanaa, "Dual Safeguard: Intrusion Detection and Prevention System in Web Applications", International Journal of Computer Application, Volume 67– No.9, April 2013.

[3] V. Felmetsger, L. Cavedon, C. Kruegel, and G. Vigna, "Toward Automated Detection of Logic Vulnerabilities in Web Applications," Proc. USENIX Security ACM ., 2010.

[4] M. Cova, D. Balzarotti, V. Felmetsger, and G. Vigna, "Swaddler: An Approach for the Anomaly-based Detection of State Violations in Web Applications" RAID 2007.

[5] G. Vigna, F. Valeur, D. Balzarotti, W. K. Robertson, C. Kruegel, E. Kirda, "Reducing errors in the anomaly-based detection of web-based attacks through the combined analysis of web requests and SQL queries", Journal of Computer Security, 2009.

[6] Andreas Kind, Marc Ph. Stoecklin, and Xenofontas Dimitropoulos, "Histogram-Based Traffic Anomaly Detection" IEEE transactions on network service management, vol. 6, no. 2, June 2009.

[7] Sachin J.Pukale, M. K.Chavan, "A Review Of Anomaly Based Intrusions Detection In Multi-Tier Web Applications", International Journal Of Computer Engineering & Technology, Volume 3, Issue 3, October - December (2012), pp. 233-244.

[8] Snehal Khedkar, Mangal Vetal, Surekha Kotkar, R. S. Tambe, " Security Model for Multi-Tier Web Application by Using Double Guard", International Journal of Emerging Technology and Advanced Engineering, Volume 4, Issue 2, February 2014.

[9] Shinde Jyoti R, Prof. Dabhade Sheetal V, "Advance Double Guard System: Detecting & Preventing Intrusions in Multi-Tier Web Applications", International Journal of Emerging Technology and Advanced Engineering, Volume 4, Issue 6, June 2014.