DOUBLEGUARD: DETECTING INTRUSIONS IN MULTI-TIER WEB APPLICATIONS

Prof.Roopali Lolage, Vaidehi Dalvi, Chaitali Chindarkar, Trusha Chodankar
Department of Information Technology
K.C.College of Engineering,Management Studies and Research,Kopri,Thane(e)
roopali.lolage@gmail.com
vaidehivdalvi@gmail.com
chaitalichindarkar@gmail.com
ctrusha28@gmail.com

***Abstract*— Now a day's computers are widely used for web application. Majority of transactions are done online these days. Thus data from web server and database server are prone to be hacked easily which relies to provide more security to web applications. Hacking is the gaining of access(wanted or unwanted) to a computer without the intention of destroying data or maliciously harming the computer. To overcome this issue DoubleGuard system is used.By monitoring both web and subsequent database requests, we are able to ferret out attacks that an independent Intrusion Detection System (IDS) would not be able to identify. IDS is used in DoubleGuard system to detect & prevent attacks. By using mapping of request and query IDS system can provide security for both web server and database server. The network behavior of user sessions across both the front-end web server and the back- end database is modelled by an IDS system. By isolating the flow of information from each web server session the DoubleGuard system solves the problem.**

Keywords—DoubleGuard, IDS, database server, hacking, web server

I. INTRODUCTION

Web-delivered services and applications have increased in both popularity and complexity over the past few years. Daily tasks, such as banking, travel, and social networking, are all done via the web. Such services typically employ a web server front-end that runs the application user interface logic, as well as a back-end server that consists of a database or file server. Due to their ubiquitous use for personal and/or corporate data, web services have always been the target of attacks. These attacks have recently become more diverse, as attention has shifted from attacking the front-end to exploiting vulnerabilities of the web applications in order to corrupt the back-end database system (e.g., SQL injection attacks an Intrusion Detection System lack in multi tiered Anomaly Detection (AD) systems that generate models of network behavior for both web and database network interactions.

Is multi tiered web applications web servers remotely accessible over internet? Back end systems i.e. Database servers are protected from direct remote attacks but they are susceptible to web attacks that consist of web requests as means to exploit back end website. To detect attack using normal traffic in multi tiered web architecture Double Guard system can be used. Double Guard system can employ normal models for user sessions that use both web front end server (HTTP) and database back end server (SQL). In this system each user's web session can be assigned to dedicated container using light weight virtualization technique. The container ID can then be used to associate web request with corresponding database queries. So in this system both web server and database server can be protected using casual mapping between them.

II. RELATED WORK

A. BASICS OF INTRUSIONS

A network Intrusion Detection System (IDS) can be classified into two types: anomaly detection and misuse detection. Anomaly detection first requires the IDS to define and characterize the correct and acceptable static form and dynamic behavior of the system, which can then be used to detect abnormal changes or anomalous behaviors. These stored forms can then be used to detect abnormal forms and anomalous behaviors. Anomaly detection relies on models of normal behavior of computer system. Behavioral models are built by performing statistical analysis of historical data and rule based approach. An anomaly detector then compares actual usage patterns against established profiles to identify abnormal patterns of activity.

Misuse detection systems take a complementary approach. Misuse detection systems are equipped with a number of attack descriptions. These descriptions (or "signatures") are matched against a stream of audit data to find evidence that the modeled attack is occurring. To detect intrusions an IDS uses temporal information. An IDS correlate events on timely basis, which runs the risk of mistakenly considering independent but concurrent events as correlated events.

Double Guard uses the container ID for each session to map casually related events whether they may be concurrent or not to overcome such a limitation.

The Figure1 shows simple intrusion detection system :

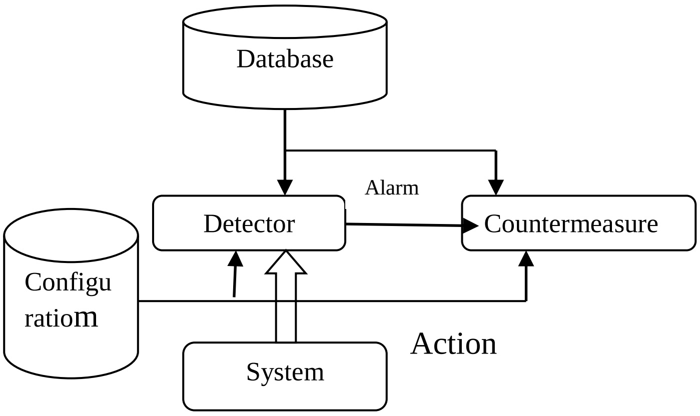

Fig.1. Simple intrusion detection system

The new container-based web server architecture in DoubleGuard enables users to separate the different information flows by each session. By using this it is possible to track the information flow from the web server to the database server for each user session. This approach also does not require users to analyze the source code or know the application logic. For the static webpage, DoubleGuard approach does not require application logic for building a model. However, for dynamic web pages full application logic is not required, to model normal behavior basic user operations is not required. To detect or prevent SQL or Cross Site Scripting (XSS) injection attacks validating input is useful. This is useful to the DoubleGuard approach, which can utilize input validation as an additional defense. However, by taking the structures of web requests and database queries without looking into the values of input parameters DoubleGuard can detect SQL injection attacks.In DoubleGuard, the container ID are utilized to separate session traffic as a way of extracting and identifying causal relationships between web server requests and database query events.

B. MULTI-TIER WEB APPLICATION

A multi-tier application is used to divide an enterprise application into two or more components that may be separately developed and executed. In general, the tiers in a multi-tier application include the following:

- The **presentation tier** for user interface generation and lightweight validation.
- The **business tier** for heavyweight processing, validation, business rules, workflow and interfaces to external systems.
- The **integration tier** for data transformation and persistence services.

The Figure2 shows multi-tier web application.

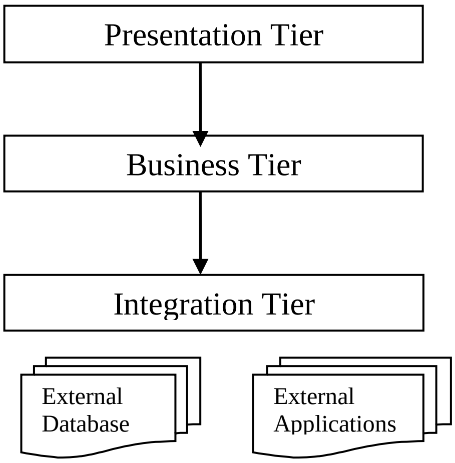

Fig.2. Multi-tier web application

III. DOUBLEGUARD SYSTEM ARCHITECTURE

To improve mechanism to detect intrusions in multitier web applications DoubleGuard system uses lightweight process containers referred to as “containers,” as ephemeral, disposable servers for client sessions. It is possible to initialize thousands of containers on a single physical machine, and these virtualized containers can be discarded, reverted, or quickly reinitialized to serve new sessions. In the classic three-tier model database side, it is unable to tell which transaction corresponds to which client request. The communication between the web server and the database server is not separated, and we can hardly understand the relationships among them .

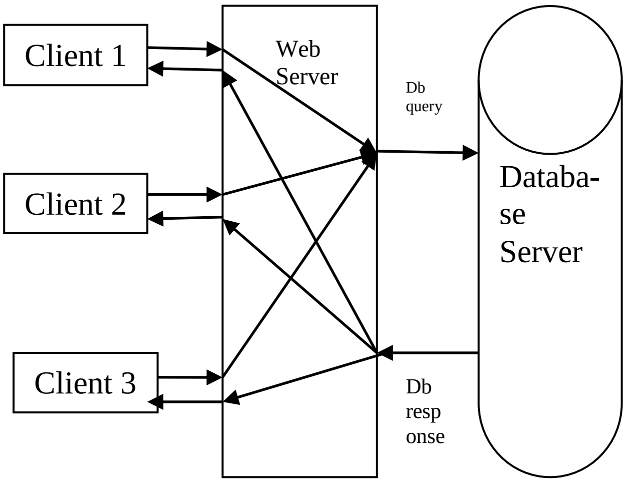

Fig.3. Classic 3-tier Model.The web server acts as the front-end,with the file and database server as the content storage back-end

This container-based and session-separated web server architecture not only enhances the security performances but also provides us with the isolated information flows that are separated in each container session. It allows us to identify the mapping between the web server requests and the subsequent DB queries, and to utilize such a mapping model to detect abnormal behaviors on a session/client level.

Once we build the mapping model, it can be used to detect abnormal behaviors. Both the web request and the database queries within each session should be in accordance with the model. If there exists any request or query that violates the normality model within a session, then the session will be treated as a possible attack.

A. Attack scenarios

DoubleGuard Intrusion Detection System is effective at capturing the following types of attacks:

1) **Privilege Escalation Attack** : A privilege escalation attack is a type of network intrusion that takes advantage of programming errors or design flaws to grant the attacker elevated access to the network and its associated data and applications.

2) **Hijack Future Session Attack**: This class of attacks is mainly aimed at the web server side. An attacker usually takes over the web server and therefore hijacks all subsequent legitimate user sessions to launch attacks. .For instance,by hijacking other user sessions, the attacker can eavesdrop, send spoofed replies, and/or drop user requests. A session hijacking attack can be further categorized as a Spoofing/Man-in-the-Middleattack,Denial-of-Service attack, or a Replay attack.

3)**Injection Attack**: SQL injection is a code injection technique, used to attack data-driven applications, in which malicious SQL statements are inserted into an entry field for execution (e.g. to dump the database contents to the attacker). SQL injection must exploit a security vulnerability in an application's software, for example, when user input is either incorrectly filtered for string literal escape characters embedded in SQL statements or user input is not strongly typed and unexpectedly executed. SQL injection is mostly known as an attack vector for websites but can be used to attack any type of SQL database.

4) **Direct DB Attack**: It is possible for an attacker to bypass the web server or firewalls and connect directly to the database. An attacker could also have already taken over the web server and be submitting such queries from the web server without sending web requests. Without matched web requests for such queries, a web server IDS could detect neither. Furthermore, if these DB queries were within the set of allowed queries, then the database IDS itself would not detect it either. However, this type of attack can be caught with our approach since we cannot match any web requests with these queries.

5) **XSS(Cross site scripting):** Cross Site Scripting allows an attacker to embed malicious JavaScript, VBScript, ActiveX, HTML, or Flash into a vulnerable dynamic page to fool the user, executing the script on his machine in order to gather data. The use of XSS might compromise private information, manipulate or steal cookies, create requests that can be mistaken for those of a valid user, or execute malicious code on the end-user systems. The data is usually formatted as a hyperlink containing malicious content and which is distributed over any possible means on the internet

B. Modeling for Static Websites

In the case of a static website, the non-deterministic mapping does not exist as there are no available input variables or states for static content. The algorithm generates the mapping model by considering all three mapping between http requests and the database requests that would happen in static websites.

C. Modeling of Dynamic Patterns

Since the algorithm for extracting mapping patterns in static pages no longer worked for the dynamic pages, we created another training method to build the model. First, we tried to categorize all of the potential single (atomic) operations on the web pages. For instance, the common possible operations for users on a blog website may include reading an article, posting a new article, leaving a comment, visiting the next page, etc. All of the operations that appear within one session are permutations of these operations. If we could build a mapping model for each of these basic operations, then we could compare web requests to determine the basic operations of the session and obtain the most likely set of queries mapped from these operations. If these single operation models could not cover all of the requests and queries in a session, then this would indicate a possible intrusion.

D)System Execution:

The working part should be consistent in all phases should be dependable at all conditions. Considering the discussion on most important thing to be studied is intrusion detection system. The system detecting changes in the web based document by using checksum detecting any errors in the data transfer web based services.The transferring of information from session to the database layer and detects intrusion by IDS system to increase the performance of the data transfer in a web services

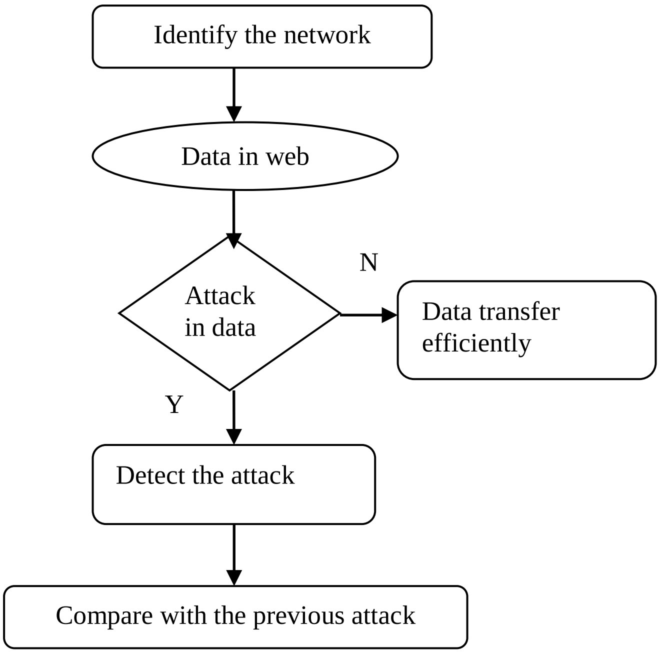

Fig.4. System Execution Flow chart

E) DoubleGuard Limitations

1) Vulnerabilities because of Improper Input Processing: To build a mapping model based on the structures of HTTP requests and DB queries the entire user input values are normalized in DoubleGuard. DoubleGuard cannot detect attacks hidden in the values, once the malicious user inputs are normalized. Based on the characterization of input values. DoubleGuard offers a complementary approach to those research approaches of detecting web attacks.

2) Distributed DoS:

DoubleGuard is not designed to mitigate Distributed DoS attacks. These attacks can also occur in the server architecture without the back-end database.

IV. FUTURE SCOPE

To protect multitier web applications against attacks is the main concentration point in Intrusion Detection System using DoubleGuard. We propose a prototype of DoubleGuard using a web server with a back-end DB. We also propose some modification to existing DoubleGuard to increase its performance by preventing cross site scripting attack, reliability in case of static and dynamic web sites. In our prototype, we propose to assign each user session into a different container; however, this will be a design decision.

V.CONCLUSION

We proposed an intrusion detection system that builds models of normal behavior for multi-tiered web applications from both front-end web (HTTP) requests and back-end database (SQL) queries. Unlike previous approaches that correlated or summarized alerts generated by independent IDSs', DoubleGuard forms container-based IDS with multiple input streams to produce alerts. DoubleGuard forms container-based IDS with multiple input streams to produce alerts. Such correlation of input streams provides a better characterization of the system for anomaly detection because the intrusion sensor has a more precise normality model that detects a wider range of threats. This is achieved by isolating the flow of information from each web server session with a lightweight virtualization. For static websites, a well-correlated model is built to detect different types of attacks. Moreover, for dynamic requests where both retrieval of information and updates to the back-end database occur using the web server front end.

DoubleGuard is able to identify a wide range of attacks with minimal false positives. As expected, the number of false positives depended on the size and coverage of the training sessions that are used.

REFERENCES

[1] Dependable and Secure Computing, IEEE Transactions on (Volume:9 , Issue: 4)
[2] www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 11, November 2013)
[3] http://nsl.cs.columbia.edu/projects/minestrone/papers/double_guard_TDSC11.pdf
[4] Computer Security Principles and Practice, by William Stallings, Pearson Education
[5] "Virtuozzo Containers," http://www.parallels.com/products/pvc45/, 2011.
[6] Wordpress. http://www.wordpress.org/..
[7] Wordpress bug. http://core.trac.wordpress.org/ticket/5487..
[8] http://searchsecurity.techtarget.com/tip/Intrusion-detection-basics
[9] http://www.srjis.com/srjis_new/index.php/using-joomla/extensions/components/content-component/article-categories/238-double-guard-detecting-intrusions-in-multi-tier-web-applications
[10] http://www.warse.org/pdfs/2013/ijns02222013.pdf