ISSN 2395-1621

Detection of Data Modification information using the MD5 Algorithm

[#1]Dipika Jagtap, [#2]Priyanka Kesare, [#3]Piyush Shah, [#4]Swapnil Gawai,
[#5]Prof. H.A.Hingoliwala

[1]dipikag.jagtap@gmail.com
[2]kesarepriyanka1996@gmail.com
[3]piyush7020@gmail.com
[4]swapnilgawai13@gmail.com

[#1234]Department of Computer Engineering,
[#5]Professor, Department of Computer Engineering,

JSPM's JSCOE Hadapsar, Pune,
Savitribai Phule Pune-41, India.

ABSTRACT

: In today's world we have many issues in internet security and privacy. We use internet in travelling, E-Commerce site, social media, banking, study etc. But we often face the problems with the privacy of the network system and private data. To accommodate this increase in application and data complexity, web services have moved to a multi-tiered design wherein the web server runs the application front-end logic and data is outsourced to a database or file server. IDS play a key role in computer security technique. But it also has drawbacks of its own. To overcome those drawbacks Duel Security technique is introduced based on ecommerce application. We implemented duel security using MD5 algorithm and hashing function, an in built web server of windows 7 ultimate, with My SQL Server. This System presents those models the network behaviour of user sessions across both the front-end web server and the back-end database. Implementing system monitoring both web and subsequent database requests. Most of the people do their transaction through web use. So there are chances of personal figures gets hacked then need to be provide more refuge for both web server and database server. For that purpose duel security system is used. The duel security system is used to identify & prevent attacks using Intrusion detection system. Duel security prevents attacks and prevents user account data from unauthorized updating from his/her account.

Keywords; Anomaly detection, virtualization, multi-tier web application, data leakage detection.

ARTICLE INFO

Article History

Received: 16th February 2019

Received in revised form :

16th February 2019

Accepted: 18th February 2019

Published online :

19th February 2019

I. INTRODUCTION

Database is a major component of each and every organization. But to store data in database is not sufficient for any organization, since they have to deal with all issues related to database, from which one of the main issue is database security. We deals with the basic approach that determines whether data stored in database is tampered or not. Any business cannot afford the risk of an unauthorized user observing or changing the data in their databases. Web services are widely used by people. Web services and applications have become popular and also their complexity has increased. Most of the task such as banking, social networking, and online shopping are done and directly depend on web. As we are using web services which is present everywhere for personal as well as corporate data they are being attacked easily. Attacker attacks backend server which provides the useful and valuable information thereby diverging front end attack. Data leakage is the big issue for industries & different institutes. It is very hard for any system administrator to find out the data leaker among the system users. It is

creating a serious threat to organizations. It can destroy company's brand and its reputation.

Most of the IDS examine the attack individually on web server and database server. In order to protect multi-tiered web services an efficient system call Intrusion Detection System is needed to detect attacks by mapping web request and SQL query, there is direct causal relationship between request received from the front end web server and those generated for the database backend. Dynamic web site allow persistent back end data modification through the HTTP requests to include the parameters that are variable and depend on the user input. Because of which the mapping between the web and the database rang from one to many as shown in the mapping model.

A web application framework (WAF) is a software framework that is designed to support the development of dynamic websites, web applications and web services. The framework aims to alleviate the overhead associated with common activities performed in Web development. One of the basic frame work is MVC (Model-View-Controller) which works in unidirectional and triangular fashion. The MVC architecture can be described as follows. Model–view–controller (MVC) [5] is a software architecture that separates the representation of information from the user's interaction with it. The model consists of application data and business rules, and the controller mediates input, converting it to commands for the model or view. A view can be any output representation of data, such as a chart or a diagram. The disadvantages of MVC architecture are

- Increases the complexity of solutions
- Increses the user interface code which increases the complexity in debugging
- Frequent changes in Model causes frequent updates in views which is a burden for programmer
- MVC applications being hard to deploy
- MVC restricted to making changes to data only on a local file system.
- For parallel development there is a needed multiple programmers
- Knowledge on multiple technologies are required

Multi-tier architecture (often referred to as n-tier architecture) is a client–server architecture in which client, logic, and data management functions are logically separated. It works in bidirectional and linear fashion. The most widespread use of multi-tier architecture is the three-tier architecture. Three-tier architecture has the following three tiers:

A. Client Tier This is the topmost level of the application. The presentation tier displays information related to queries made through browsing and displays the result.

B. Logic Tier It is implemented as a separate layer and it process and controls an application's functionality.

C. Data Tier This tier consists of database servers or file servers. Here information is stored and retrieved. This tier keeps data neutral and independent from application servers or business logic.

The **MD5 algorithm** is a widely used hash function producing a 128-bit hash value. Although MD5 was initially designed to be used as a cryptographic hash function, it has been found to suffer from extensive vulnerabilities. It can still be used as a checksum to verify data integrity, but only against unintentional corruption.

MD5 was designed by Ronald Rivest in 1991 to replace an earlier hash function MD4. The abbreviation "MD" stands for "Message Digest."

SQL injection is a code injection technique, used to attack data-driven applications, in which nefarious 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.

SQL injection attacks allow attackers to spoof identity, tamper with existing data, cause repudiation issues such as voiding transactions or changing balances, allow the complete disclosure of all data on the system, destroy the data or make it otherwise

unavailable, and become administrators of the database server.

To create a system for intrusion detection on static and dynamic web pages (creating session ID's for each user containing the web front end[HTTP] and back end[SQL server]) also make it able to prevent those intrusions from attacking the web pages and it should be able to find out the perpetrator.

II. LITERATURE SURVEY

Paper Name: New Publicly Verifiable Databases with Efficient Updates
Author Name: X. Chen, J. Li, X. Huang, J. Ma, and W. Lou,
Year: 2015
Summary:
An author has developed a model which notion of verifiable database (VDB) enables a resource-constrained client to securely outsource a very large database to an untrusted server so that it could later retrieve a database record and update it by assigning a new value. Also, any attempt by the server to tamper with the data will be detected by the client. Author proposes a new VDB framework from vector commitment based on the idea of commitment binding. The construction is not only public verifiable but also secure under the FAU attack. Furthermore, he proves that our construction can achieve the desired security properties.

Paper Name: NPP: A New Privacy-Aware Public Auditing Scheme for Cloud Data Sharing with Group Users
Author Name: Anmin Fu, Shui Yu, Yuqing Zhang, Huaqun Wang, Chanying Huang
Year: 2016
Summary:
In this paper, author proposes a new privacy-aware public auditing mechanism for shared cloud data by constructing a homomorphic verifiable group signature. Unlike the existing solutions, our scheme requires at least group managers to recover a trace key cooperatively, which eliminates the abuse of single-authority power and provides non-frameability. Moreover, our scheme ensures that group users can trace data changes through designated binary tree; and can recover the latest correct data block when the current data block is damaged. In addition, the formal security analysis and experimental results indicate that our scheme is provably secure and efficient.

Paper Name: Detecting and Preventing Intrusions In Multi-tier Web Applications
Author Name: Ekta Naik, Ramesh Kagalkar
Year: 2014
Summary:
In this paper, author proposes implemented double guard using IIS(internet information and service manager Furthermore, it quantify the limitations of any multitier IDS in terms of training sessions and functionality coverage. I am implementing the prevention techniques for attacks. I am also finding IP Address of intruder. 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 charac-terize the correct and acceptable static form and dynamic be-havior of the system, which can then be used to detect abnor-mal changes or anomalous behaviour.

Paper Name: A hybrid architecturefor interactive verifiable computation
Author Name: V. Vu, S. Setty, A.J. Blumberg, and M. Walfish
Year: 2013
Summary:
This work is promising but suffers from one of two problems: either it relies on expensive cryptography, or else it applies to a restricted class of computations. Worse, it is not always clear which protocol will perform better for a given problem. He describe a system that (a) extends optimized refinements of the non-cryptographic protocols to a much broader class of computations, (b) uses static analysis to fail over to the cryptographic ones when the non-cryptographic ones would be more expensive, and (c) incorporates this core into a built system that includes a compiler for a high-level language, a distributed server, and GPU acceleration. Experimental results indicate that our system performs better and applies more widely than the best in the literature.

Paper Name: Privacy, security, and trust issues arising from cloud computing.
Author Name: S. Pearson and A. Benameur.
Year: 2010

Summary:
Cloud computing is an emerging paradigm for large scale
infrastructures. It has the advantage of reducing cost by sharing computing and storage resources, combined with an on-demand provisioning mechanism relying on a pay-per-use business model. These new features have a direct impact on the budgeting of IT budgeting but also affect traditional security, trust and privacy mechanisms. Many of these mechanisms are no longer adequate, but need to be rethought to fit this new paradigm. In this paper he assess how security, trust and privacy issues occur in the context of cloud computing and discuss ways in which they may be addressed.

III. ATTACK SCENARIOS

Types of attacks:

3.3.1 Privilege Escalation Attack

Let's assume that the website serves both regular users and administrators. For a regular user, the web request will trigger the set of SQL queries Qu; for an administrator, the request ra will trigger the set of admin level queries Qa. Now suppose that an attacker logs into the webserver as a normal user, upgrades his/her privileges, and triggers admin queries so as to obtain an administrator's data. This attack can never be detected by either the webserver IDS or the database IDS since both ru and Qa are legitimate requests and queries.

3.3.2 Hijack Future Session Attack

This class of attacks is mainly aimed at the webserver side. An attacker usually takes over the webserver 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-Middle attack, an Exfiltration Attack, a Denial-of-Service/Packet Drop attack, or a Replay attack.

3.3.3 Injection Attack

Attacks such as SQL injection do not require compromising the webserver. Attackers can use existing vulnerabilities in the webserver logic to inject the data or string content that contains the exploits and then use the webserver to relay these exploits to attack the back-end database. Since our approach provides a two-tier detection, even if the exploits are accepted by the webserver, the relayed contents to the DB server would not be able to take on the expected structure for the given webserver request. For instance, since the SQL injection attack changes the structure of the SQL queries, even if the injected data were to go through the webserver side, it would generate SQL queries in a different structure that could be detected as a deviation from the SQL query structure that would normally follow such a web request.

IV. PROPOSED SYSTEM

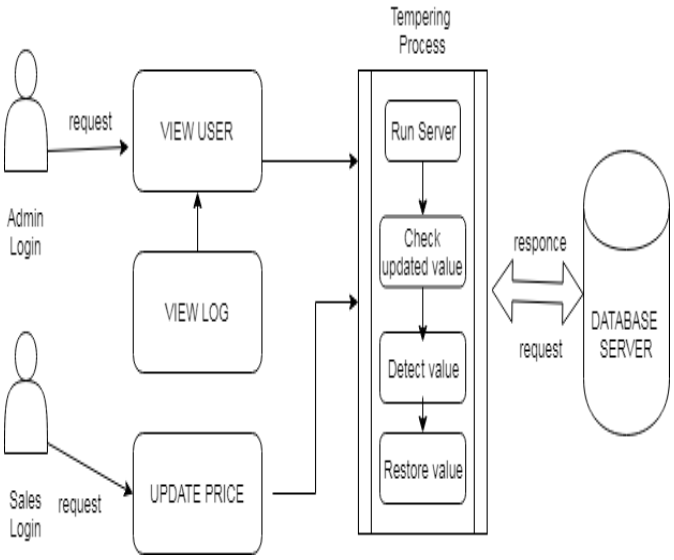

Fig 2. System architecture

Many Systems are providing one way security for the web applications protecting a web application in terms of interface and at database end with proper recovering options is best part of the system The proposed system designs idea in breakdown model to evaluate security of the web applications along with its database in every step.

Module Explanation:

User Module:

User can authorize login access. He can update all personal information. He also can give authority to generated secure encryption process.

Sales Department:

Sales department work as a hacker. Here hacker change the database value of any product without authentication.

Admin Module:
Admin is the authorized person, he check all the user activity records as well as profile. He also watch the tempering on changing the values from data base.

Advantages:
1. The proposed system provides authencation.
2. It also prevents hacking.
4. The system prevents identity theft.

Summary: First of all normally database engines are started and tampering detection is initialized as soon as attack is performed a pop up value is generated at the admin's panel and the data value is restored successfully.

Algorithm: Message Digest 5(MD5)
Input: Input data D = D1, D2, D3,.., Dn saves into the hash table.

Step 1: Arrange all input data into matrix format (save into log files).

Step 2: Consider m as a selected data act as a new selected data.

Step 3: m position gets changed after allocated time period.

Step 4: If () data get hacked.

Step 5: Data leakage is occurs.

Step 6: We have to check the leakage data and prevent

Step 7: Using Revert back function we have to get original data.

Step 8: When user calls that corrupted file, hash function gives to user a previous data.

Step 9: Return True.

V. SYSTEM ANALYSIS

We implement these system for avoiding the network security threads occurring when people online transaction. We analysis this system using the following points: In this paper we use the following algorithm for implementing the secure system.

VI. MATHEMATICAL MODEL

System Description:
Input:

Function DATABASE INTRUSION DETECTION ()
Set V:
V0=Get the time in seconds (T)
V1=Visit Database table for reach interval of T
V2=Get a record from the database
V3=Hash it using MD5 Algorithm
V4=Create vector of hash values
V5=Send to Notarize

Output:

VALIDATOR: (Here this module is responsible for periodically scans the audited tables, computing the hash values on a per transaction basis

Success Conditions: Success system when do not change any value from database.
Failure Conditions: Our system fails when attacker get success form data base insertion.

VII. CONCLUSION

We propose a tampering detection system, which constructs the model of normal behaviour for multitier web applications from in co-operation the front end web (HTTP) requests and back end DB (SQL) queries.

VIII. ACKNOWLEDGMENT

I wish to express my profound thanks to all who helped us directly or indirectly in making this paper. Finally I wish to thank to all our friends and well-wishers who supported us in completing this paper successfully I am especially grateful to our guide Prof. Prof. H.A.Hingoliwala for him time to time, very much needed, valuable guidance. Without the full support and cheerful encouragement of my guide, the paper would not have been completed on time.

REFERENCE

[1]X. Chen, J. Li, X. Huang, J. Ma, and W. Lou,New Publicly Verifiable Databases with Efficient Updates, IEEE Transactions on Dependable and Secure Computing, In press, 2015.

[2] Anmin Fu, Shui Yu, Yuqing Zhang, Huaqun Wang, Chanying Huang, "A New Privacy-Aware Public Auditing Scheme for Cloud Data Sharing with Group Users" IEEE, 2016.

[3] Ekta Naik, Ramesh Kagalkar, "Detecting and Preventing Intrusions In Multi-tier Web Applications", International Journal of Scientific & Engineering Research, Volume 5, Issue 12, December-2014.

[4] V. Vu, S. Setty, A.J. Blumberg, and M. Walfish,A hybrid architecturefor interactive verifiable computation, IEEE Symposium on Securityand Privacy (SP), pp.223-237, IEEE, 2013.

[5] S. Pearson and A. Benameur. "Privacy, security, and trust issues arising from cloud computing." Proc. Cloud Computing and Science, pp. 693–702, 2010

