International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 06, 2020
ISSN: 1475-7192

DEMERITS, DETECTION & PREVENTION OF SQL INJECTION ATTACKS OVER THE CLOUD COMPUTING

Smita Chavan[1], Dr.Sharvaree Tamane[2]

Abstract: *Web services that continuously deliver services to customers are basically connected to the backend database which contains highly sensitive information. As demand of deploying such applications increases, it also increases the possibility of such attacks that target applications. SQLIA is the most popular security attacks in the web application system. This type of attack is caused due to lacking of SQL parameters used and input validation. Some popular SQL injection attack that can affect the system and its prevention techniques are mentioned in this paper. Today's internet world, securing data on cloud is very important issue. One of the most important challenges to secure web application is acknowledged by SQL injection attack. Most sensitive SaaS vulnerability which allows attacker to break the integrity and confidentiality of user's data is called SQL injection attack. It breaches the security policy. Attacker inserts some code in the query which is not written by application developer. SQL injection is also called as web security vulnerability. Mostly it permits attacker to see data that they are not able to see. This paper proposes simulation of six case studies of SQL injection attack. System presents SQL injection attack with protection and without protection. System shows test case with protection means by specifying rules. If rule specification done then ontology logic is used. It uses test case without protection means creation of unknown user name or history of user. Implementation of system also classifies how attack happens, detection of attacks without protection and with protection.*

Keywords: *SQL injection; Cloud computing; Input validation; Cloud security; Deployment models.*

I. INTRODUCTION

Cloud computing in today's era is a most demand, self-service which spread over a large network access that helps in visualizing the computing that is one step ahead. It is related to the grid, parallel and computing that is distributed depending on the internet that deploys all the data sets so as the application is provided with all the resources required such as platform, hardware, data, and software [1]. The storage and processing technologies are getting cheaper day by day as new techniques strikes the market. But the loopholes are been created when a third party is involved over the internet for many of the unreliable string operations. The cloud helps in storing large private and public data which seems to be cheaper and safe as compared to other means. The first cloud service was opened by Amazon known as AWS (Amazon Web Service) in 2006. Various privacy checks must be applied to the stored data in the cloud database as today almost everyone is using cloud services. Suitable methods must be applied for the cloud services to overcome the obstacles and challenges [2].

Received: 22 Dec 2019 | Revised: 13 Jan 2020 | Accepted: 05 Feb 2020

[1] *Assistant Professor, Information Technology Department, Government Engineering College, Aurangabad, India.*
[2] *Professor, Information Technology Department, MGM's Jawaharlal Nehru Engineering College, Aurangabad, India*

The services that help to connect the customers with the database having most precise information are provided by web-based servers. As this rate of deploying such applications increases it also increases the rate of number of attacks affecting these applications. Under an observation study it was observed that 80% of such cyber-crime attacks are conducted on the most sensible layer that is application layer. SQLIA are known for the mostly used frequent security threat that affects the web-based applications [3]. SQLIA launches a sensitive query which leads in the direction of destroying of the server connected the application and also helps attacker in gaining unauthorized access to the system which further rights the access of deleting, retrieving and modifying the confidential information from the database [4].

II. PLATFORM FOR CLOUD

This section states that there are four platforms used to fulfill the requisite and possibility of technology related to the cloud computing. The organization can choose any of the platforms from the available ones as per their needs. Every organization can have different needs. When the SQL queries are injected, it harms the client server database. The cloud types are as follow [5].

Private Cloud: This type of cloud services have private infrastructure with the organization members and is not shared with any other organization. This makes it more expensive but also it is much secure and reliable than public cloud. E.g. Oracle, IBM, Dell EMC.

Public Cloud: The cloud vendor hosts such type of cloud infrastructure which can be shared by number of organizations based on the vendor premises. E.g. ESDS eNlight Cloud, IBM Blue Cloud, Amazon EC2.

Community Cloud: This type of cloud infrastructure within the cloud community where more than one public or private organization can use it. E.g. Group of colleges/schools that falls beneath a particular university.

Hybrid Cloud: Both public and private cloud together forms hybrid cloud. Organizations can host the important applications on private cloud while less secured concern on public cloud.

III. CLOUD COMPUTING FORMATION

This part describes the two most important concepts that are "cloud" and "'computing". When it comes to "cloud", one is concerned to internet, but it becomes more complicated when it comes to "computing". Cloud computing is a way which helps organizations to choose whatever cloud services they want and pay only for the services used, which helps to reduce the company's expenditure and switch instantly to grow and lower as per the community requirements. Mainly cloud components are applications, cloud based services, cloud client, platform and database. Cloud computing also includes virtual servers, software, hardware, desktops etc. which helps the organization to rely and use whichever service they want [5].

The content to be stored on the cloud database is authorized through several checkpoints. The security of the database is ensured by these checkpoints. These checkpoints can be harmed by the SQLIA at any of the service(SaaS, IaaS, PaaS, DaaS, EaaS).The technology used provides wide access of network which uses resource sharing among several users, which results in reduction of overhead and availability of resources whenever needed so as to ensure data security [6].

Received: 22 Dec 2019 | Revised: 13 Jan 2020 | Accepted: 05 Feb 2020

IV. MOTIVATION

The SQLIAs queries are injected which attacks the client database. It does not matter that the data is flowing through internet or is stored in the database. The queries affect and harm the data which eventually results in accessing the data to the attacker. The server system configuration can be modified by the attacker and can also create a fake platform which may lead to the server crash which contains the confidential data. It is somewhat easier to recognize and get over SQLIAs attack on the DaaS level but it is more important to find the solutions on the SaaS, PaaS, IaaS and EaaS levels too. The solutions found must be almost 80% effective to avoid the SQLIAs attack.

V. CLOUD MODELS & SQLIA PREVENTION METHOD

Cloud computing is essential for real world as it is internet-based computing, where organizations are provided several services such as servers, data services etc. Several devices such as PC, laptop, notebook, mobile phones etc. can be used for the communication with this internet-based cloud. The services provided are actually divided into three layers mainly known as software, platform and infrastructure. But this all layers are counted into a single phase to provide better services to the user. The attacker is trying to harm, hack or spoof any of the services provided to the user. So as to ensure the security and detect the hacks these layers are separated from each other [7].

Platform as a service (PaaS): Platform for computing is provided by this layer which majorly involves operating system, programming language etc. Users can create, edit, and modify their applications using this type of platforms. E.g. Google App Engine, Heroku, Alibaba Cloud E-HPC, OpenShift etc.

Software as a service (SaaS): The application which can be used from anywhere and anytime by making use of internet is provided by this layer. It provides features such as encryption, cryptography. One can make use of this software anytime. e.g. Google Apps, Dropbox, ZenDesk, Hubspot etc.

Infrastructure as a service (IaaS): infrastructure and hardware is provided on rent and lease for the users for whatever time they want under IaaS. Also this layer is referred to as hardware as a service. E.g. Amazon Web Services (AWS), Linode etc. The first three layers were invented in sixties and are the most basic layers and afterwards more two layers were discovered on the basis of research and studies [8].

Data as a Service (DaaS): The data over the internet is stored in a unmanaged or meaningless way, which has to be managed by applying some sorting methods to it. Therefore this models works over bulky data retrieving which ensures the availability and privacy which eventually results in efficiency and concurrency in the data storage. This method is cost effective and gives good quality data. E.g. Urban Mapping, VMware etc.

Education as a service (EaaS): EaaS provides some modern education-oriented services such as smart classes and e-learning. This model provides distant learning which helps the users to gain the knowledge and access the services from the remote locations. E.g. Indiamart, Educomp etc. This layer follows some security protocols which are strong enough to maintain the breakthrough possibility and helps in stopping the attacks that affect the system [9].

While implementing the software testing some precautions are taken against incorrect and malicious input and user not only need to be validated using parameters as testing length, format, type, and range but also some aspects such as architecture and deployment criteria of application must be taken into the consideration [10]. The prevention is

properly explained below with figure 1. The items that need to be validated are as below:

1. No any assumptions should be made about size, type or any of the contents related to the data which application receives.

2. The input should be verified on the basis of range and variable type and proper restrictions should be enforced.

3. User must be validated by using the stored procedures.

4. The user input should not be concatenated, as it is not validated.

5. String variable contents must be checked.

6. Transact-SQL statements should not be built directly from the user.

VI. SQLIA MODEL

To get rid of the SQL infection attacks a defense model has been proposed which helps in preventing the SQL injection attacks. The model is a multi-tasking model. This model is used to validate input and also specify information based on web address, which is relatively important too, especially for detecting the sensible characters. IP address reliability gets verified by the server side. User gets rejected from logging in if the input values are found unreal or illegal. After IP address verification input values are tested by the server side using various parameters such as length, format, range, type. If the input string matches with that of SQL rules then only the user is allowed to access the web page. After that the server side verifies the privilege of the user. If access permissions gets exceed for a particular user, results in blocking of user and message is been send to the system administrator by the system. When all the verifications performed by the server side are incorrect then at such situation the injection attack is recorded by the server [8].

VII. RISK MODEL

There can be several risks when attacker initiates SQL queries. One must be aware of the risks and harms that can be caused [7]. Some of them are mentioned below

1.Sensitive information such as secret numbers, security numbers or credit card details can be extracted and gained by the hacker.

2.Original data can be deleted and tables can be dropped which eventually can result to corruption of the database, and can make the website unusable.

3.When users visit the website, some malicious code is further added to the code being currently executed which can give access of the data to the attacker.

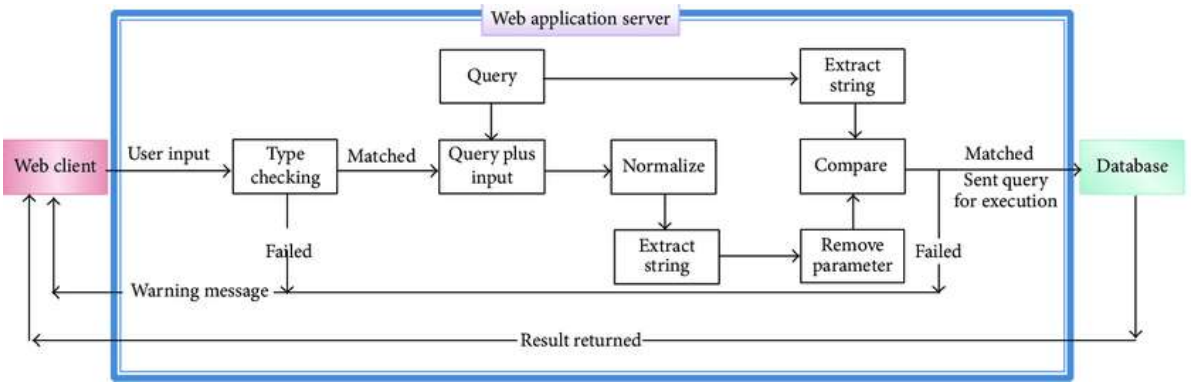

Figure 1. SQLIA Prevention Method

The figure 1 explains the detailed view of preventing the SQLIA. The SQLIA must be prevented so as to ensure privacy and reduce risks. There are several methods to overcome the SQLIA, above figure explains one of the method. The web client is connected to the database with the help of web application server, which has several components within it which helps to protect the SQLIA. The web client provides user input which is type checked and if it does not matched then the operation is failed and message is send to the web client regarding login failed. If the input string is matched then a query is added to the input which is further normalized for the extracting purpose. Now string containing information is extracted from this normalized query so as to gain appropriate data. Parameters are removed from the extracted string and original query which was added to the input is also extracted into a string. Now these both queries are compared with each other and if they match with each other than query is processed for further execution to the database [11]. Now the result is forwarded to the web client. If the queries don't match with each other than the operation is rejected and further operations are terminated and web client is not able to access the database. Web client has to start newly when gets rejected at any of the stage in the model here [12].

VIII. Attacker Injecting SQL Queries

The figure 2 completely describes how the attacker adds SQL queries so as to gain access or to destroy the data stored in the database. The attacker firstly gathers the information that is stored into the cloud database. Then the attacker can access the applications that are been stored to the cloud database. Till this stage the attacker acts as a non-attacker so as to bypass or to avoid rejection to the web application. Now the attacker injects the SQL queries which are harmful and by which the data extraction can be done. As soon the password hash is generated, the password hacked by the hacker and gains access to the whole system or we can say that now the attacker has control over the server. The password hash cracking fools the system by showing attacker as an admin and gaining him/her access to the server where the hacker again injects PHP backdoor queries. Now through the server the attacker gets the privilege to the server root which contains the sensitive data such as secret numbers, confidential data, credit card details etc. which can now easily be accessed or destroyed by the hacker [13].

Received: 22 Dec 2019 | Revised: 13 Jan 2020 | Accepted: 05 Feb 2020

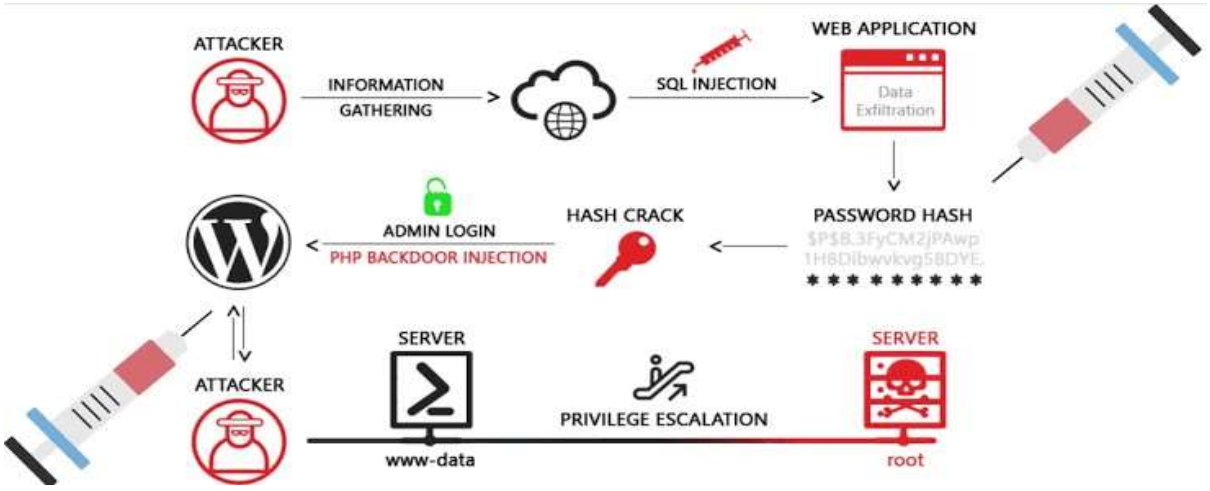

Figure 2. Attacker Injecting SQL Queries

Existing Applications as a Solution to SQLIA

SQLIA is not a new problem, but the organizations are facing this problem from an uncertain time. SQLIA has infected e-commerce the most till now. This problem however is on a high rise since past few years. Many companies claim that they have application that has a prime solution of this attack, but not every company has that type of application. Some of the applications are listed below:

1. Mod Security: It is an open source application which makes use of detection engine foe applications that provides tips and helps in detecting SQL injections. Mod Security is developed for java which runs servlet 2.3 and keeps application and browser separated.

2. Airlock: Airlock helps in combining private reverse proxy with invasion prevention, filtering content, and verifying user compulsion.

3. Green SQL: Green SQL separates web server from database server, which prevents the database from SQL injection attacks. SQL commands can create a risk of blocking the database (e.g. CREATE, ALTER, DROP) is done by Green SQL.

4. Dot Defender: One type of firewall which protects web application is dot defender which provides solutions to the SQL injection attacks. This application runs on Apache & IIS web servers which makes it a multi-platform solution.

5. Static Analysis: Detection and prevention of SQLIA is done at compilation time under static analysis. This approach is fast in its own nature but it also has a big disadvantage that this approach is not able to catch illegal, shoulder attacks and SQL injection attacks on the stored procedures.

6. Dynamic Analysis: The queries are detected at compilation time using static analysis, so foremost queries can be passed to the web application by the attacker. But dynamic analysis helps to detect and prevent, queries injected at the run time.

7. Secure Sphere: This software uses advanced deviation detection, correlation of events, and set of signature directories so as the database remains protected and also web applications are not affected.

Received: 22 Dec 2019 | Revised: 13 Jan 2020 | Accepted: 05 Feb 2020

IX. ATTACKER INJECTING SQL QUERIES

In today's era many of the web applications are in a huge scope of development phase and that too in extremely short time, which makes its very difficult to eliminate many of the security vulnerabilities such as cross-site scripting and SQL injection [10]. Some of the tips to be secured from SQL injection are as follows:

1. Before transmitting the user input to the server, it must be validated.
2. Only specific privileged accounts must be permitted transmit the server input to the server.
3. Few necessary privileges must be run on SQL server.

There are three types of SQL which are as follows:

1. Blind SQL Injection.
2. Error message based SQL.
3. Redirecting and reshaping a query.

These methods are explained briefly below.

1. Tautologies: Tautology-based attacks are known for injection one or more conditional SQL statement. This attacks act in such a way that it makes SQL command rate seems to be a real condition (i.e. '1=1'). This technique is mostly used for the web page authentication bypass to get access over the database.

2. Piggy-based Query: In such type of attacks, the attacker adds some extra queries to the earlier existing query string. When attack succeeds, the attacker gets the access to the database and now can execute a query string which may contain different types of multiple queries [4]. In such type of attacks the early query is said as original query while the additional queries are called as infected queries.

3. Logically Incorrect: Use of error messages is done in this type of approach by the attacker which are displayed or flashed on the screen by database for a wrong query. These error messages support the attacker to get the important information.

4. Union query: The attacker adds additional statement to the earlier SQL string in such type of approach. In such approach attacker adds some UNION query string or a SQL statement. This output gained by this type of attack is database which is addition of the result of the main query and the output of the inserted query.

5. Stored Procedure: The attacker aims on the procedures that are stored in the database in this type of attack. This is a type of shoulder surfing. Stored procedure is one type of code which can be vulnerable as a program code. Stored procedure is used to give true or false for legal or illegal user.

6. Blind injection: Details regarding error are protected by the developers, so as to ensure the safety of the user database from the attacker. At such situation the attacker has to go through a universal page which is made compulsion for the user to enter, by developer which excluding the message that contains errors. Still an attacker asks multiple strings of two way questions i.e. true or false type through the SQL codes which helps the attacker for hacking the database.

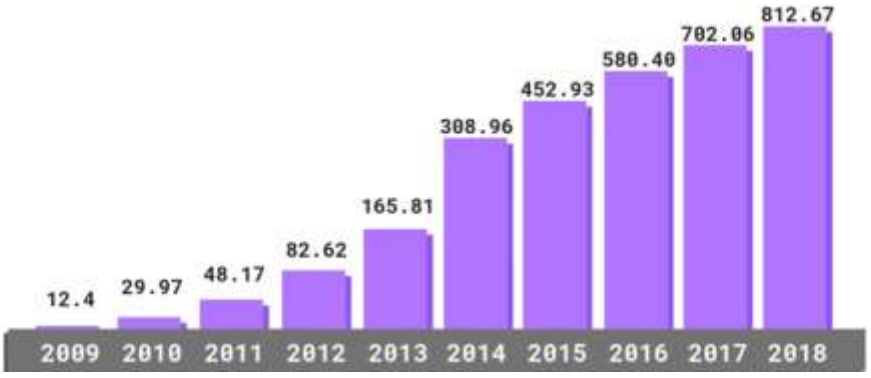

Figure 3. Increasing rate of SQLIA since last 10 years

X. RESULT

Test case 1 (without protection)

USERNAME xie' or '1'--'

PASSWORD No Password

User Name: xie User Pass: darren1 Type: Account Balance: 50000

Test case 2 (without protection)(password is the same to user name)

USERNAME 1' or '1'='1

PASSWORD 1' or '1'='1

User Name: xie User Pass: darren1 Type: Account Balance: 50000

User Name: ram User Pass: ram1 Type: Account Balance: 40000

User Name: sham User Pass: sham1 Type: Account Balance: 30000

User Name: kiran User Pass: kiran1 Type: Account Balance: 20000

Test case 3 (without protection) (Common user login)

USERNAME xie

PASSWORD darren1

User Name: xie User Pass: darren1 Type: Account Balance: 50000

User Name: ram User Pass: ram1 Type: Account Balance: 40000

User Name: sham User Pass: sham1 Type: Account Balance: 30000

User Name: kiran User Pass: kiran1 Type: Account Balance: 20000

The above test cases are same for with and without protections only the conditions are different. Applied same test case with protection and checked the results.

XI. CONCLUSION

The SQL injection approach this paper is typically a runtime one where attacker tries to inject the SQL queries so as to get access into the database where he/she can modify, copy, delete data sources. The approach ensures that no any vulnerable code is executed which can affect the system or the OS and devices, partly or totally. This SQL injection approach is applies on server side database which is further related with the cloud environment which is distributed,

which provides a clear system which ensures the security of executing of all the queries that are requested without hacking any database. Proxy server is proposed over the cloud Data Service Provider which almost reduces the half of the attacks. For excluding the another half attacks security tool is launched for the proxy server which helps comparison original queries using some of the rules that are prescribed by the security tool earlier which helps in filtering all of the affected queries and prevents the firewall from getting bind.

REFERENCE:

[1] Kim, Mi-Yeon; Lee, Dong Hoon.Data-mining based SQL injection attack detection using internal query trees.[J]Expert systems with applications. 2013,9:416-430

[2] Anley C. Advanced SQL injection SQL sever application.[EB]. http://www.creangel.com/papers/advanced sq I_injection. pdf.

[3] Mittal, Piyush. A fast and secure way to prevent SQL injection attacks.[Cl2013 IEEE Conference on Information and Communication Technologies, ICT 2013, P 730-734

[4] Meng Ting. MySQL Injection Attacks and Defense Methods. [J]Information Security and Technology,2013.11 :26-38

[5] Jang, Young-Su; Choi, Jin-Young,Detecting SQL injection attacks using query result size[J]COMPUTERS & SECURITY,2014.44: 104-118

[6] Mishra, Nitin; Chaturvedi, Saumya; Sharma, Anil Kumar.XML-Based Authentication to Handle SQL Injection.[J]Advances in Intelligent Systems and Computing.2014,236.: 739-749

[7] "Implement of cloud computing for e-Learning system", Manop phankokruad,2012 International Conference on Computer & Information Science (ICCIS), pp. 7-11

[8] 2. Extended results on privacy against coalitions of users in user-private information retrieval protocols. Colleen M. Swanson, Douglas R. Stinson. 4, s.l. : Springer, February 12 , 2015,

Cryptography and Communications, Vol. 7, pp. 415-437.

[9] 3. Global sensitivity measures from given data. Elmar Plischkea, Emanuele Borgonovob, Curtis L. Smithc. 3, s.l. : elsevier, may 1, 2013, European Journal of Operational Research, Vol. 226, pp. 536-550. 10.1016/j.ejor.2012.11.047.

[10] 4. Cache Serializability: Reducing Inconsistency in Edge Transactions. Eyal, I., Birman, K. and van Renesse, R. columbus, OH : IEEE, june-july 29-2, 2015, 2015 IEEE 35th International Conference on Distributed Computing Systems (ICDCS), pp. 686-695. 10.1109/ICDCS.2015.75.

[11] 5. Combining Static Analysis and Runtime Monitoring to Counter SQL-Injection Attacks. W. Halfond, A. Orso. s.l. : IEEE, Proceeding of the Third International ICSE Workshop on Dynamic Analysis .

[12] 6. Detection and Prevention of SQL Injection Attacks. Halfond, William G.J. and Orso, Alessandro. s.l. : Springer, 2007, pp. 85-109.

[13] 7. CANDID: Preventing SQL Injection Attacks using Dynamic Candidate Evaluations. Bandhakavi, Sruthi, et al., et al. Alexandria, Virginia, USA : ACM, October-November 29-2, 2007.

[14] 8. Privacy-enhanced architecture for smart metering. Félix Gómez Mármol, Christoph Sorge, Ronald Petrlic, Osman Ugus, Dirk Westhoff, Gregorio Martínez Pérez. 2, s.l. : Springer, november 28, 2012, International Journal of Information Security, Vol. 12, pp. 67-82. 10.1007/s10207-012-0181-6.