International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 2 | ISSN : 2456-3307

Analysis on Database Security Model Against NOSQL Injection

S. Priyadharshini*, R. Rajmohan

Computer science and engineering, Anna University/IFET College of Engineering, Villupuram, Tamil Nadu, India

ABSTRACT

Nowadays, Attackers analyse the NOSQL data structure and inject malicious code as well as perform cross-site request forgery attacks. Study a Database Protection System which is used between the dynamic application and database. The Data centric security model is used for encrypting data before storing into database repository. Mobile users across an untrusted network are authenticated through Kerberos. The testing on NOSQL injections performed with JavaScript and PHP is studied.

Keywords: NOSQL, Mongo DB, Security, Kerberos.

I. INTRODUCTION

The organizations moved into non-relational databases in the late 90's. It does not support Relational database. NOSQL databases are also a non-relational databases and it means "Not Only SQL". The NOSQL has a different data storage models. It has no structured data. The NOSQL data models are document, graph, key-value and column wide. Security may be difficult and no strong inconsistency. . NOSQL databases such as Cassandra, Mongo DB, CouchDB, Redis and HBase. Few techniques need to mitigate the attacks on NOSQL Databases. This paper examines the maturity of security measures for Mongo DB, a typical NOSQL Database system, with aspects in both attacks and defence at the code level.

II. METHODS AND MATERIAL

A. Related Work

In database security, there is an attack in non-relational databases. In addition, design and implements a NOSQL database called SensibleThingsNOSQL for IOT [1]. Without wasting the storage, this query execution is implementing in proposed method. STNOSQL grant load balance to reform the reliability and scalability during implementation between different methods. Based on the symmetric encryption, this paper design security mechanisms with access control [1].Database provides the two approaches. There are data management and real time web applications. It is distinguished data storage. It supports the flexibility and availability. NOSQL database need not structured format. NOSQL are convenient than relational database, because performance and real time access are more correlated than concord such as indexing and retrieving huge amount of data. In addition, business data are stored in cloud. It is easily and quickly accesses the data, it has been security issue of non-relational databases .At the code level, both attacks and defence examines the maturity of security measures for mongo database. These injections are performing in JavaScript and PHP. JavaScript inject attack reveals the private data's of customer and preventing the security problems. For avoiding malicious code, the security layers are build [2].In NOSQL database; it has flexible structure and support available, soft state and eventually consistent properties. NOSQL Database supported by sharding. Storing data records across multiple machines are processed. Sharded are latency sensitive and it has no memory. It is faster to read and write in database. Secured the Sharded data has various servers to distributed and transmitted over unsecured network .So there is a challenging problems. For the evaluation of various open source and Sharded NOSQL database, it analyses and identified lacking of security [3].Non-relational data storage system has very popular to scalability [2]. Addressing new queries and access mechanisms. NOSQL database has few techniques for attacking such as Injections and CSRF. Moreover, analyse the

methodologies to alone the attacks. Security measures and awareness are enhanced in NOSQL system otherwise it suffer from the security risks [4].The distributed web applications and cloud computing has created to store huge set of data. NOSQL provides high availability and scalability [2]. Nowadays a growing companies are adopted various types of non-relational database, it referred to as NOSQL database. It does not support SQL functionality. The most popular NOSQL database are Mongo database and Cassandra are outlined their security problems and features [5].

B. NO SQL ATTACK MECHANISM

Fig.1 depicts the model of NoSql attack in databases.
In NOSQL database, it uses different query language, which makes SQL injection techniques irrelevant. NOSQL systems are immune to injections. The query language's security and drivers has largely improved techniques for injecting malicious queries. Reports of NOSQL injection techniques are provided. The emerged has initial application-scanning projects and the Open Web Application Security Project has published for testing NOSQL injection code. In NOSQL attacks vectors, NOSQL databases to store customer data in web applications and its services. NOSQL are used to store the data accessed.

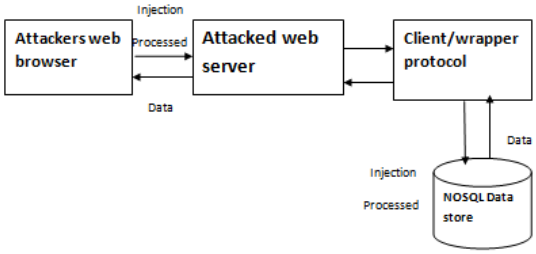

Figure 1 NOSQL Attack

C. NOSQL INJECTION VULNERABILITIES

NoSql injection attack is experienced in different area of an application. NOSQL database has a less consistency while comparing traditional databases [6].

1. NOSQL injection vulnerabilities in MongoDB:

Using FIND query [7],

- ✓ Within SQL, operators are used in filter and check: db.myCollection.find ({$where:"this. credit==this. debit"});
- ✓ To allow more condition in JavaScript: db.myCollection.find ({$where: func (){ return obj.credit- obj.debit<0,}});
- ✓ Without sanitization in mongodb query: db.myCollection.find {(active: 1, $where: func () {return obj.credit- obj.debit<$userInput ;}});
- ✓ In mongodb, replacing the operator with malicious data: db.myCollection.find ({$where: func (){ return obj.credit- obj.debit<0,}});

D. SECURITY MEASURES

Comparing the two different databases

1. MONGODB STATUS:

To configure Mongodb for Kerberos support and authenticate,

- ✓ Configure mongodb with Kerberos authentication on Linux and
- ✓ Configure mongodb with Kerberos authentication on windows

Table 1. MongoDB status

Category	Status	Recommendations
Data	No encryption	Encryption is critical
Authentication	Unsharded configuration	Supports proxy authentication
Authorization	Unsharded configuration	To generate complete authentication
Auditing	Not available	Support auditing and generate audit events
Injection attacks	JavaScript	Malicious user cannot modify the code

2. CASSANDRA STATUS

In Kerberos, Sqoop can use for user authentication. In addition, to generate the authorization after complete the authentication.

Table 2. Cassandra Status

Category	Status	Recommendations
Data	No encryption	Possible in Encryption and Key management platform
Authentication	Not available	Sqoop can be used for authentication in Kerberos
Authorization	Not available	Implement authenticate and authority
Auditing	Not available	Enable audit logging
Injection attacks	CQL	PHP

E. NOSQL ATTACK IN MONGODB

The NOSQL attacks are injecting code when the inputs are not sanitized and the solution is to sanitize them [8].

- Using mongo dB and node.js:

```
app.post (‘/user’, func (register, res) {
var query = {
Usrname: request.body.usrname,
Pasword: request.body.pasword
}
```

- To receive the following request:

```
POST http://www.example.com / user HTTP /1.1
Content-Type: application/JSON
{
“Usrname”: {“$ne”: null},
“Pasword”: {“$ne”: null}
}
```

- To sanitize the input:

```
Var sanitize =require (‘mongo sanitize’);
App.post (‘/user’, func (register, res) {
var query = {
Username: sanitize (req.body.username),
Password: sanitize (req.body.password)
}
db.Collection (‘user’).find (query, function (err, user)
{
Console.log (user);
});
});
```

- To attack the $where operator :

$operator has a very dangerous and it allows you to pass a string .Then it will be evaluated inside your server.

```
var query = {
$where. “this.cancelorder>”request.body.cancelorders}

db.Collection (‘user’).find (query) each (func (error, document)
{Console.log (doc);
});
```

- If it won’t work, it gives error :

```
Error: error :{
“$err:” reference Error: threshold is not defined\n at_func2 (_funcs2:1:45) near ‘s. cancelorders > threshold}’ “,
“code”:16722
}
```

F. NOSQL ATTACK IN CASSANDRA

Cassandra database is used to prevent the attacks. It resistant to injection contains:

- CQL query must contain one statement
- There are not statement types that contain other statements, and it would be another common vector for an injection using rule of thumb.

Without encryption, Cassandra’s data files are stored. The database does not have an automatic data encryption, so hackers can an access the data.

To communicate with the client, Cassandra utilizes Apache Thrift framework. The current version of Cassandra supports the Secure Socket Layer. Secure Socket Layer cannot be enabled by default. Communication between the client and server becomes no encryption. In new version, the Cassandra supports the encryption. This DB manage the CQL. Cassandra performs one thread per one client, so there is a problem with denial attacks.

III. RESULTS AND DISCUSSION

The proposed architecture is depicted in fig.2. Kerberos is an authentication protocol for securing the network. It provides the authentication service and mutual authentication between user and server. It has distributed client-server architecture. The Kerberos has additional features like: i) Auditing ii) Data at rest and iii) Injection attacks.

Kerberos protocol builds on symmetric key cryptography. It requires a trusted third party, and it may use cryptography during authentication.

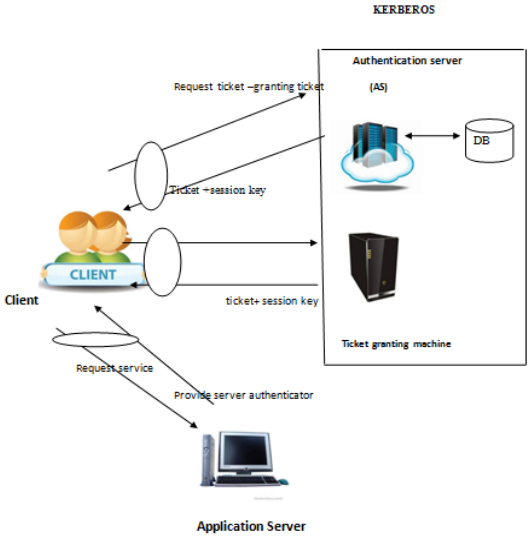

Fig.2 Proposed architecture

The Kerberos requirements are scalable, secure, reliable and transparent. It is a solution for network security problems. This protocol used the strong cryptography. Clients can identity to a server across an insecure network connection.

ALOGRITHM:

Step 1: Symmetric key cryptography is used to authentication Kerberos
Step 2: Kerberos Key Distribution Centre provides scalability.
Step 3: Secure transport of a session key can provided by Kerberos ticket.
Step 4: The session key is distribute by Kerberos KDC send to its client.
Step 5: The use of the entities master keys can be limited by the Kerberos ticket granting Ticket.

IV. CONCLUSION

The Kerberos is designed to address critical security flaws in NOSQL and to validate the Data Centric security model for encrypting the data. This model helps to design and enhance appropriate security mechanism for securing NOSQL databases. To the existing network security problems, an accurate solution would be Kerberos mechanism. It provides the tools of strong cryptography over the network to help you secure your information systems across your entire enterprise. Kerberos can be extended to provide auditing service, thus securing the NOSQL database.

V. REFERENCES

[1]. Suna Yin, Dehua Chen, Jiajin Le,China, 2016 IEEE,"STNOSQL Creating NOSQL Database on the SensibleThings Platform.

[2]. Boyu Hou, Kai Qian, Lei Li, Yong Shi, Lixin Tao, Jigang Liu, USA, 2016 IEEE 3rd International Conference on Cyber Security and Cloud Computing ,"Mongo Database NOSQL Injection Analysis and Detection".

[3]. Anam Zahid, Rahat Masood, Muhammad Awais Shibli, 2014 Conference on Information Assurance and Cyber Security (CIACS)," Security of Sharded NOSQL Databases"

[4]. Aviv Ron, Alexandra Shulman-Peleg, Emanuel Bornstein, "NO SQL, NO Injection-Examining NOSQL".

[5]. Loir Okman, Nurit Gal-Oz, Yaron Gonen, Ehud Gudes, Jenny Abramov, 2011 International Joint Conference of IEEE TrustCom-11/IEEE-11/FCST-11, "Security issues in NOSQL Databases ".

[6]. Preecha Noiumkar and Tawatchai Chomsiri "A Comparison the Level of Security on Top 5 Open Source NOSQL Databases".

[7]. NOSQL Injection for Mongodb: https://github.com/cr0hn/nosqlinjection_wordlists

[8]. Injection attacks on Mongodb: https://www.quora.com/Why-are-injection-attacks-not-possible-on-MongoDB