BASIC NOSQL INJECTION ANALYSIS AND DETECTION ON MONGODB

Vrinda Sachdeva
Department of CSE
MVN University
Palwal., India
Vrinda08@gmail.com

Sachin Gupta
Department of CSE
MVN University
Palwal, India
achin.gupta@mvn.edu.in

***Abstract* -- In recent years the usage of Information technology has unexpectedly increasing resulting in huge data generation. Many companies have taken initiative to use no relational database for managing the data .It has powerful function to manage big data. Huge amount of data has been generated every day. It possibly results in malicious information into the database. No sql database gaining popularity due to its powerful features like scalability, flexibility, faster data access and availability. In this paper, we will analyze the injection on NOSQL database .We also propose defense method by using php and java script. MongoDB is one of the most secure and powerful no sql database .In this paper we demonstrate, basic no sql injection attack and propose defense method to secure the no sql database .In this way, no sql database programmer be aware of the basic no sql injection attack mechanism and create a more secure database to store huge data.**

Keywords: nosql, mongo DB, injection, attack, Vulnerability, defense.

I. INTRODUCTION

In today's world generation of data is at very rapid speed and there is requirement to check every piece of data to maintain secure environment [2] .Security of database is a challenging task for companies. As the database become complex, the more complex security measures have to be applied. However, monitoring every piece of data is very expensive for time as well as for money. Although it slowdowns the data transaction. There is one solution to resolve this problem is to check the input before entering into the database [2].Malicious attacks are mostly done by input box. When attacker get access to database then he has the dangerous amount of control over the database, even he can deface the database. Injection is one of the hacking methods that is used by the attacker on traditional sql database. For example: sql injection can completely destroy the database and generally it is used for data driven application. Sql injection usually occurs on input box.

But now a days generation of data is occurring at a very rapid speed. So mostly companies migrating their database from sql to nosql database. Although injection can also be executed in nosql database .There are various method to implement injection in nosql database .NOSQL database usually offer availability ,horizontal scaling, performance, partition tolerance. Examples of NOSQL database are MongoDB, Cassandra, redis. MongoDB can accommodate to individual as well as to all size of companies. NOSQL database does not use fixed schema. In this user can add data at anytime and anywhere. Since there is no fixed schema of data although user can add new attributes. More attention is paid to real time data processing application in NOSQL database. However, nosql does not use sql language [1], it uses JSON query. Still it is immune to the threat of injection attacks. Principle of nosql injection is exactly same as of sql injection, only the change in the grammar form of injection. JSON injection is the threat for nosql database .So, our study show that MongoDB is still vulnerable to JSON injection attacks. Even they offer the looser consistency restrictions, horizontals calling. In this paper, we will discuss Basic NOSQL injection using php script and java script and we will also provide the defending solution.

II. RELATED WORK

NOSQL database provide looser consistency restrictions than relational database [1].Researchers have done lots of work on data security. NOSQL database offer performance advantage, horizontal scaling and partition tolerance benefits. Mostly NOSQL injection attacks tries to inject code at the input box. So, the solution is to sanitize the input before using it.

Hackers try to inject malicious code either into input box or URL using GET or POST method. It can be injected by using PHP and Java script code [3].Input box are generally used in query which means that parameter of input method are used to store the value and participate in execution.

III. NOSQL INJECTION

In today's world, generation of data is occurring at a very rapid speed. To manage the data and its security is very challenging task. Critical information such as online reservation, electronic banking service, online shopping making it vulnerable to attacks from the hackers[2].If some data leaks then whole organization would suffer enormous loss, then there will be need to spend more time on the database security.

Now a day's mostly companies migrating their database

978-1-5386-5367-8/18/$31.00 ©2018 IEEE

from sql to NOSQL .Method of attack on NOSQL database are similar as SQL database, only the grammar form is change. It produce a negative impact on the information security of NOSQL database. NOSQL injection attack and sql injection attack are performed on different area application. NOSQL database attacks are performed on the database level or application level, depending upon the data model used and API of NOSQL, whereas SQL injection are performed on the database engine. In this paper, MongoDB is used to show the injection example and defense method.

A. THREATS ON THE SECURITYOF MONGODB NOSQL DATABASE

MONGODB provides CURD operation, in which we can create, update, retrieve and delete the database. Query is the most useful function that is used by mostly database users. Database system provides an input box which allows users to input information of student they want to find. They can search the student record by email id or phone no. Developers takes the value from input box into variable and these variable would be used in query statement .Malicious injection usually happens from the input box because user can enter any information whatever they want. These variables are used in query execution which brings serious problem. If developer do not check the value of variable or sanitize the user input. This is the biggest threat that the malicious code will be executed when we call the query function which produces an negative impact on database security .When malicious code is executed then it will let the result always be true by using notation in Nosql Mongodb. When injection occurs then it results in insertion of unexpectedly data, deletion of data and even crashing the whole database. Query student by phone no in sql and in nosql Mongodb is given below.

```
    SQL          :    “select * from student where (phone = ‘ “
+ ph_number  +” ’);”

    MongoDB  :     db.collection.find ({phone= ph_number})
```

Fig.1 query in sql and mongoDB

Hacker can enter malicious code into input box to execute which is called injection. Hackers get the information of the database by executing injection successfully.

B. MONGODB NOSQL DATABASE

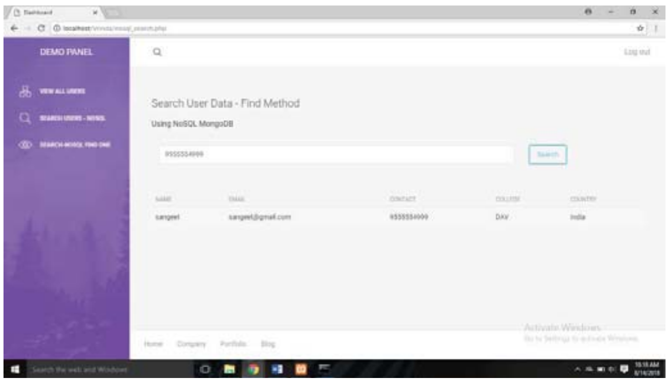
Mongodb provides high scalability, good performance and intended to be scalable .it is used for those application that required unstructured or semi structured data. It supports JSON or BSON format .MongoDB can handle millions or billions of records. It is a real time database .The database supports indexing over array and embedded objects. The name of query language of MongoDB is Mongo Query language .In this we can retrieve required document from a database collection. It uses CURD operation. MongoDB nosql database, lacked security functionality when they first emerged [7,8]. Mongodb is used by those projects that deal with big data. Day by day, MongoDB is gaining more and more popularity for companies because it can store huge amount of big data. Injection occurs in input boxes as we mentioned in the previous section.

C. MONGODB NOSQL DATABASE INJECTION

MongoDB has been opted by more and more organisation for data management .In this paper, we will use injection in mongoDB [1]. We will also analyse and detect injection in nosql database. Injection is a method to attack on the database or even to crash the database .The detection of nosql injection can be understood through an example demonstrating the various nosql injection attack classification. Injection occurs on database by inputting a malicious code into the input box. By using this malicious code hackers gets user authority and obtain information from the database. Hackers inject malicious code into input box .In the case of searching ,the malicious code becomes a variable that participate in execution .Program starts to search on the basis of user input .when malicious code is executed then it always bring true results. As a result, hackers will get all data without passing input.

Consider, there is a student database in mongodb which stores all the student data such as name, college _name, email _id, phone _no, location etc. Users search the student information by inputting email _id or phone _no in input box. If the email _id or phone _no is matched, the system will displaydetailof student.

Current output with phone _ no is given in figure 2.



Search User Data - Find Method

Using NoSQL MongoDB

9555554999

NAME	EMAIL	CONTACT	COLLEGE	COUNTRY
sangeet	sangeet@gmail.com	9555554999	DAV	India

Fig. 2output with phone no

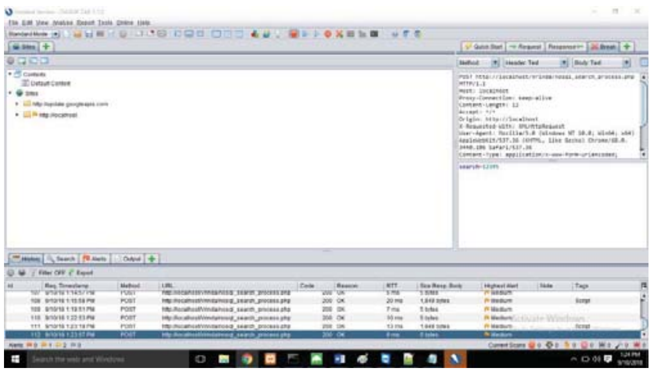
We mainly use injection by input box .For this, Assume that system is written by php and javascript for passing the values and query to database.

```
$search = $_POST['search'];
$collection=$db->student;
$cursor=$collection->find(
                              array(
                                  '$or' => array(
                                        array('phone'=>$search)
                                        array('email'=>$search)

          )

          )

          );
```

Fig.3. passing the value and query to database

In this case, the system gets the value from input box by $post [‘search’] and then send it to a search variable. After that the value of database will be store in collection variable. The system starts to query by email _ id or phone _ no and the matched value will be stored in cursor variable. Then it start to search document wise and output the student details.

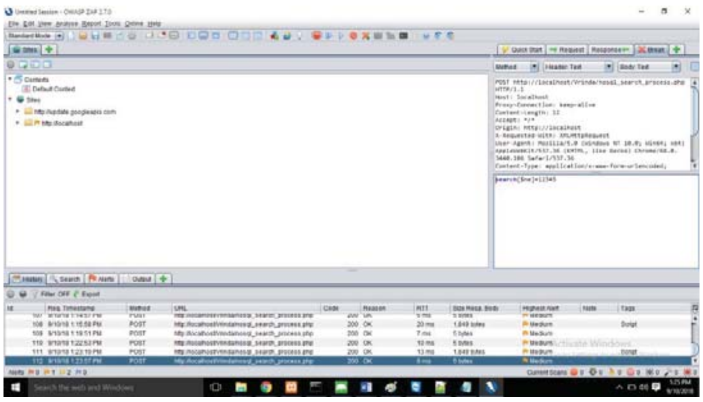
It means student information displayed only when the results of the find () function is true. Thus the hackers enter malicious query in order to true its statement.The malicious code inserted by the hacker will be look like this.



```
POST http://localhost/Vrinda/nosql_search_process.php
HTTP/1.1
Host: localhost
Proxy-Connection: keep-alive
Content-Length: 11
Accept: */*
Origin: http://localhost
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.
3440.106 Safari/537.36
Content-Type: application/x-www-form-urlencoded;

search=12345
```

Fig. 4 window to enter the malicious code

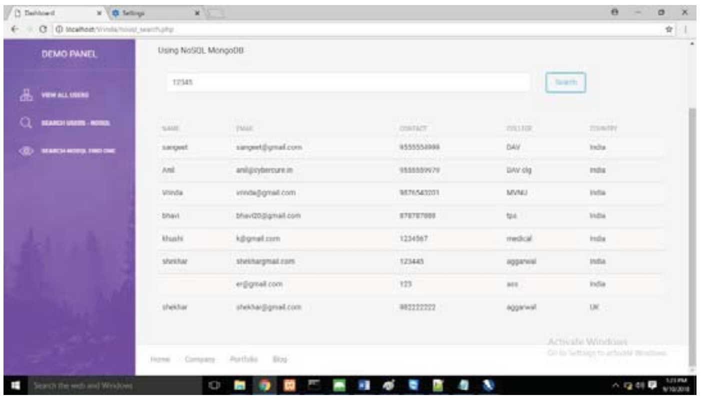
By inserting this malicious code, it does not matter what the hacker input because of the ! = operator. The results of find () function will always be true so that it will display all the data except where phone _ no =12345.the system searches the student Information by the given feature the code is similar to the one shown in fig. 5



```
POST http://localhost/Vrinda/nosql_search_process.php
HTTP/1.1
Host: localhost
Proxy-Connection: keep-alive
Content-Length: 12
Accept: */*
Origin: http://localhost
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.
3440.106 Safari/537.36
Content-Type: application/x-www-form-urlencoded;

search[$ne]=12345
```

Fig. 5 malicious code using $ne

We cannotapply! = directly on our system. We execute this on some web Penetration testing framework.This tool help us to identify vulnerabilities and to also help in verify attack vectors. When used as a proxy server it allow the user to manipulate all of the traffic that passes through it, including traffic using https.This tool is written in java language. According to the parameter standard, we use [$ne]insteadof! Operator .Hence, the injection code in input box will be like this search [$ne] =123445.The result of injection is given in fig 6.



Using NoSQL MongoDB

12345

NAME	EMAIL	CONTACT	COLLEGE	COUNTRY
sangeet	sangeet@gmail.com	9555554999	DAV	India
Anil	anil@cybercure.in	9555559979	DAV clg	India
Vrinda	vrinda@gmail.com	9876543201	MVNU	India
bhavi	bhavi20@gmail.com	878787888	tps	India
khushi	k@gmail.com	1234567	medical	India
shekhar	shekhargmail.com	123445	aggarwal	India
	er@gmail.com	123	aes	India
shekhar	shekhar@gmail.com	982222222	aggarwal	UK

Fig.6 Result of injection

By using this strategy, notation $ will be executed in MongoDB and hence injection will also be executed. Basically,to execute injection, the main idea behind this is to make the condition statement always true so that the result can get passed when user search document wise and it will display all the information of the database to the user.

IV. MONGODB NOSQL DEFENSE AND DETECTION ANALYSIS

For searching issues, we make so much effort to make our system robust. But if there is one place of negligence then all effort will be in vain because the attacker use that point to make a breakthrough.

Injection results in the leak of confidential information.The hacker gets root authorization .By getting root authorization, he can access all the drives or even he can delete the data. Attacker can deface the system as well. Therefore, it is necessary to build the function of defense in order to improve the system security.

For defense, it is necessary how to avoid injection from input box in website .After that we will provide detection approach, according to the vulnerability.

A. MONGODB NOSQL DEFENSEANALYSIS

In this paper, for MONGODB defense we will discuss 2 ways. The first one input validation which is to limit user input. As for phone no., query that should be all numbers. While writing the code, developer can add some constraint in code to limit the input box so that it can accept only number in phone no as shown in fig.7

```
$id =$_GET [‘id’];
 If ( ! preg_match (‘ / ^ [0-9] * $ / ’, $id ) )
{
echo “ Error ”
        }
  else
        {
           echo  “this is  valid number ”
        }
```

Fig.7 Code to limit user input

Since notations are used in malicious code. So this method is used to prevent all unwanted characters .By using this solution, we can check the input before it is passed to variable .The second one, parameterized statement is to check and filter the variable for a query. While writing condition statement, variable cannot be directly embedded into the condition statement.

However, we can use parameterized statement to pass variables for the user input. Parameterized statement uses parameter rather than embedding user input variable into the query. Using this method, we can remove most of injection attack. For example there is some code to find the character in a variable if it contains number only or not, by checking the phone no .if yes, then it will pass the value otherwise reject. The code is shown in fig. 8

```
If ( is_numeric ( $search ) = = “true”) )
Else echo “incorrect”;
```

Fig.8 Call for parameterized statement

Beside parameterized statement, developer can also check and filter the special character. It is very important to ensure that values in the user input cannot be directly passed into the condition statement .If the malicious code cannot be executed then the system will be much secure.

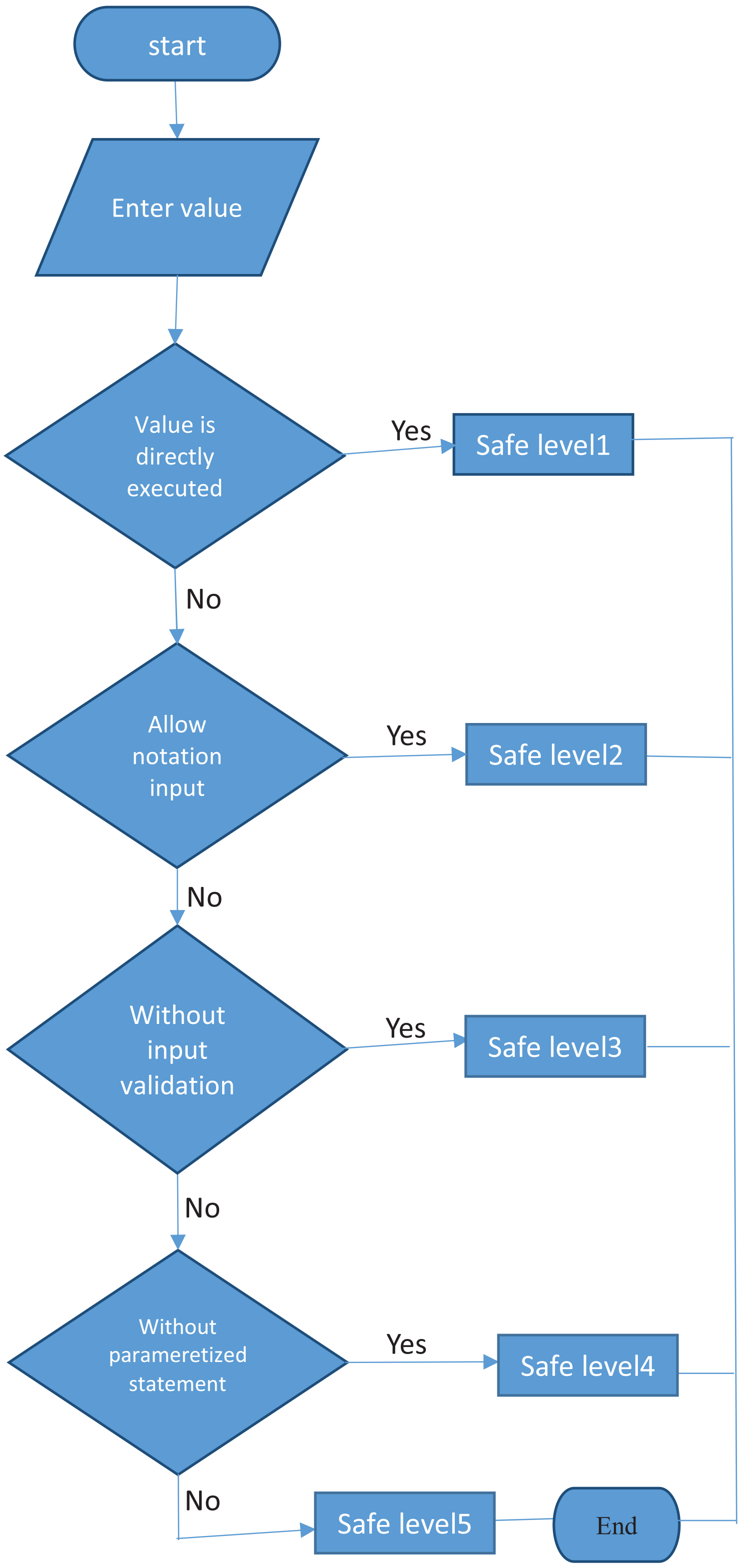


Fig.9Flow chart to detect maliciousfeature usingMONGODB

In the paper, input validation and parameterized statement method are discussed for defense purpose. Malicious feature detection technique find out if something is challenging for security. It also detects those system or software feature which are harmful for security .Based on malicious code and feature, flow chart is shown in fig. 9.

This simple flow chart for malicious detection help developer to detect malicious code .From this, we can detect nosql basic injection. This detection also helps to find the level of safety. Safe level 5 is highest safe level so that nosql database is more secure in this level.

V. CONCLUSION AND FUTURE WORK

Information security play most important role in system. While writing code, developer should apply defense method to make secure system .By applying defense method, user developer can prevent from injection or any other attack.

In this paper, we have discussed basic nosql injection. In future work we plan to examine some advanced nosql injection possibilities on MONGODB, as well as to study how to defense and mitigate attack to make more secure system.

REFERENCES

[1] Boyu Hou,Kai Qian (2016) “MongoDB NoSQL injection Analysis and detection” *2016 IEEE 3rd International Conference on Cyber Security and Cloud Computing .*

[2]Boyu Hou,yong shi (2017) “ Towards analyzing MongoDB NoSQL security and designing injection defense solution”*2017 ieee 3rd international conference on big data security on cloud (bigdatasecurity), ieee international conference on high performance and smart computing (hpsc), and ieee international conference on intelligent data and security (ids), 26-28 may 2017.*

[3] Chickerur, Satyadhyan, Anoop Goudar, and Ankita Kinnerkar(2015) "Comparison of Relational Database with Document-Oriented Database (MongoDB) for Big Data Applications." *28th International Conference on Advanced Software Engineering & Its Applications (ASEA) 25 Nov. 2015: 41-47.*

[4]Ron, Aviv, Alexandra Shulman-Peleg, and Emanuel Bronshtein.(2015) "No SQL,No Injection? Examining NoSQL Security." *arXiv preprint arXiv:1506.04082 .*

[5]Changlin He(2015),“Survey on nosql database technology”,journal of applied science and engineering innovation vol. 2 no. 2.

[6] Roshni Bajpayee,Sonali priya Sinha,Vinod Kumar (2015),“Big data :A brief investigation on NOSQL database”,*International journal of innovations & advancement in computer science,volume 4, issue 1 january 2015*

[7] Kadebu, Prudence, and Innocent Mapanga(2014), "A Security Requirements Perspective towards a Secured NOSQL Database Environment." *International Conference of Advance Research and Innovation.*

[8]Sharma, Chandershekhar, and SC Jain(2014), "Analysis and classification of SQL injection vulnerabilities and attacks on web applications." *Advances in Engineering and Technology Research (ICAETR), International Conference on* 1 Aug. 2014

[9] Noiumkar, Preecha, and Tawatchai Chomsiri(2014),"A Comparison the Level of Security on Top 5 Open Source NoSQL Databases." *The 9th International Conference on Information Technology and Applications (ICITA2014) .*

[10] Manoveg Saxena,Zakir Ali,Vinod Kumar Singh,(2014),“NOSQL database –analysis,Techniques and classification” *journal of advanced database management & system,volume 1 issue 2.*

[11]Abramova, Veronika, and Jorge Bernardino(2013). "NoSQL databases: MongoDB vs cassandra." *Proceedings of the International C* Conference on Computer Science and Software Engineering* 10 Jul: 14-22.

[12] Pokorny, Jaroslav(2013). "NoSQL databases: a step to database scalability in web environment." *International Journal of Web Information Systems* 9.1 : 69-82.

[13]Vatika Sharma,Meenu dave(2012),“SQL and NOSQL databases”,*international journal of advanced research in Computer science and software engineering,volume 2 issue 8,August .*

[14]Okman, Lior et al(2011), "Security issues in nosql databases." *2011IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications 16 Nov. 2011: 541-547.*

[15] Jing Han,haihong E,Guan Le,Jian Du(2011),“survey on nosql database*”, 2011 IEEE.*

[16] http:// php.net/manual/en/mongo.security.php

[17] “No SQL Injection in MongoDB” https://zanon.io/posts/nosql-injection-in-mongodb.

[18] https://www.mongodb.org