Available at: www.sabauni.net/ojs

S.J.I.T.N
Saba Journal of Information Technology and Networking

Article

METHODS OF SAFEGUARDING THE SITES FROM SQL INJECTION

Muneer A. S. Hazaa, Muneer Ali Saif Algabry, Mahmoud Mahub Qaid Altayar*

Thamar University Faculty of Computer Science and Information System

Article info

Article history:

Accepted Jan, 2016

Keywords:

Structure Query Language (SQL)

Injection Attacks

Abstract

Due to the rapid expansion of internet, web applications have become a part of everyday life. Consequently, this has increased the number of web application incidents and exploits web application vulnerabilities. For that consider the SQL injection type of attacks that target web ap-plications and allows attackers to obtain unauthorized access to the backend database to change the intended application-generate.

* Corresponding author: Muneer A. S. Hazaa
E-mail address: muneer_hazaa@yahoo.com
© 2016 Saba Journal of Information Technology and Networking, Published by Saba University. All Rights Reserved.

Introduction

SQL injection is considered to be one of the simplest types of attacks in nature and the most dangerous for web applications. There are a lot of Website developers who do not realize the nature of the attacks and therefore many of them do not perform the simplest of preventive measures to protect databases they are dealing with against the impact of these attacks. This gap is considered to be common in most of the Websites and through which most of the websites are penetrated and quite important data gets stolen. In this paper, we will focus in the first section on how this gap happens and how to address it. In the second part, we will focus on how to raise the protection level of the site for the prevention of pirate attacks by hackers. [1] Thus, we will focus on a big problem in websites. This study makes it open for other researchers to study other problems in sites.

Methodology

The authors used programmable functions (see section 2) that codify the password through making it pass through many levels in a way as to make it much more complex for hackers to execute their plans. Such functions are so useful in making webs much safer. We also used a program to discover all the gaps (see section 5).

Research Problem

There are programmable gaps by SQL injection in the websites that make them accessible for hackers. So, such gaps need to be detected and countermeasures should be made to save the websites.

SQL Injection

Query Language injection is to add symbols and SQL statements to the variables that are passed as parameters for the query where these sentences are implemented with the underlying query and then the attacker can have unauthorized access to the system databases and retrieve sensitive information-on from databases .[2]

Attacks pose greater risk due to the fact that they impact databases which are critical to any organization. [3]

Background on SQL injection vulnerability

Many people say they know what SQL injection is, but all they have heard about or experienced are trivial examples. SQL injection is one of the most devastating vulnerabilities that has a great impact on a business; as it can lead to exposure of all of the sensitive information stored in an application's database, including handy information (such as usernames, passwords, names, addresses, phone numbers, and credit card details).[4]

Examples of realistic breakthroughs caused by the SQL gap

Among the institution breached banks are PNC Bank Nasdaq Stock Exchange, Heartland Payment Systems and many others, which led to losses estimated at hundreds of millions of dollars suffered by these companies. The interesting thing is that these people carried out the break through

on a long period of time from 2005 until the time of their arrest in 2012.

According to the source, they use these gaps in SQL databases to enter them and then install some of the codes, that allow them to enter through a back door to private networks of such institutions breached the time they want .They have been able to obtain data on the numbers and more than 160 million bank account Credit Cards through that process. [5]

Some major hacks regarding SQL injection [7]

- July 2012, Yahoo confirms 4 million accounts hacked.

-June 2011, Hactivist 'Lulzsec' breached the website of SONY.

-May 2011, COMODO Brazil got breached

-March 2011, Official homepage of MySQL website was compromised.

-November 2010, Royal Navy website was attacked

-January 2009, Heartland payment systems got breached.

-June 2007, Microsoft UK website was defaced.

Preventing SQL In Existing Applications

SQL injection issues are relative new in the information security area. Many old systems were designed when developers were not aware of such threats. In fact, SQL injection vulnerabilities are so prevalent that simple Google searching can find many of them. To rewrite all of the vulnerable code sections of an existing system is both time-consuming and often impractical due to financial or time constrains. Therefore, techniques for protecting deployed systems against SQL injection attacks are important.

[8]This part will present how to protect our gaps SQL. We must follow the following:

Screening and matching input variables in terms of the type and length

We must examine any input before passing it to the query sentences. Let's say we want to display data for a product, according to the product number and the imposition of the latter part of the link would be as follows:

index.php? cat = 20

As seen at the link above, there is a variable called cat value that was passed as 20 and this value will definitely query in a table of products supposing that the query is as follows:

Select * from category whereid_cat = '$ cat'

Select * from category whereid_cat = '20 '

As noted in the previous query that the value of the variable cat has been passed to the query without any examination of the data contained before passing it by the attacker. The attacker could exploit this vulnerability to pass other values for the implementation of other queries such as

20 '+ union + select + * + from + category + order + by + '1';

index.php? cat = 20 '+ union + select + * + from + category + order + by + '1'

The query will be as follows:

We note here that the database category table or spreadsheet or others have been queried and the attacker can pass other queries to view other data

or executing orders that harm the site. This happens because of the lack of screening examination though the input examination is simple and we can address the former gap as follows : Since the value of the variable cat will be passed, the value to the field id_cat whose data type is digital .We can receive its digital value as follows:

$ cat = intval ($ _GET ['cat']);

The Previous function intval receives the variable values cat on the grounds that it is numeric values . Therefore any text that enters, the function ignores it and maintains only the number .Even if the data entered is a text, it will become zero.

We use this method with the variables of numeric type, while with the variables that receive text values , we must pass these values on one of the functions that add mark / before Marks ' or' and it will be as follows:

$ cat = mysql_real_escape_string ($ _GET ['cat']);

We can also test the length of the variable before passing them by the attacker. For example, if the length of the variable is 20 characters, we will receive the value of the variable only to the limits of twenty symbols as follows:

$cat=mysql_real_escape_string (substr($_GET['cat'],1,20));

Example for SQL Injection Show in Figure1

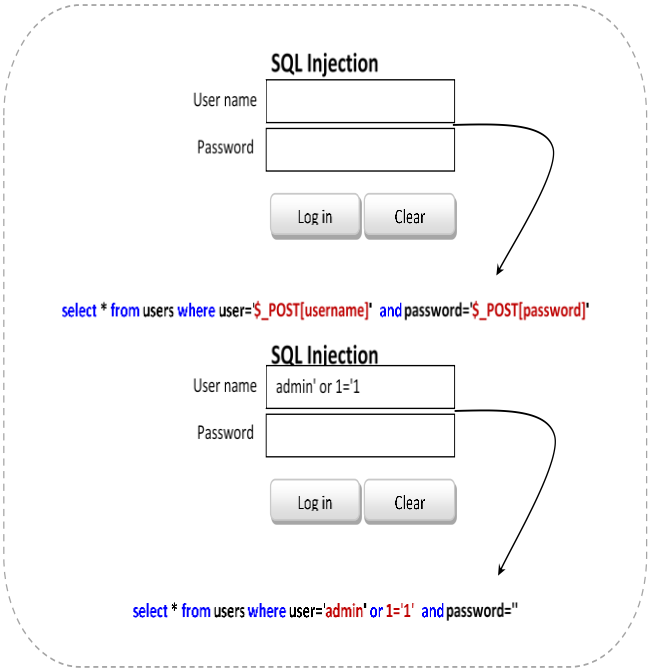

Figure 2 show how can pass a query by attacker in the bar.

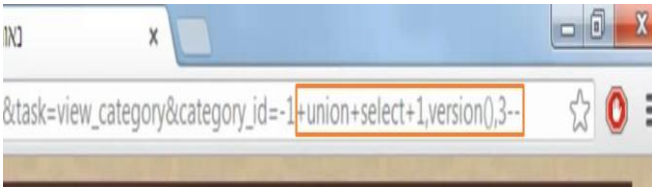

Figure 2. Pass a query union in the web site a bar.

Error messages (Hide)

Site programmer must hide error messages that appear when there is an error in the site so as not to be exploited by the attacker.

For example, we can adjust the settings file php.ini to prevent the emergence of error messages by controlling the following characteristic.

display_errors = off

log_errors = on

We can also work messages prepared in advance in order to appear for the user in case any error may occur.

<? Php

$ cat = intval ($ _GET ['cat']);

$ sql = "select * from category where id_cat =". $ cat;

$ q = mysql_query ($ sql) or die (" Unable to execute the query because of erroneous input ");

?>

We also note that the above-mentioned function of Die has been used to display a message when there is an error in the implementation of the query.

We can also use the @ sign before the name of the function to avoid the appearance of error messages.

Encoding Passwords Using More Than One-Way Function.

Encrypting passwords in the database using unidirectional encryption as follows,

If the encryption function here is by using md5,

The decryption is not impossible. But when we encrypt the password using the overlapping encryption, the encryption would be very strong.

Example:encryption password by more than function is as follows:

```
<?php

$pass=sha1(mysql_real_escape_string(strip_tags($_POST['pass'])));

echo "Leve 1 : ",$pass,"<br>";

$pass=sha1($pass);

echo "Leve 2 : ",$pass,"<br>";

$pass=md5($pass);

echo "Leve 3 : ",$pass,"<br>";

echo "Leve 4 : ",md5(sha1(sha1(mysql_real_escape_string(strip_tags(($_POST['pass'])))))),"<br>";

?>
```

In this way we have made the password encryption as difficult as possible and the more the multiplicity of levels of encryption the more difficult the decryption is.

For example: if we encrypt (muneer), the password encryption levels will be as follows:

Level1: 1c2c0fef3c4f1b85f97db36724a08eb291ce6d84

Level2: 07e1bdb71b98487a181245275efaff2e1be89052

Level3: e1aaf499a7be62f4a5aa586801906470

Level4: e1aaf499a7be62f4a5aa586801906470

As we have seen above, the password encryption has become strong. If the penetrator tries to decode this code e1aaf499a7be62f4a5aa586801906470, it will appear for him after a huge effort as follows: 07e1bdb71b98487a181245275efaff2e1be89052

Thus, this is not the decoding of the encryption and so we have increased the difficulty of decryption. If the penetrator arrived to the database and got the password, there will be several levels of difficulty ahead.

Hiding Of Variables That Appear In URL

As we have seen previously, SQL injection is passed through the URL addresses so that we can hide the variables that are passed by titles by concealing links and converting them into html by

Mod rewriter in order to limit the penetration site via a URL.

Using the file: we can perform this by using Htaccess. The code will be as follows:

URL Rewriting with PHP

http://www.apache.org/BookDetails.pl?id=5

You could provide a filter which accepts URLs such as http://www.apache.org/Book/page5.html

The following is what needs to go into your htaccess file to accomplish that:

Rewrite Engine on.RewriteRule ^ Book/page([09]+)\.*(html*)$BookDetails.pl?id=s 1

Note: to activate this characteristic, we must go to the file httpd.conf,and #LoadModulere-write_module modules/mod_rewrite.so

We remove # sign and it becomes in this form:

LoadModulere-write_modulemodules/mod_rewrite.so

Hence, the feature is activated.

Note: Must be enabled Mod_rewrite.

LoadModulere-write_modulemodules/mod_rewrite.

Put A Fire Wall On The Control Panel Folders

To protect the folders, we must first logon the control panel of the site through C Panel. http://www.website.com/cpanel

Figure 3 shows protect folders Control Panel.

Password Protected Administration (Htpasswd)

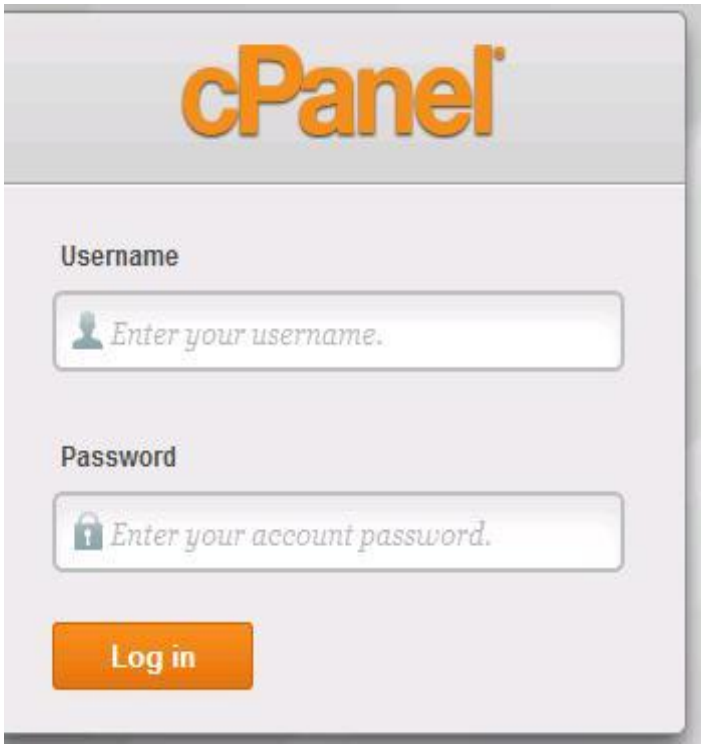

Figure 3, Login Control Panel.

We write the username and the password and then click on login which has in previous figure3.

The control panel appears as figure 4.

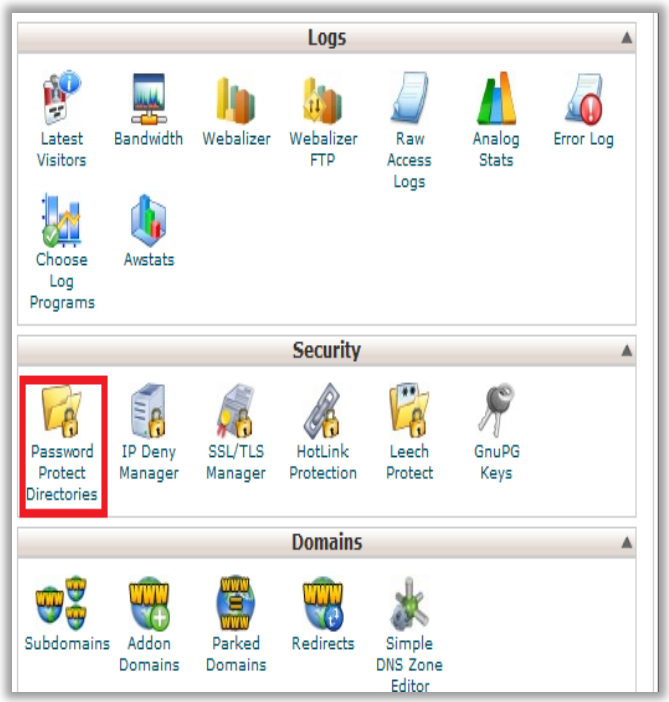

Figure 4, Protected Directory

From the previous window we click on Password Protect Directories to appear as shown in the following *figure:*

Figure 5. Password protect Directories

From the previous window we click on the name of the folder that we want to protect for, example the folder Admin, to appear the window as shown in figure *6:*

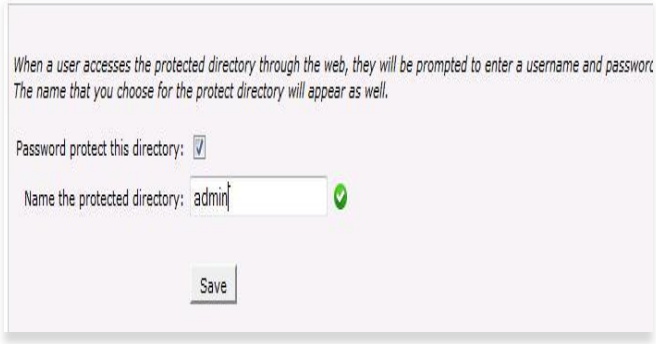

Figure 6. Protected directory

From the previous window, we write the name of the folder that we want to protect and then click on Save.

Figure 7, Add or Modify Authorized User

From the previous (figure 7), we create a user name and a password, and then we click on the button Add /modify authorized user for protecting the folder of management with a password .When you attempt to access the Management folder , the firewall window appears as following figure :

http://www.website.com/admin

Figure 8, Access to the Management Folders

We have thus put a firewall on the control panel of management to ensure that access to the Management folder is not easy. Also, we would better choose an uncommon name for Management folder , that is ,we change the management folder

name to numbers and names which are difficult for the hackers to guess .

Change Paths and Names of Folders Control Panel

Methods of protection task are to change the names of folders and paths to the control panel so that it is difficult for the hacker to guess them. We should not give the Management folder names that are commonly known. Such as Administrator, Admin, manager, user, control, login, log, Cpanel, panel,

We must give them names that are unknown and difficult to guess by hacker programs that are used in the process of guessing For example, we can name the Management folder as d3xcs08e better than to give it the name Admin. After giving it an unknown name, we should also protect it with a firewall. It is also better to create folder names for known names to the control panel. Such as admin and administrator and make these folders empty and set up the firewall so as to delude the penetrator that this is the control panel so as not try to search for the real folder of the Control Panel.

Prohibition of Visitors Who Pass On Codes of Injection Queries

Useful ways to prevent the hacker from targeting a site is blocking the site when the error message of the prevention of injection process appears. We record the IP address of the visitor that caused the error and when the error is repeated more than twice or three times. For example, we block the site for the visitor for an hour so as to avoid the process experience injection thread by hackers. This is an effective way to prevent the pirates from trying to experiment on the target site.

Web Vulnerability Scanner

Vulnerability Scanner scans your web application for vulnerabilities. We used a program called ***Acunetix vulnerability scanner*** to scan the web application in a way as to show all kinds of gaps (see figure 8).This helped us to discover the dangerous gaps by which the hackers get an unauthentic access to the backend databases. Therefore, we can make protection for such applications. The program can be used this way easily:

Open Web application and click “Scan Site" for whole site scanning or "Scan URL" only for current URL.

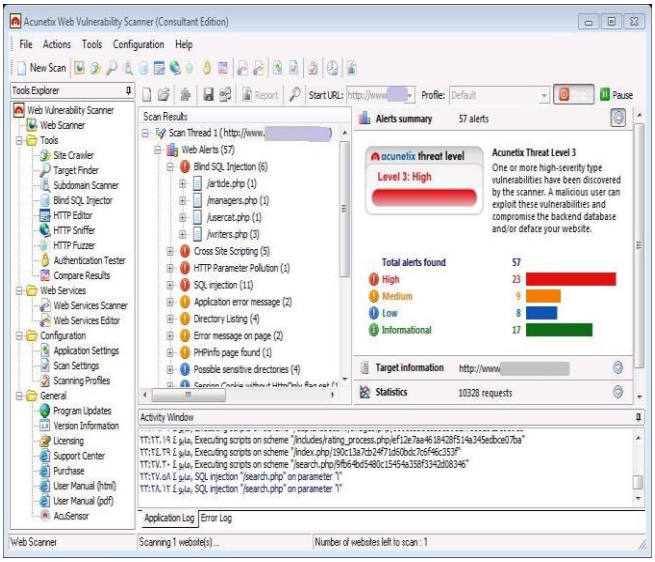

Figure 9, Web Vulnerability Scanner

Conclusions and Recommendations

This section includes the most important conclusions of this research.

In this research, we found that the intrusion protection requires a re-examination of the site, and programming to address the gaps in it. Moreover,

it is found that developing several levels of protection on the website is required even if the first level exceeded breach collide at the second level and so on .In this research, we found that choosing a hosting company, which provides powerful means of protection on the part of the server is important, in addition the level of security at the sites needs to be improved after the addition of the levels of protection mentioned in this research. There are many gaps that were not mentioned in this research so it is not sufficient to address the gaps mentioned, but you should search the rest of gaps and address them.

Means of protection that have been mentioned in this research are not considered as 100% efficient for protection because there is no 100% protection. The field is broad and it is pirated everyday so we must always search for the latest attacks and innovate new means of defense to make it more difficult to be hacked. Finally, detecting and blocking attacks against known vulnerabilities is required. The knowledge base of exploitable weaknesses in the application must be frequently updated

References

[1] Center of Excellence for Information Security.

[2] A Survey on SQL Injection attacks, their Detection and Prevention Techniques *V. Nithya, IJECS Volume 2 Issue 4 April, 2013 Page No. 886-90*

[3] Mittal, P. (2013). A Fast and Secure Way to Prevent SQL Injection Attacks using Bitslice Technique and GPU Support (Doctoral dissertation)..

[4] SQL Injection Attacks and Defense 2009 Justin Clarke Lead Author and Technical Editor.

[5] Department Of Justice http: //www.justice.gov/opa/pr/2013/July/13-crm-842.html

[6] History of SQL Injection"[Online]. Available: http: //hackertarget.com/10-years-of-sql-injection.

[7] DARK SIDE OF SQL INJECTION ASAR International Conference, Bangalore Chapter- 2013, ISBN: 978-81-927147-0-7

[8] A Survey Of SQL Injection Countermeasures. June 2012. Dr. RP. Mahapatra and Mrs. Subj Khan