iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** V **Month of publication:** May 2020

DOI: http://doi.org/10.22214/ijraset.2020.5011

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

Web Penetration Testing

Ujjwal Gupta[1], Sarthak Raina[2], Prabhat Verma[3], Priyanshu Singh[4], Mr. Madhup Aggarwal[5]
[1, 2, 3, 4]*Fourth Year,* [5]*Assistant Professor, Department of Computer Science, RKGIT , Ghaziabad , U.P,*

***Abstract**: As technology changes, it becomes increasingly challenging for businesses of all types to keep their personal and customer's information on the web secure. Web security is important to keeping hackers and cyber-thieves from accessing sensitive information. Without a proactive security strategy, businesses risk the spread and escalation of malware, attacks on other websites, networks, and other IT infrastructures. If a hacker is successful, attacks can spread from computer to computer, making it difficult to find the origin. This project deals with preventing the potential errors while developing a basic website in order to prevent it from possible cyber-attacks. Cyber-attacks will be performed on unsecured site and then its vulnerabilities will be compared with the secured site.*

***Keywords**: Web Penetration Testing, SQL injection, Local File Inclusion, Session Hijacking, Parameter Tampering, Cyber Attacks, PHP, HTML, CSS, SQL.*

I. INTRODUCTION

A Penetration testing is the most commonly used security testing technique for web applications. Web Application Penetration Testing is performed by simulating unofficial attacks internally or externally to get access to sensitive data. A web penetration helps end user find out the possibility for a hacker to access the data from the internet, find about the security of their email servers and also get to know how secure the web hosting site and server are.

Importance and the need for Web App Pen Testing:

1) Penetration testing helps in identifying unknown vulnerabilities.
2) Helps in checking the effectiveness of the overall security policies.
3) Help in testing the components exposed publicly like firewalls, routers, and DNS.
4) Lets user find out the most vulnerable route through which an attack can be made.
5) Helps in finding the loopholes which can lead to theft of sensitive data.

This project deals with preventing the potential errors while developing a basic website in order to prevent it from possible cyber-attacks. Author will be making two websites in which one will be fully secured and the other will be containing some potential errors. The websites will be containing front end and back end. The front end of the website will be consisting of HTML,CSS,BOOTSTRAP and further technologies if required , the backend of the website will be developed using PHP and any other language if needed. We will be performing the general cyber-attacks on both the websites. The attack will not be able to harm from the secured website (like fetching data etc.) while it may be able to harm the other one. So this will clearly elaborate what are the important points which must be taken care of while developing any basic website to make it secure.

II. ATTACKS TO PERFORM

Following are the attacks that will be performed on both the sites the one which is less secured and the one which is secured in order to compare the vulnerabilities of unsecured site with the secured one.

A. SQL Injection

1) SQL Injection: SQL Injection (SQLi) is a kind of an infusion assault that makes it conceivable to execute pernicious SQL explanation s. These announcements control a database server behind a web application. Aggressors can utilize SQL Injection vulnerabilities to sidestep application safety efforts. They can circumvent confirmation and approval of a website page or web application and recover the substance of the whole SQL database. They can likewise utilize SQL Injection to include, change, and erase records in the database.

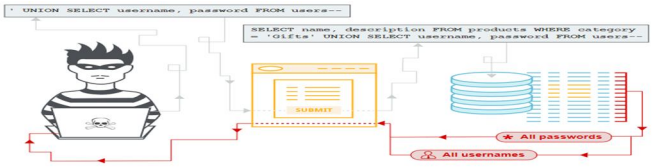


Fig. 1 Example of SQL Injection

B. Cross-Site Scripting (XSS)

Cross-site Scripting (XSS) is an attack on client case code injection. The attacker aims to make malicious documents in the victim's web browser by entering malicious code on the official web page or web application. The real attack occurs when a victim visits a web page or web application that uses malicious code. A web page or web app becomes a vehicle for deploying malicious script in the user's browser.Vulnerable vehicles that are commonly used for Cross-site Scripting attacks are forums,message boards, and web pages that allow comments.

C. Local File Inclusion (LFI)

Local file Inclusion is a file type vulnerability file found on PHP based websites and is used to affect web applications. This problem usually occurs when an application tries to retrieve certain information from a particular server where the input to a particular file location is not treated as a trusted source.

Usually, it refers to an attack where an attacker can provide valid input to get a response from a web server. In response, the attacker will be able to judge whether the input he gave you is valid or not. If applicable, then whatever/any file the attacker wants to see can be easily accessed. [1]

Fig. 2 Local File Inclusion Vulnerability

D. Parameter Tampering

It is one of the most common web based attacks. In it, the attacker tries to change certain fields in the Uniform Resource Locator without the permission of the user. This leads the user to another page which is different from the actual page which the user wants to access but looks similar to the page that the user wants to access. Parameter Tampering is used by cyber criminals and thieves in order to get the confidential information of the users or clients. And this information is used by the attacker later for cyber malpractices.

Best example that can be used as tampering of prices in an e-commerce website that uses hidden fields to refer to its items as follows,

<input type=”hidden” id=”1008” name=”cost” value=”70”>

E. Session Hijacking

Session Hijacking means gaining unauthorized access over an currently active TCP/IP session. It is done without the permission of the user. When it is done successfully, then the attacker gains the same access of the system as the original user. Now the attacker has same privileges on the data as the original user. This type of attack is possible only at the beginning of a TCP session. The major threat is that the attacker can get the information out of the respective database without having to hack of any of the authorized user accounts. [2]

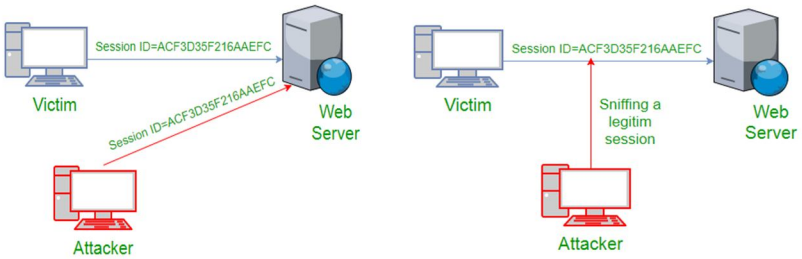


Fig. 3 Session hijacking process example

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

III. PREVENTION FROM ATTACKS

Prevention of these attacks is necessary because if our sites are vulnerable to these attacks then a huge loss of data can be experienced as well there will also be tampering and changing of the real values which was originally entered by the users. Prevention is done to keep the hackers away from exploiting the sites.

A. SQL Injection

If user input is inserted without modification into an SQL query, then the application becomes vulnerable to SQL injection like in the following example

```
$unsafe_variable = $_POST['user_input']; mysql_query("INSERT INTO `table` (`column`) VALUES ('$unsafe_variable')");

That's because the user can input something like value'); DROP TABLE table;--, and the query becomes:

INSERT INTO `table` (`column`) VALUES('value'); DROP TABLE table;--')
```

Fig. 4 SQL Query with vulnerability

So we can prevent this by,

```
$stmt = $dbConnection->prepare('SELECT * FROM employees WHERE name = ?'); $stmt->bind_param('s', $name); // 's' specifies the variable type => 'string' $stmt->execute(); $result = $stmt->get_result(); while ($row = $result->fetch_assoc()) { // Do something with $row }
```

Fig. 5 Secured SQL Query

B. Cross Site Scripting

Cross Site scripting is generally done by running a script in a specific area where a user is allowed to input something. The basic prevention is to prevent user to use syntax which runs the script.
Example : `echo htmlspecialchars($string, ENT_QUOTES, 'UTF-8');`

C. Local File Inclusion (LFI)

Improper use of few PHP functions is responsible for file inclusion flaws like in below example “included” takes a request parameter specifying the name of the file to be included at runtime. The code is not checking the input in any way hence the attacker can specify any file parameter and it will get executed, whether local or remote; an IP address, port number and filename.
The most effective solution for removing file inclusion vulnerabilities is to prevent users from passing input into the file systems and framework API. If this is not possible, the application can maintain a whitelist of files. These files must contain only characters (a-z) and numbers for file names. Special characters -- for example, the colon and slashes found in a URL, like http:// -- must not be included.

```php
<?php

    echo "File included: ".$_REQUEST["file"]."<br>";
    echo "<br><br>";
    include $_REQUEST["file"];
    echo "<br><br>";

?>
```

Fig. 6 LFI example

D. Session Hijacking

For regular browser user following some basic prevention techniques can help them to reduce the risk of session hijacking, but because session hijacking works by exploiting fundamental mechanisms used by the vast majority of web applications, there is no single guaranteed protection method.

1) Set the HttpOnly attribute using the Set-Cookie HTTP header to prevent access to cookies from client-side scripts. This prevents XSS and other attacks that rely on injecting JavaScript in the browser.
2) Web frameworks offer highly secure and well-tested session ID generation and management mechanisms. Use them instead of inventing your own session management
3) Use HTTPS to ensure SSL/TLS encryption of all session traffic. This will prevent the attacker from intercepting the plaintext session ID

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

IV. METHODOLOGY

This project deals with preventing the potential errors while developing a basic website in order to prevent it from possible cyber-attacks. Author will be making two websites in which one will be fully secured and the other will be containing some potential errors. We will be performing the general cyber-attacks on both the websites. The attack will not be able to harm from the secured website (like fetching data etc. while it may be able to harm the other one. So this will clearly elaborate what are the important points which must be taken care of while developing any basic website to make it secure.
Following approach is used:

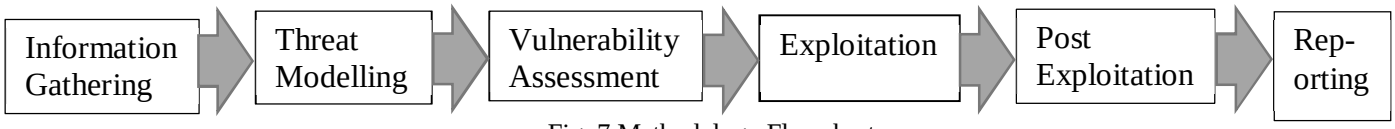


Fig. 7 Methodology Flowchart

A. Information Gathering

The first phase in a web application penetration test is focused on collecting as much information as possible about a target application. Information Gathering, is one of the most critical steps of an application pen test.

B. Threat Modelling

It is a procedure for optimizing network security by identifying objectives and vulnerabilities, and then defining countermeasures to prevent, or mitigate the effects of, threats to the system.

C. *Vulnerability Assessment*

A Vulnerability assessment is the process of defining, classifying, and prioritizing vulnerabilities.

D. Exploitation

A website vulnerability is a weakness or misconfiguration in a website or web application code that allows an attacker to gain some level of control of the site, and possibly the hosting server.

E. Post Exploitation

As the term suggests, post exploitation basically means the phases of operation once a victim's system has been compromised by the attacker. The value of the compromised system is determined by the value of the actual data stored in it and how an attacker may make use of it for malicious purposes.

F. Reporting

Reporting the vulnerabilities of a website to its owner and making them aware of the weak security if there is any. It helps in making the website more secure so that any security breach doesn't happen in future which can lead to losses.

V. PERFORMANCE METRICES

A. Technology to be Used

This application will use PHP as a scripting language. Since the authors will be developing 2 websites. So, it will involve both front end and back end technologies. For the front end part the authors will be using HTML, CSS and SQL for the database.

1) *Technology-1:* The first technology to be used is the HTML and CSS for the front end part. For the whole development phase Visual Studio Code and Notepad++ will be used as the integrated development environment (IDE). HTML is the standard markup language for the Web pages. Whereas CSS (Cascading Style Sheets) will be used for the formatting part
2) *Technology-2:* The second technology to be used is PHP for the back end part. PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages. PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP. PHP 7 is the latest stable release.
3) *Technology-3:* The third technology to be used is the SQL for the database. SQL is a standard language for storing, manipulating and retrieving data in databases. SQL stands for Structured Query Language. SQL is used to communicate with a database. According to ANSI (American National Standards Institute), it is the standard language for relational database management systems. SQL statements are used to perform tasks such as update data on a database, or retrieve data from a database.

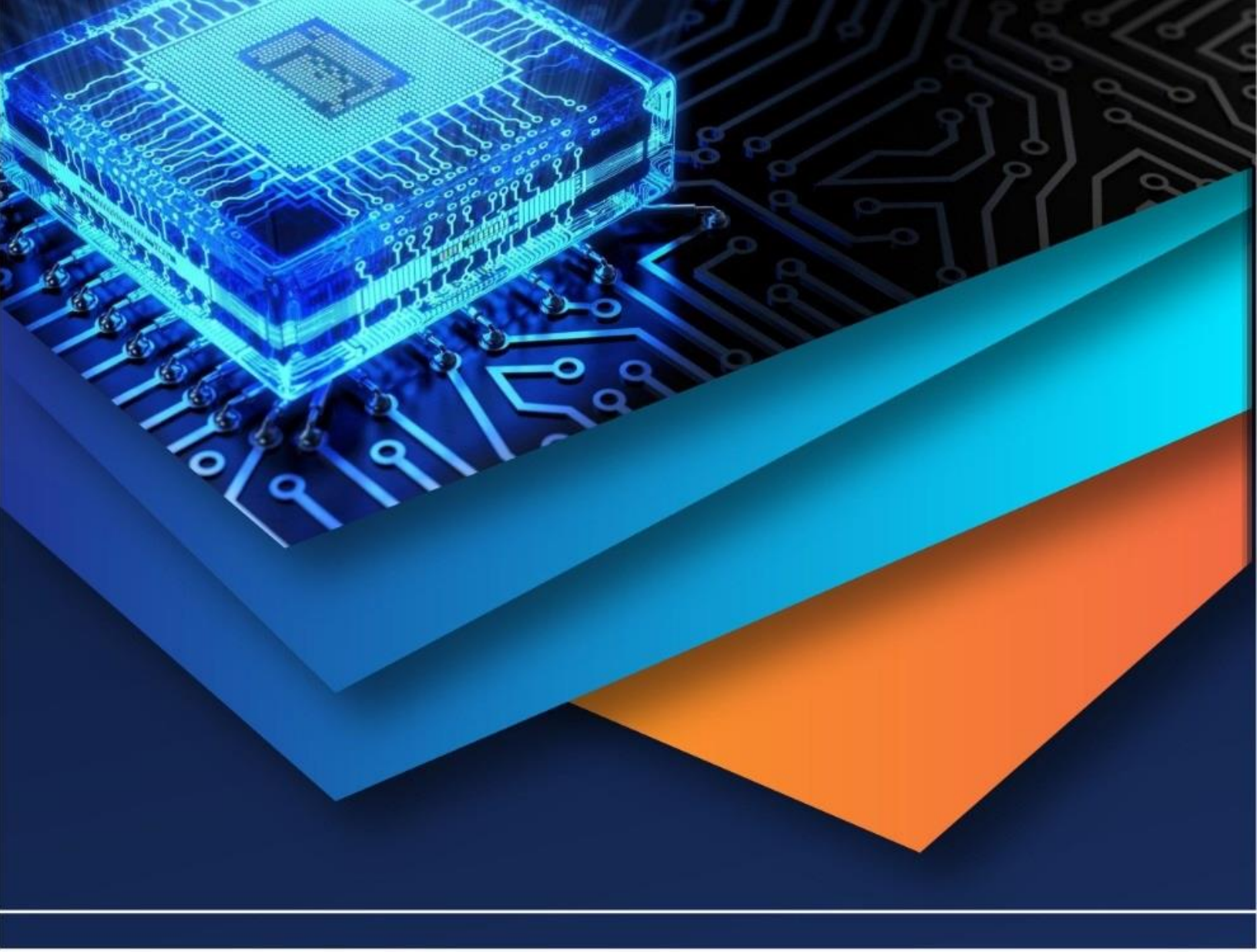
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

VI. DISCUSSION

In this paper we have discussed about the various attacks that can be performed in various website and the techniques to prevent those attacks. These all attacks will be carried out on the two sites in this project for the example purpose.

REFERENCES

[1] Author deepamamknp from geeksforgeeks https://www.geeksforgeeks.org/local-file-inclusion-lfi/
[2] Author Akash Sharan from geeksforgeeks https://www.geeksforgeeks.org/session-hijacking/



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY