

Web Vulnerability Detection and Security Mechanism

Katkar Anjali S., Kulkarni Raj B.

ABSTRACT– Web applications consist of several different and interacting technologies. These interactions between different technologies can cause vast security problems. As organizations are taking their businesses online they make their systems accessible to the world. They might have a firewall in place and possibly even their web server is running an up-to-date version of its software but that is not enough to protect their resources.

The research areas of this paper outline the major publicly reported security vulnerability in recent year's strong growth of the web applications. Unvalidated Input, Broken Access Control, Broken Authentication and Sessions Management, Insecure Configuration Management, Improper Error Handling, Parameter Modification, Cookie Modification and Directory Traversal have been the most dominant class of web vulnerabilities. Further, the research includes methods for detecting the vulnerabilities and then providing security mechanism to protect web application from those vulnerabilities.

The result shows the security mechanisms against the attacks and vulnerabilities. Securing the websites against these vulnerabilities is very difficult and challenging task as day to day new techniques for attacks are invented, so the study of various types of vulnerabilities, detecting the attacks and providing solution for these vulnerabilities is essential part in internet world.

KEYWORDS- Security, Vulnerability detection and Web applications.

I. INTRODUCTION

The World Wide Web, have become one of the most common communication mediums in the World. Millions of users connect every day to different web-based applications to search for information, exchange messages, interact with each other, conduct business, perform financial operations and many more. Some of these critical web-based services are targeted by several malicious users intending to exploit possible vulnerabilities, which could cause not only the disruption of the service, but also compromise the users and organizations information.

Most of the times, these malicious users succeed in exploiting different types of vulnerabilities and the consequences can be disastrous.

A. Literature Review

According to Johari, R.; Sharma, P., In the article "A Survey on Web Application Vulnerabilities (SQLIA, XSS) Exploitation and Security Engine for SQL Injection", main objective of this paper is only on the study of various types of Structured Query Language Injection attacks and Cross Site Scripting vulnerabilities and their security techniques. They proposed to future study on Structured Query Language Injection attacks.

Silic, Marin; Krolo, Jakov; Delac, Goran states in their article "Security vulnerabilities in modern web browser architecture". Studied various web browser architecture, as browser is the main source of vulnerability. Further, discussed on the various new web browser vulnerabilities, assess the web browsers architecture and how they response in web security problems.

According to Fonseca J., Vieira, M.; Madeira, H. "Testing and Comparing Web Vulnerability Scanning Tools for SQL Injection and XSS Attacks". On web application automatic vulnerability scanner are used to identify vulnerabilities. Main Objective of this paper is on study of XSS and SQL injection vulnerabilities, methods to detect and apply benchmark automatic scanners to detect software fault injection techniques.

Vieira, M.; Antunes, N.; Madeira, H. in their article "Using web security scanners to detect vulnerabilities in web services" detailed studied on various available scanners in the market to scan the vulnerabilities in the web application and compare their performance which will be suitable to apply on web services.

After the review of various researches on web vulnerabilities and security, focused mainly on the survey, scanner, different tools, various tester and detectors of vulnerabilities but not emphasizes on the prevention techniques to protect against vulnerabilities. The research area of this paper is focused not only on the study of various vulnerabilities such as Unvalidated Input, Broken access control, Broken Authentication and Sessions management, Insecure Configuration Management, Improper Error Handling, parameter modification, cookie modification and directory traversal. But also the methods on how to detect and provide security mechanism to protect the web applications.

We build a Blog website and demonstrate both secured and insecure version of the website. We demonstrate the attacks through Black box testing of each vulnerability.

Our main objective is to find out whether these security mechanisms provide significant help to the builder of a web application. That is, help in reducing the potential security risk of running the application. Secondly we will assess the level of security expertise needed to gain benefit from these security mechanism. Our scope, however, is limited to qualitative aspects of the products. Any performance issues are out of the scope of this paper.

Intended audience of the paper are web application designers, testers and architects, system administrators responsible for maintaining web applications, security professionals and academics.

The scope of this paper is limited to study the various vulnerabilities and the technic of assessing the vulnerability in web application and providing the protection according to their approach to web application. Thus, we will not test the performance of security on the web application. When possible, however, we will provide the security to help better understand their approach.

Manuscript received on September, 2012

Katkar Anjali S., Computer Science and Engineering department, Shivaji University, Walchand Institute of Technology, Sholapur, Maharashtra, India,

Kulkarni Raj B., Computer Science and Engineering department, Shivaji University, Walchand Institute of Technology, Sholapur, Maharashtra, India.

The rest of the paper is organized as follows. Section II discusses security vulnerabilities found in web applications and the current methods for analyzing vulnerabilities of web application. In section III we will evaluate security mechanism available for web application security assessment. In section IV represents the result analysis of the objective of research paper. In section V we will draw our conclusions of the study of web vulnerabilities and security mechanism.

II. VULNERABILITIES FOUND IN WEB APPLICATION

In this section we will discuss the various vulnerabilities mostly independent of the underlying platform and which are commonly found in commercial sites even though these vulnerabilities have been publicly known for some time now. Commonly web servers, application servers, and web application environments are susceptible to following types of vulnerabilities.

Unvalidated Input

Web application request inputs from user to determine how to respond in accordance to provide web service. The user enters input values to web application request. Attackers may pass harmful information to the web application which tries to bypass the website's security mechanisms.

Broken Access Control

In web application the users are categorized in the different level of privileges. Access control determines how the web application allows access to functions to some users and not others, also called authorization. But attacker may be access higher level of authority.

Broken Authentication and Sessions Management

Web application creates session when the user logged in, which specify the period of time that a unique user interacts with a web application. Using session maintains state by providing the client with a unique id. This id is stored in a cookie which is used between the user browser and web server. If this session's details are not protected correctly, attacker can steal it and misuse it.

Improper Error Handling

In the web application when error occurs under normal transactions and if not handled by proper error message to the user then there is chances of getting clues on flaws about the web application to the attackers and may disturb to the normal user.

Parameter Modification

Parameter modification is the problem where the attacker's do not fill the form but rather passes the parameters from URL itself, bypassing the form validations. Therefore it may lead to ambiguous effect on the form data and the overall site data.

Insecure Configuration Management

The web server that hosts the web application consists of web configuration files and directories which should not be accessed or view by someone unauthorized. So must be protected against the attackers.

Cookie Modification

Cookie stores the information in the text format which is used for state management. The web application uses

cookies; the server sends cookie and stores at client browser. The browser then returns the cookie to the server the next time the page is requested. The attackers can easily connect with the server to modify the contents of user's cookie.

Directory Traversal

Web server consist of directories where files, information, application functions are stored to provide the services which does not have access to users. But attackers obtain these unauthorized directories, by traversing the directory in the address area of web browser and may misuse it.

B. Vulnerability Detection

In this section we examine the techniques to find out vulnerabilities present in the web application.

Unvalidated Input

All user input that provided to the web applications requested, need to check by against strict format that specifies exactly what input must be allowed. Ensure that all parameters are validated before they are used. A tool or library is most effective, as the performing the checking should be placed.

Broken Access Control

The code implementation of the access control policy should be verified. Penetration testing can be useful in verifying if there are problems in the access control. In the web application, if there is categories of users that can be accessed through the interface, verify each interface to make sure that only authorized users can allowed access.

Broken Authentication and Sessions Management

Detailed review of authentication mechanisms to ensure that user's credentials are protected and only an authorized user can change them. Review your session management mechanism, that session identifiers are always protected.

Improper Error Handling

Error handling should be consistently focus on the entire application. A code review will reveal how the system is intended to handle various types of errors. Simple testing can determine how your site responds to various kinds of input errors.

Parameter Modification

Ensure that application must not allow parameter values, query string or form GET parameters to the URL. These are handled through dynamic parameter detection.

Insecure Configuration Management

Web server configuration files must not be accessed by the user, unauthorized files and directories permissions, Server software flaws or misconfigurations that permit directory listing and directory traversal attacks, some of these problems can be detected by scanning.

III. SECURITY MECHANISM

This section lists security mechanism to provide protection against attacks and vulnerabilities that a web application should be prepared for.

Unvalidated Input

All the input parameters to the web application should be validated against specification such as use of data type, length of all fields, whether null, duplicates are allowed, parameter is required or not, numeric range, specific legal values (enumeration), patterns (regular expressions).

Parameter validation services must be configured with definition of what is valid for each parameter for web application. This includes properly protecting all types of input from web application request, including URLs, forms, query strings and parameters.

Broken Access Control

Attackers will use path traversal method, which provides path information as part of a request for information. Such attacks may try to access files that are normally not directly accessible. Such attacks can be submitted in URLs as well as any other input that ultimately accesses a file. In such case redirect the page to custom error page message.

Broken Authentication and Sessions Management

Ensuring that implementation consistently enforces to have a secure authentication and session management mechanism which includes: passwords should have restrictions that require a minimum size; securing session id, a user's entire session should be protected; browser cache protection, authentication and session information never submitted as part of a GET parameter. Authentication pages must be specified with no cache tag to protect from using the back button in a user's browser; session tokens should be expired on the server, and destroyed when a browser is closed; Everywhere authenticate must be provided.

Improper Error Handling

In the implementation, ensure that the site is built to handle all possible errors. When errors occur, the application should inform the user with proper error message without unnecessary internal details. Provide the user with diagnostic information (e.g., validation errors), but do not provide developer level diagnostic/debug information. Limit error messages regarding user ID and password errors; do not describe the password complexity.

Parameter Modification

In this type of vulnerability attacker involves providing the input parameter in the URL of the web application in such case don't allow the URL as for example "...../User Login Page? Username = xyz & password =xyz123". In such case redirect the page to custom error page message.

Insecure Configuration Management

Develop a configuration tool; protect the web server's configuration files from user to access. The configuration process should include: protecting all security mechanisms from unauthorized user; remove the unused services from web application; provide the access rights(privileges) as roles, permissions, and accounts, including disabling all default accounts or changing the passwords option, logging options and various alerts.

Cookie Modification

Cookies are destroyed at the point the user closes the browser and unless a deletion date has been set. If the user closes the browser without sign out then default deletion date has been set, the cookie will be destroyed on that deletion date instead.

Directory traversal

Ensure you have installed the latest version of your web server software. In the web server don't allow anyone to access the root directory or any other configuration files as

only administrators can maintain access control list and whether a file can be viewed or executed by users, as well as other access rights. Also don't allow attackers to access files from root directory by using the address bar of web browser.

IV. RESULT ANALYSIS

Unvalidated Input

Any input web applications accept must be checked against a strict format that specifies exactly what input will be allowed. (Fig1)



Fig1: Unvalidated Input

Broken Access Control

Attacker may try to access the unauthenticated page directly by passing the inaccessible URL, in such case redirect the page to custom error page message. (Fig 2 and 3)

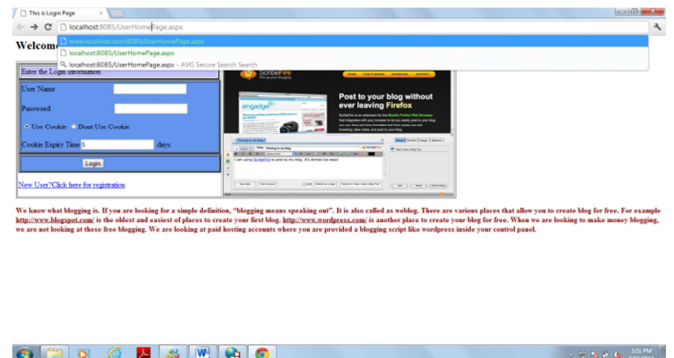


Fig2: Broken Access Control

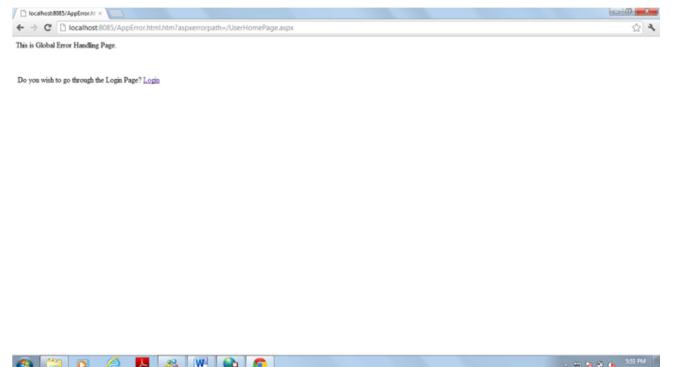


Fig3: Broken Access control (Custom Error page is displayed)

Parameter Modification

In this vulnerability, attacker tries to pass parameter in the URL of web application such as "http://...../UserLoginPage.aspx?username=xyz&password = xyz123"

Which don't allow attacker to login and redirect the attacker to custom error page. (Fig4)

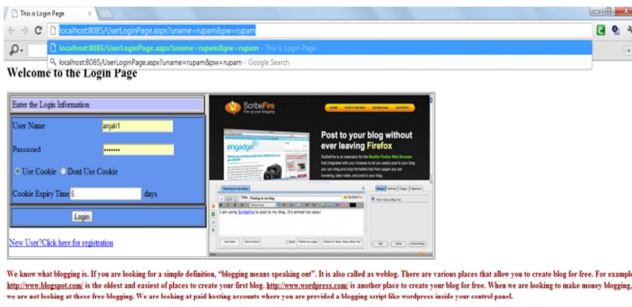


Fig4: Parameter Modification

Improper Error Handling

When errors occur, the site should respond with a specifically designed result that is helpful to the user without revealing unnecessary internal details. So if attacker passes the not existent URL, instead of displaying browser error message, it will redirect to custom error page.(Fig5 and 6)

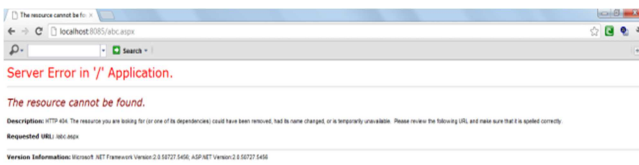


Fig5: Improper Error Handling

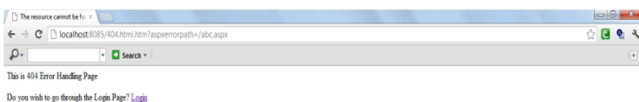


Fig6: Improper Error Handling (Custom Error page is displayed)

Directory Traversal

The root directory prevents attackers from accessing files, also the files residing outside of the root directory. If attacker tries to use http://...../ BlogWebsite, (For Example "BlogWebsite" is root directory) it will not enlist the files or directory of root directory.(Fig 7)

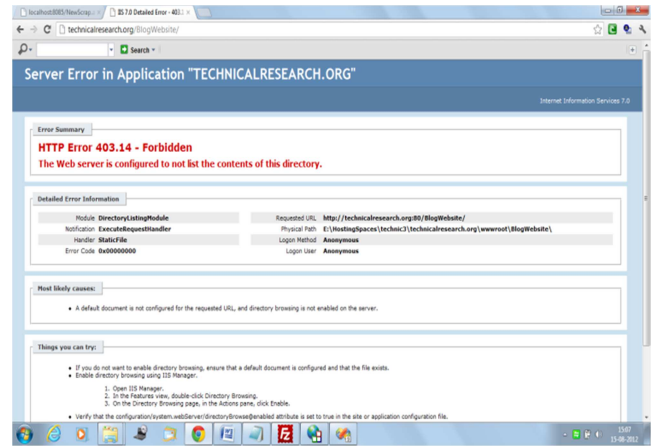


Fig7: Directory Traversal

Insecure Configuration Management

In this vulnerability, even if we try to access root directory locally, it will enlist all the files and directories. Click on any file like a bmp file or text files, it will be displayed. But on click of "web.config" access is denied. As the security is provided on configuration file of web application.(Fig 8)



Fig8: Insecure Configuration Management

Session Management

If session handling is not provided, then if open the login page, fill the form and hit login. You can see error is generated. This means without session, the system will not permit to access the site.(Fig 9)



Fig9: Session Management

Cookie Modification

Cookies are destroyed at the point the user closes the browser and unless a deletion date has been set. If the user closes the browser without sign out then default deletion date has been set, the cookie will be destroyed on that deletion date instead.(Fig 10)

http://docs.google.com/viewer?a=v&q=cache:mOsA6ra8_ccJ:owasp-top10.googlecode.com/files/OWASP%2520Top%252010%2520%252010.pdf+owasp+top+10+2010+pdf&hl=en&gl=in&pid=bl&srcid=ADGEESj6fIEQYD7H6tuUIKe5EB4ZLpx3FFtsWkqCWdzCbea mMUq3b54xKws3vDFEpij4RtzkzTcrexP9ewzlGaBiSq_ur7IQIGmi Fws_d3ljK44nSmht7uRya5v7mdmMCE6yvJWBCx-&sig=AHIEtbQ1okR1m5cPSvZ1wD4RAPlcC7BuFw

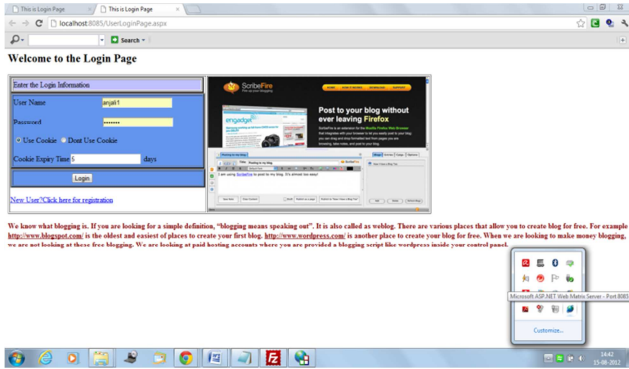


Fig10: Cookie Modification

V. CONCLUSION

There are several website security issues. These issues are common to various types of websites, irrespective of the type of website development language and technology. Analyzing the threats and how that can affect the data and the site itself is an important aspect of web development. Though several techniques and extensions exist in virtually all the technologies, no work presents an interface where both secured and insecure version of the site can be checked.

Therefore in this work we presented a novel technique to demonstrate the web vulnerabilities through live site and prove that simple configuration and coding changes can ensure a highly secured website. Hence we studied various vulnerabilities like Unvalidated Input, Broken Access Control, Broken Authentication and Sessions Management, Insecure Configuration Management, Improper Error Handling, Parameter Modification, Cookie Modification and Directory Traversal. Also methods for detecting those vulnerabilities and successfully implemented security mechanism to all those vulnerabilities to provide protection.

The work can be further improved by demonstrating the other various vulnerabilities like SQL injection, Cross Site Scripting, Buffer Overflow, Denial of Service and providing solution for the same. The objective of this paper is only a starting point for those issues that represent the most serious risks to web application security.

REFERENCES

- [1] Top 10 web security threats part-1 URL: <http://www.emate-econtent.org/security/top-10-web-security-threats-part-1/>
- [2] Top 10 web security threats part-2 URL: <http://www.emate-econtent.org/security/top-10-web-security-threats-part-2/>
- [3] A Survey on Web Application Vulnerabilities (SQLIA, XSS) Exploitation and Security Engine for SQLInjectionCommunication Systems and Network Technologies (CSNT), 2012 International Conference on Date of Conference: 11-13 May 2012 Author (s): Johari, R.; Sharma, P. USIT, GGSIP Univ., Delhi, India
- [4] Security vulnerabilities in modern web browser architecture MIPRO, 2010 Proceedings of the 33rd International Convention Date of Conference: 24-28 May 2010 Author(s): Silic, Marin; Krolo, Jakov ; Delac, Goran Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, 10000, Croatia
- [5] Testing and Comparing Web Vulnerability Scanning Tools for SQL Injection and XSS Attacks Dependable Computing, 2007. PRDC 2007. 13th Pacific Rim International Symposium on Date of Conference: 17-19 Dec. 2007 Author(s): Fonseca, J. CISUC - Polytechnic Inst. of Guardia, Guardia Vieira, M. ; Madeira, H.
- [6] Using web security scanners to detect vulnerabilities in web services Dependable Systems & Networks, 2009. DSN' 09. IEEE/IFIP International Conference on Date of Conference: June 29 2009-July 2 2009 Author(s): Vieira, M.; Antunes, N. ; Madeira, H. Dept. of Inf. Eng., Univ. of Coimbra, Coimbra, Portugal
- [7] OPEN WEB APPLICATION SECURITY PROJECT; OWASP Top 10-2010 PDF URL:

Katkar Anjali S. received her B.E. Degree in Computer Science and Engineering from Walchand Institute of Technology, Shivaji University. She has worked as Lecturer for four years undertaken subjects are Core Java, J2EE and Visual Basics in Computer Technology Department. She is pursuing M.E degree in Computer Science and Engineering from Walchand Institute Of Technology, Sholapur, Maharashtra, INDIA. Her research interest includes web development, web security, web application attacks and its defense.

Kulkarni Raj B. is working as Professor in Computer Science and Engineering Department, Walchand Institute of Technology, Sholapur, Maharashtra, INDIA. He is pursuing Ph.D. in Computer Science and Engineering and has published international journal, National journal, international conference and national conference to his credit. He has taught various subjects such as , Network Engineering, Operating System part I and II, Data warehouse and Data mining, Web Technology etc. at Graduate and Post Graduate Level. He has guided several projects at graduate and post graduate level.