International Journal of Intelligent Computing and Information Science

A PROPOSED LOGICAL FRAMEWORK FOR ENHANCE WEBSITE'S SECURITY FROMTHE ATTACKS

M. Elbialy G. Mostafa N. Elkhamesy

Department of Computer and Information Systems, Sadat Academy for Management Sciences

Mohamed@MohamedElbialy.com ghada5552003@yahoo.com wessasalsol@gmail.com

Abstract:*Security is a major concern for the modern age systems, network, and database administrators. Recently there has been a remarkable interest by both professional and scientific committee about identifying and detecting tacks while also making all possible actions to enhance security. Many models and frameworks are proposed in literature, however few have updated list of actions adapted to types of attacks. This paper presents an effective framework that classifies and detects the different types of attacks along with their symptoms and features. Such a researcher has clearly tested and evaluated a common twelve types of attacks the research has covered and analyzed a survey which spanned over 25 Web developers working with dynamic websites. Numbers of important observation and results were validated which are centered on the weakness of the applied protection mechanisms. The research presents a logical framework a long with guideline criteria that enable fast detection of the common attacks and detective a set of actions that enhance protection and security of dynamic* websites.

Keywords: *Website, Framework, attacks,Hacking,Web server.*

1. Introduction

The most valuable asset of an organization in an information society must be the information. It includes a constant risk of hazard and the greater and more than ever before. This is due to the evolution of the Internet, and leads organizations to share information without enough protect[1].

Rapid advances in network and information security technologies are gradually making the dream of ubiquitous high-speed network access a reality. At the same time, however, such ubiquitous network access allows vandals and criminals to exploit vulnerabilities in networked systems on a widespread basis [2].

All organizations must ensure the implementation ofsecurity practices within their operations to gain customersconfidence and trust and also to protect their privacy andsensitive data of been stolen, sabotage or destroyed accidentally [3].

The rapid growth of internet has created many services, which have become an integral part of our day to today life. Websites are used for making reservations, paying bills, and shopping on-line. With advent of Business-to-Business (B2B) and Business-to-Consumer (B2C) interaction, it is has become a necessity that information be exchanged in a secure and accurate way. Most of the websites contain

security vulnerabilities, which enable hackers to exploit them and launch attacks. As a result of the attacks confidentiality, integrity and availability of information are lost [4].
The history of hacking begins with the rise of the personal computer and the movement of computer resources from controlled laboratory environments to homes of private citizens. The early communities of hackers were small in number consisting mostly of youths trading pirated copies of computer games and exploring ways to manipulate the phone system [5].

The Logical Framework Approach was developed in 1969 for the United States Agency for International Development. The creator of the LFA was Leon J. Rosenberg, as a principal of Fry Consultants, based on worldwide study performed by Rosenberg, Hanley, and Posner [6].
The Logical Framework Approach (LFA) is a management tool mainly used in the design, monitoring and evaluation of international development projects. It is also widely known as Goal Oriented Project Planning (GOPP) or Objectives Oriented Project Planning (OOPP) [6].

2. Types of Website Attacks

The following illustrate different kinds of security vulnerabilities in web applications. Also, include a wealth of real-world examples. Just as application developers can benefit from understanding the methods used by attackers and hackers to detect each type of vulnerabilities.

2.1 Authentication Attack

Authentication is the assurance that the communicating entity is the one that it claims to be [7]. A system can authenticate a user to determine if the user is authorized to perform an electronic transaction or get access to information or a system [8]. Attackers adopt several mechanisms to retrieve passwords stored or transmitted by a computer system to launch this attack [9]. Authentication is a dangerous feature of this process, but even hard authentication mechanisms can be damaged by flawed credential management functions, including password change, forgot my password; remember my password, account update, and other related functions [10]. Authentication cannot protect assets if users do not use them properly [11].

2.2 SQL injection Attack

The term “SQL injection” dates back to 1998, while its first public use was in the year 2000 [12]. The SQL injection attacks pose greater risk due to the fact that they impact databases which are critical to any organization [13]. It occurs when a malicious user modifies the semantic or syntax of a legitimate query by inserting new SQL keywords or operators consequently generating unexpected results not intended by web applications [14].

2.3 Session Management Attack

The Hypertext Transfer Protocol (HTTP) is the basis for today’s World Wide Web (WWW) [15]. Sessions are commonly used in a client - server architecture [16]. Session management vulnerabilities can exist when a web application does not provide a secure mechanism for maintaining a user’s state, both while the user is interacting with the web application, and after the user finishes his session [17]. Remain open after the transmission of a request and its response. Multiple requests can be transmitted over a single TCP connection until client or the server sends the Connection: close message to close the connection [18].

2.4Malicious File Execution Attack

Malicious file execution attacks allow hackers to achieve internal system compromise, perform remote code execution, and install remote root kits [19]. This type of attack occurs when a web application is tricked into including non-approved remote files with malicious code by accepting file names or files from an attacker [20].Malicious file execution attacks affect PHP, XML and any framework which accepts filenames or files from users [21].

2.5Failureto Secure URL Access attack

Failure to restrict URL Access vulnerability usually occurs when unauthorized users are able to access the content of web pages that are only intended to be viewed by users with special privileges, for example administrators [22]. In 2007, the Macworld Conference & Expo web site failed to restrict special URL access to a Steve Jobs keynote speech and let users get "Platinum" passes worth nearly $1,700, all for free [23]. If an application fails to appropriately restrict URL access, security can be compromised through a technique called forced browsing [24].

2.6Cross-Site Scripting – CSS attack

According to OWASP Top 10 - 2010: The Top Ten Most Critical Web Application Security Risks list, Cross-Site Scripting (XSS) is listed as number two [25]. Cross-site scripting (XSS) is an attack against web applications in which scripting code is typically injected into the output of an application that is then sent to a user's web browser [26]. Attacks occur when a script is injected and executed on a victim's browser [27].

2.7 Cross-Site Request Forgery – (CSRF) attack

A CSRF attack forces a logged-on victim's browser to send a pre-authenticated request to a vulnerable web application, which then forces the victim's browser to perform a hostile action to the benefit of the attacker [28]. Many web applications forget that HTTP requests they receive from browsers may have been forged by another web page opened in the same browser [29]. Without the user being aware of it, this malicious web page can take over his identity and send a request to other website on his behalf. This kind of attack is called Cross-Site Request Forgery (CSRF). This name was given by Peter Watkins in a June 2001 [30]. A CSRF can occur on an HTTP request using either the GET or the POST method [31].

2.8 Insecure Communications Attack

Most contemporary web applications collect and store information such as usernames, passwords, social security, account statements, medical history and various other proprietary information. The collected information must be kept in a highly secured storage area [32]. Many web applications do not properly protect sensitive data, such as credit cards, SSNs, and authentication credentials, with appropriate encryption or hashing. Attackers may use this weakly protected data to conduct identity theft, credit card fraud, or other crimes [33].

2.9 Directory Traversal Attack

As OWASP website explains, this category of attacks exploits various path vulnerabilities to access files or directories that are not intended to be accessed[25]. This attack works on applications that take user input and use it in a "path" that is used to access a file system. If the attacker includes special characters that modify the meaning of the path, the application will misbehave and may allow the attacker to access unauthorized resources [34]. Directory traversal exploits use strings like ".. /.. /.. / ".

Most IDSs have signatures to detect this, but attackers replace the " / " with the Unicode equivalent, "%c0%af," and evade the IDS and thus traverse other directories[35].

2.10 InsecurecryptographicStorage Attack

Each web application stores sensitive data when having a login form for users [36]. Many web applications do not properly protect sensitive data, such as credit cards, SSNs, and authentication credentials, with appropriate encryption or hashing. Attackers may steal or modify such weakly protected data to conduct identity theft, credit card fraud, or other crimes [37].
In this type Data and Credential are rarely protected with cryptographic functions because of that Data collected can be used by attackers i.e. Crimes like Credit Card Fraud [38].
Protecting application's data shall be main goal of any project or business that somehow collects information about users [39].

2.11 Information LeakageAndImproper Error Handling Attack

It is a big issue known and understood by many organizations. An error message can give the attacker the information needed for refining the attack [40]. The vulnerability can be remediated through source code analysis. The vulnerabilities consist of: - Discover the web server path on Windows platform; - Read and delete arbitrary files from the host server with the permission of the service account; - Execute external replay attacks [41].

2.12 Buffers overflow attack

A buffer overflow vulnerability occurs when data can be written outside the memory allocated for a buffer. Buffer overflows allow a malicious user to overwrite other pieces of information, such as a return address on the stack, a function pointer, or a data pointer, which may then alter the program's control flow [42]. A non-executable stack would have no effect on this attack [43].

3. Survey Result and analysis:

A researcher has clearly tested and evaluated a common twelve types of attacks the research has covered and analyzed a survey which spanned over 25 Web developers working with dynamic websites. A number of important observation and results were validated, which are centered on the weakness of the applied protection mechanisms and it was reached following results.

Results:

3.1 Table (1): Knowledge regarding the effect of the attacks on websites. This table displays that, there are some types of attacks are unknown to Web developers, which leads to increasing attacks on websites.
3.2 Table (2): Protection regarding the attacks on websites. This table reveals that the percentages of protection ways to websites attacks are very weak for most methods of attacks, which leads to increasing attacks on websites.
3.3 Table (3): Dangerous rate regarding the attacks on websites. This table shows that, the highest percentage of dangerous attacks on websites for Web developers.
3.4 Table (4): Incidence rate regarding the attacks on websites. This table clarifies that, the difference between the incidences of attacks on websites because of the ease of use of certain methods of attacks on other attacks.

Table 1: Knowledge regarding the effect of the attacks on websites

Items	Yes		N0	
	No	%	No	%
1. Authentication attack	23	92	2	8
2. SQL injection attack	19	76	6	24
3. Session Management attack	13	52	12	48
4. Malicious File Execution attack	19	76	6	24
5. Failure to secure URL Access attack	16	64	9	36
6. Cross Site Scripting – CSS attack	21	84	4	16
7. Cross-Site Request Forgery – (CSRF) attack	10	40	15	60
8. Insecure Communications attack	14	56	11	44
9. Directory traversal attack	9	36	16	64
10. Insecure Cryptographic Storage attack	17	68	8	32
11. Information Leakage and Improper Error Handling attack	14	56	11	44
12. Buffer overflow attack	13	52	12	48

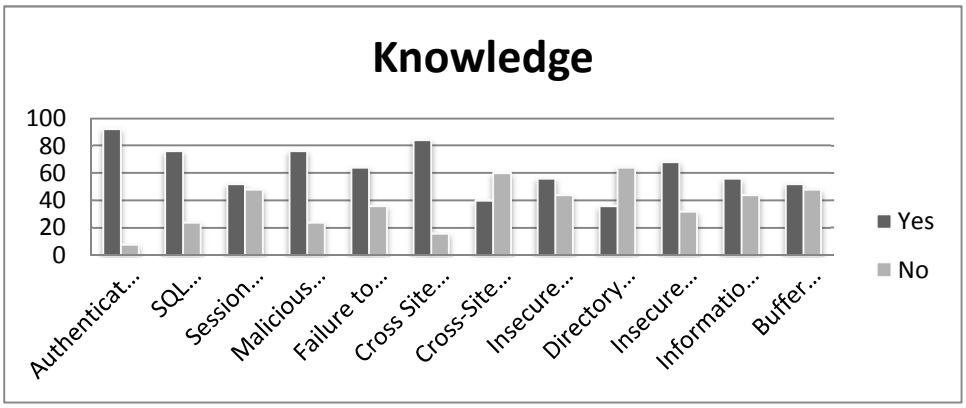

Table 2: Protection regarding the attacks on websites

Items	Yes		N0	
	No	%	No	%
1. Authentication attack	8	32	17	68
2. SQL injection attack	9	36	16	64
3. Session Management attack	9	36	16	64
4. Malicious File Execution attack	7	28	18	72
5. Failure to secure URL Access attack	5	20	20	80
6. Cross Site Scripting – CSS attack	7	28	18	72
7. Cross-Site Request Forgery – (CSRF) attack	8	32	17	68
8. Insecure Communications attack	7	28	18	72
9. Directory traversal attack	3	12	22	88
10. Insecure Cryptographic Storage attack	7	28	18	72
11. Information Leakage and Improper Error Handling attack	5	20	20	80
12. Buffer overflow attack	5	20	20	80

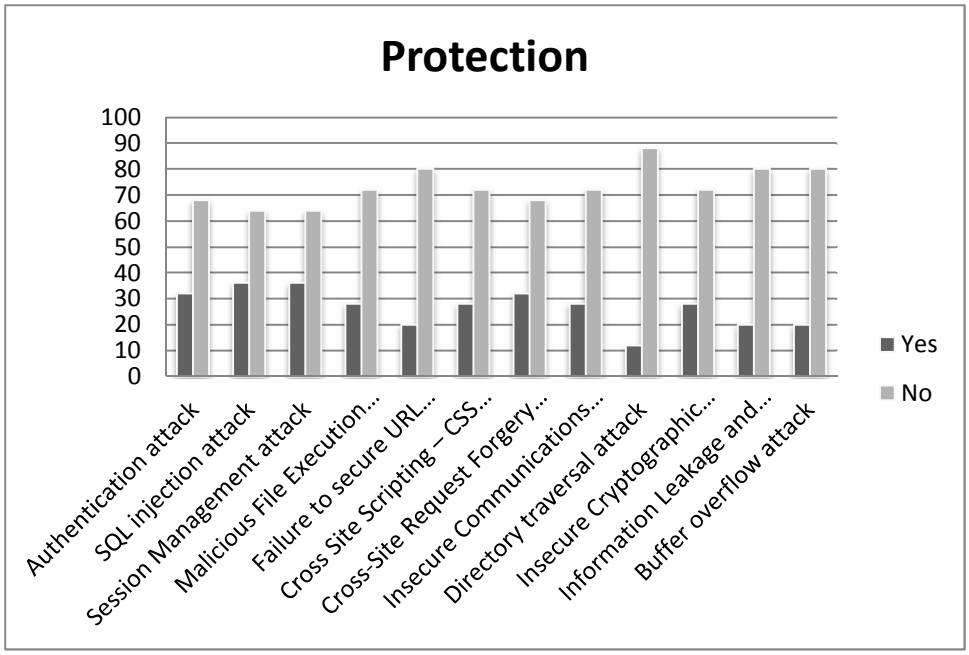

Table 3: Dangerous rate regarding the attacks on websites

Items	Very High rate		High rate		Normal rate		Pass rate		Fair rate	
	No	%	No	%	No	%	No	%	No	%
1. Authentication attack	9	36	0	0	14	56	0	0	2	8
2. SQL injection attack	9	36	3	12	0	0	7	28	6	24
3. Session Management attack	10	40	3	12	10	40	0	0	2	8
4. Malicious File Execution attack	16	64	0	0	7	28	0	0	2	8
5. Failure to secure URL Access attack	23	92	0	0	0	0	0	0	2	8
6. Cross Site Scripting – CSS attack	14	56	6	24	3	12	0	0	2	8
7. Cross-Site Request Forgery – (CSRF) attack	17	68	3	12	0	0	3	12	2	8
8. Insecure Communications attack	16	64	0	0	3	12	4	16	2	8
9. Directory traversal attack	11	44	0	0	0	0	6	24	8	32
10. Insecure Cryptographic Storage attack	14	56	3	12	3	12	3	12	2	8
11. Information Leakage and Improper Error Handling attack	14	56	3	12	3	12	0	0	5	20
12. Buffer overflow attack	11	44	6	24	0	0	6	24	2	8

4. Proposal Logical Framework

The research presents a proposal logical framework a long with guideline criteria that enable fast detection of the common attacks and detective a set of actions that enhance protection and security of dynamic websites

4.1. Proposed Logical Framework

Activity description	Performance Indicators	Means of Verification	Risks and Assumptions
Goal: Secure Websites from types of attacks.	Increase the percentage of securing websites from attacks	Rate of securing websites	Types of the websites
Purpose: Implementation steps to Preventing Vulnerabilities in Websites on the Internet.	decreased rate of hacked websites on the internet	Continuous scan of the websites Vulnerabilities	many of types for attackwebsites methods on the internet
Outputs: Reduce the incidence of attacks websites on the internet	Low rate of the incidence attacks on the websites	Measuring the rate of attack the websites	web developer not aware enough about the methods of protection
Activities: • Backup Sites – often • must be Separated between the web pages web developer and the protection web developer • Control Short URLs • Use own domains for Email • Keep Content on Own Domains • Set Up Malware Alerts • Ensure Domains Have Accurate WHOIS records • Setup Own Domain Expiry Reminders • Secure e-mail address like email address used in website form • Don't leave e-mail addresses anywhere like email use to send emails between all members in forums • Setup firewall • Check for software installed in the web server update	• Steps to prevent the existence of security flaws in websites on the Internet very clearly for web developers	• Web developers knowledge • Websites attacks rate	• The use of experts in the field of websites security • Training The web developer on the website security

5. Conclusions and Future work

In the light of the present study findings, it can be concluded that 25 Web developers who working with dynamic websites who have been working at the departments of websites development in some of government organizations and private companies and included in these study have unknown knowledge regarding some types of attacks, which leads to increasing attacks on websites, and protection ways to websites attacks are very weak for most methods of attacks, which leads to increasing attacks on websites. Meanwhile, highest percentage of the

dangerous rates of the attacks on websites for web developers and the difference between the incidents of attacks on websites because of the ease of use of certain methods of attacks on other attacks.

6. Recommendations

Findings of this study showed the different types of websites attacks and how important of logical framework. Accordingly, the following are the main recommendations deduced by this research:

- Application of the logical framework to reduce the incidence rates of attacks on websites.
- Regular training programs to web developers about the types of attacks on the websites and how to protect from this attacks.

Table 4: Incidence rate regarding the attacks onwebsites

Items	Very High rate		High rate		Normal rate		Pass rate		Fair rate	
	No	%	No	%	No	%	No	%	No	%
1. Authentication attack	6	24	0	0	11	44	6	24	2	8
2. SQL injection attack	3	12	3	12	0	0	10	40	9	36
3. Session Management attack	10	40	0	0	7	28	3	12	5	20
4. Malicious File Execution attack	9	36	4	16	10	40	0	0	2	8
5. Failure to secure URL Access attack	17	68	0	0	0	0	3	12	5	20
6. Cross Site Scripting – CSS attack	13	52	4	16	0	0	0	0	8	32
7. Cross-Site Request Forgery – (CSRF) attack	17	68	0	0	3	12	0	0	5	20
8. Insecure Communications attack	10	40	0	0	0	0	10	40	5	20
9. Directory traversal attack	11	44	3	12	0	0	0	0	11	44
10. Insecure Cryptographic Storage attack	14	56	0	0	6	24	3	12	2	8
11. Information Leakage and Improper Error Handling attack	13	52	4	16	0	0	0	0	8	32
12. Buffer overflow attack	11	44	3	12	0	0	3	12		32

ctivities for each attack:

1. **Authentication attack:**
   - Add random text on the web page presented to the authenticating browser.
   - Prevent of the Password Change Function mistake.
2. **SQL injection attack:**
   - Create stored procedures in database.
   - Replace a single apostrophe with double apostrophes inside the web application code.
   - Create a separate database user account for each website.
   - Reduce the account's privileges in the database.
3. **Session Management attack:**
   - Encrypt data in cookies.
   - Generate Strong characters in cookies.
4. **MaliciousFileExecution attack:**

- Validating user input using an only “accept known” input.
- Adding firewall rules that prevent any external connection.

5. **Failure to secure URL Access attack**:
   - Protection all URLs by an effective access control mechanism.
   - Hack tests before to publish the website to know if it can be only accessed the permitted content.
6. **Cross Site Scripting – CSS attack**:
   - Don't insertuntrustworthy data except in allowed locations.
7. **Cross-Site RequestForgery – (CSRF) attack**:
   - Using a secret cookie.
   - Using POST requests technology.
   - Check the referrer (must be referrer from your own domain).
8. **InsecureCommunicationsattack**:
   - Using Security socket layer (SSL).
   - Encrypt the database server connection.
9. **Directorytraversalattack**:
   - Install the latest version of the web server software.
   - Build a full path to the file/directory if it exists.
10. **InsecureCryptographic Storage attack**:
    - The sensitive data must be encrypt using strong.
    - use approved public algorithms
    - Do not create cryptography algorithms.
11. **InformationLeakage and Improper Error Handling attack**:
    - Using a standard exception handling to prevent information leakage.
12. **Bufferoverflow attack**:
    - Web Developer should be wary of using functions Lead to a buffer overflows.

References

1. CaiCai-qiaoHuo, Li-zhuangMeng and Kai Chen, " Research on the University Network Information Security Risk Management Model Based on the Fuzzy Sets ”, International Conference on Automation - Mechanical Control and Computational Engineering,Baoding University, China , 2015.
2. Xueqi Cheng, JinhongYuan,AliTajer,Aiqun Hu, Wanlei Zhou, " Special issue on recent advances in network and information security”, security and communication networks journal, Volume 8 , USA , 2015.
3. Julius OlusegunOyelami, NorafidaBintiIthnin, " Establishing a Sustainable Information Security Management Policies in Organization: A Guide to Information Security Management Practice”, International Journal of Computer and Information Technology, Volume 4, India, 2015.
4. Mu Qiao, Terry Wyse, " Proposed Color Workflow Solution from Mobile and Website to Printing”, Imaging and Multimedia Analytics in a Web and Mobile World 2015 Conference, Volume 9408, USA, 2015.
5. Gerard Steube, " A logistic regression model to distinguish white hat and black hat hackers”, PhD Thesis, Capella University, USA, 2004.Wikipedia, Website: http://en.wikipedia.org.
6. Ortiz, “C.E. The Wireless Messaging API.2002”, 2007.
7. Available at: http://developers.sun.com/techtopics/mobility/midp/articles/wma/.
8. Steffen GullikstadHallsteinsen, " A study of user authentication using mobile phone ” , Master Thesis, Norwegian University of Science and Technology, Norway, 2007.

9. B.Sumitra*, C.R. Pethuru, M.Misbahuddin , " A Survey of Cloud Authentication Attacks and Solution Approaches " , International Journal of Innovative Research in Computer and Communication Engineering , India, Vol. 2, Issue 10, 2014.
10. Sheo Kumar &KamleshDutta , " Investigation On Security In Lms Moodle ", International Journal of Information Technology and Knowledge Management, Volume 4 , 2011 .
11. Mohamed Ali Al-Fairuz ," An Investigation into the Usability and Acceptability of Multi-channel Authentication to Online Banking Users in Oman ", PhD Thesis, School of Computing Science College of Information and Mathematical Sciences, University of Aberdeen , England, 2011.
12. Xin Lu, Peltsverger, B.,Shijun Chen, Kai Qian,Lixin Tao, " A Static Analysis Framework For Detecting SQL Injection Vulnerabilities", In Proceedings of the 31st Annual International Computer Software and Applications Conference - Volume 01 ,COMPSAC'07,pages87–96,Washington,DC,IEEEComputer Society , USA , 2007.
13. Piyush Mittal, " A Fast and Secure Way to Prevent SQL Injection Attacks using Bit Slice Technique and GPU Support ", Master Thesis, Computer Science and Engineering National Institute of Technology Rourkela, India, 2013.
14. SAli,A.Rauf,Javed.Sqlipa, "An Authentication mechanism against sql injection", European Journal of Scientific Research 38(4), 604–611, 2009.
15. Andreas Mayer, " On the Security of Holder of Key Single Sign On", PhD Thesis , Ruhr-University Bochum, Germany, 2013.
16. FarihaNazmul, " Secure Session Management", Book , Technical University of Denmark Informatics and Mathematical University, Denmark, 2011.
17. David A. Shelly, " Using a Web Server Test Bed to Analyze the Limitations of Web Application Vulnerability Scanners", Master Thesis, Virginia University, USA, 2010.
18. BalachanderKrishnamurphy, Jerry C. Mogul, and David M. Kristol,"Key differences between HTTP/1.0 and HTTP/1.1.", Eighth international conference on World Wide Web, WWW '99, pages 1737{1751, New York, NY, USA, 1999.
19. OWASP, "OWASP Top 10 – 2013 - The Ten Most Critical Web Application Security Risks",2013.
20. A. Shelly, " Using a Web Server Test Bed to Analyze the Limitations of Web Application Vulnerability Scanners", Master Thesis, Virginia University, USA, 2010.
21. Lei Lin , " Intrusion Detection and Prevention Framework for Java Web Applications Based on Aspects and Autonomic Elements ", Master Thesis, Hefei University of Technology, China, 1996.
22. YulianaMartirosyan, " Security Evaluation of Web Application Vulnerability Scanners' Strengths and Limitations Using Custom Web Application", Master Thesis, California State University, 2012.
23. Brodkin, Jon, "The top 10 reasons Web sites get hacked",Network World Journal, 2007.
24. OWASP, "Forced browsing, 2012.Availableathttps://www.owasp.org/index.php/Forced_browsing
25. OWASP, "OWASP top 10 vulnerabilities" 2015.Availableat:http://www.ibm.com/developerworks/library/se-owasptop10/index.html
26. Florian Nentwich, NenadJovanovic, EnginKirda, Christopher Kruegel, Giovanni Vigna, " Cross-Site Scripting Prevention with Dynamic Data Tainting and Static Analysis ", In Proceeding of the Network and Distributed System Security Symposium (NDSS), San Diego, USA , 2007.
27. Michelle Elaine Ruse, "Model checking techniques for vulnerability analysis of Web applications", PhD Thesis, Iowa State University,2013.
28. RamaraoRamisetty , " Heuristics For Preventing Cross Site Request Forgery Attacks", Master Thesis& Engineering –InformationDepartment Of Computer Engineering National Institute Of Technology Karnataka Surathkal, Mangalore,2009.
29. Renaud Feil, Louis Nyffenegger, "Evolution of cross site request forgery attacks", Springer Verlag France, 2007. Availableat :http://link.springer.com/article/10.1007%2Fs11416-007-0068-7.

30. Ziqing Mao, Ninghui Li, Ian Molloy, “Defeating Cross-Site Request Forgery Attacks with Browser-Enforced Authenticity Protection”,Purdue University, USA, 2009.
31. Chuck Willis, “Preparing for the Cross site request forgery defense” Availableat :https://www.blackhat.com/presentations/bh-dc-08/Willis/Whitepaper/bh-dc-08-willis-WP.pdf , 2009.
32. Sabrina Samuel, " Coding Policies for Secure Web Applications”, Master Thesis,Eindhoven University of Technology, Holland, 2007.
33. Lei Lin , " Intrusion Detection and Prevention Framework for Java Web Applications Using Aspects and Autonomic Elements”, Master Thesis , Hefei University of Technology, China, 1996.
34. Carlos BallesterLafuente, " Evaluating Static Analysis Tools for Detecting Buffer Overflows in C Code”, Master Thesis, Norwegian University of Science and Technology, 2007.
35. Jin-Tae Oh , Sang-Kil Park, Jong-Soo Jang, and Yong-HeeJeon, " Detection of DDoS and IDS Evasion Attacks in a High-Speed Networks Environment ”, International Journal of Computer Science and Network Security, VOL.7 No.6, June 2007.
36. Christian Götz, " Vulnerability identification in web applications through static analysis”, Master Thesis,TechnischeUniversitätMünchen, Germany, 2013.
37. Vibhakti Mate "ImplimentationAppraoch For Secure Web Application By Different Prevention Stratergies”, International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 11, November 2014.
38. RakeshkumarKachhadiya,”Development of the Security Framework based on OWASP ESAPI for JSF2.0” , Master Thesis ,University of Freiburg, Germany, 2012.
39. Pilsen, JanFrone " Security in EEG/ERP Portal”, Master Thesis, University of West Bohemia, Czech Republic, 2013.
40. Sheo Kumar &KamleshDutta , " Investigation On Security In Lms Moodle ” , International Journal of Information Technology and Knowledge Management ,, Volume 4,January-June 2011.
41. Marius POPA , " Detection of the Security Vulnerabilities in Web Applications ” , InformaticaEconomica Journal , Romania, vol. 13, 2009 .
42. Cai Kendra June Kratkiewicz, " Evaluating Static Analysis Tools for Detecting Buffer Overflows in C Code”, Master Thesis, Harvard University,USA, 2005.
43. Koziol, J., Litchfield, D., Aitel, D., Anley, C., Eren, S., Mehta, N., Hassell. R., " The Shellcoder's Handbook: Discovering and Exploiting Security Holes” Wiley, 2004.