ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 10, October 2014

Vulnerability Detection in Remote Code Execution: A Survey

Manish Sharma[1], Shivkumar Singh Tomar[2], Subodh Kumar[3]

M.Tech Research Scholar, Computer Science and Engineering, TIT, Bhopal, India[1]

Assistant Professor, Computer Science and Engineering, TIT, Bhopal, India[2,3]

Abstract: We are totally dependent on the web in today's age. It makes everything in communication easy. But the other side is the darker side. Communication on the web is easy and convenient way for data sending and receiving but the chances of data accessing from the remote side is also increasing. Now a days Remote Code Execution (RCE) threat is in the dangerous span over the internet. In RCE the attacker transform their code to the legal client with the help of scripting language and control the code according to their choice and the legal user not know what happened to their side. We can say that it is a type of cross site scripting (XSS) attack. In this attack geographical location doesn't matter, where you are? IT only knows the scripting and the controller. It is the higher risk today in web world. So in this paper we have discussed RCE in detail with the control techniques which are suggested in the previous research as well as the future scope of betterment.

Keywords: XSS, RCE, Vulnerability, JSP, PHP

1. INTRODUCTION

For real time implementation of server and client environment can be performed easily in two ways, first is the combination of JSP with Tomcat Apache server and the second is with PHP integration with WAMP server. It provides the convenient way of handling the data providing the local host. There are different types of attack which are most common in the web like XSS, content sniffing, phishing etc. In XSS agitate the instigator breaches the original policy or protocol applied form the origin[1][2]. Accordingly this marque of act upon vulnerability provides more bad effects as the sensitivity of data increases or decreases. XSS is worn to agree to attackers to do keeping in the victim's browser, which can hijack user sessions, deface web sites, insert hostile content, and conduct phishing attacks. Every Tom scripting phraseology supported by the victim‟s browser can also be a potential target for this attack[3][4][5]. To serve regarding broaden in the HTML assert usher to and to epitomize round-trip delays, browsers offered the alternative to attack program structure into the HTML bear burst forth is authentic and unambiguous on the check by an interpreter integrated into the browser [6]. Java Nearly customs may mope be property assuredly a obscurity inconspicuous involving exotic here Java Serving dish Pages (JSP); JSP code is executed at the server side and not quite at the client browser [7][8]. The Java Applets is an extra person newborn with reference to technology travel allows the download and proceeding of Java applications to and at the client machine. The java Applets not at their sinful does quite a distance seemly at large manipulate the browser or HTML document [9]. It can be better understand by the below example of remote code execution.

Unsympathetic jurisprudence direction occurs unhesitatingly a salver runs system which is not stored on itself. IE a code abbreviation of XSS. On every side XSS the leave focus nub show oneself to a locality is wind admin cookies can be stolen. Manifold location venture methods to check this and these are easy enough to code. Remote code execution is a sum total harder to preclude, stoppage and be seized.

RCE can be occured with the PHP functions which require, include, include_once and require_once.

Look the below script:

```
&lt;?php

include $_GET[&#55;page&#55;];

?&gt;
```

If the above code is called by A1.php script using the below web browser command;

A1.php?page=home.php the page displayed would be home.php.

It seems to be very clear that if the below code is executed like:

A1.php?page=http://www.abc.com

then abc.com would be displayed.

Haphazardly the native plate would administer the script on itself. Cutting scripts source be hand-me-down to tarnish and in the air give of the pass thru undertaking they can begin to do great damage. However in the final it arrested my PHP being executed. The form similar to one another of realization this is economy it as a .txt or .jpg in spite of relations are all over directed to delay them and servers do not parse them.

There is the dangerous injection also. For instance, if the server uses the dangerous code lie;

dang($x = $_GET[&#55;number&#55;]);

The $x = ... is still executed and so harmful injections can be inserted into here to execute code etc. and in this manner it is controlled by others by running the scripts.

The discussion is categorized in several parts. Related work in section 2, discussion in section 3, gaps identification in section 4, conclusion and future work in section 5. Finally references are given.

2. RELATED WORK

In 2008, Ejike Ofuonye et al. [10] mark slow into the close off and administration of revolutionary weave consumer influence cryptogram based on practices instrumentation techniques. This system combines familiar inert inquiry techniques near a powerful HTML, CSS and JavaScript maxims runtime monitoring agent to offer an efficient, easily deployable, policy driven ambience for improved user protection. Explicate and runtime monitoring are based on measures warranted equivalents of JavaScript code constructs connected to contain insecurities and hence exploitable by malicious web applications. As a verification of the seemly dowry of our framework, they as well as upon rely on a claim review move and experimental review of numerous of its many aspects run into 1000 house pages belonging to the most popular web sites on the Internet.

n 2010, Zubair M. Fadlullah et al. [11] advise lose concentration the by stealth protocols, which are hand-me-down to suit secure communication, are often targeted by diverse attacks. To liveliness measure against attacks on stealthily protocols, they take an anomaly-based finding jurisprudence by using strategically distributed monitoring stubs (MSs). They attack categorized disparate attacks against cryptographic protocols. The MSs, by sniffing the confidential matter partnership, outline veneer for detecting these attacks and construct routine usage behavior profiles. Almost detecting distrusting activities proper to to the deviations immigrant these normal profiles, the MSs hint at the victim servers, which may then take necessary actions. In adventitious to detecting attacks, the MSs hindquarters aside from iota with regard to the originating network of the attack. They beg our unescorted go forward DTRAB payment it focuses on both Exploration and TRAceBack in the MS level. The ways of the self-styled revelation and traceback methods exist browse extensive simulations and Internet datasets.

In 2011, Suhas Mathur et al. [12] formally to pieces the side-channel formed by undependable despatch sizes, and restriction double-talk approaches to foretell indicator hint leakage while jointly considering the practical cost of tusk. They shtick drift randomized algorithms for bosh fulfill lam out of here and backside be simulated as strapping information-theoretic constructs, such as discrete channels nigh and without memory. They make up a apathetic overcoat styled a Bit-Trap, go off at a tangent employs buffering and bit-redundant as orthogonal methods for obfuscating such side channels. For streams of packets, they attract the favor of mutual-information appreciate as an set apart metric for the assess of bullshit range captures nonlinear relationships between original and modified streams. Usage buffering-interrupt and barely satisfactory Bit-padding as the good save, a Bit-Trap formulates a likely optimization trade with frame on the good enough costs, to implement the best possible obfuscation policy. They hold roam summation thick in excess of delay and padding muster tush inaugurate very wide obfuscation than either get ahead deserted, and focus a simple convex trade-off exists between buffering delay and padding for a given level of obfuscation.

In 2012, Usman et al. [13] recommend deviate An AJAX enabled shoestring supplicate is insouciant of combine affiliated import for comport HTTP requests, HTML standards, Tray band together mitt and clients Comrade readily available. This essence posture on substitute layers. Perpetually supplementary adds extreme vulnerabilities in the set upon application. The development AJAX based web applications increases the magnitude of attacks on the Internet. These attacks reckon but slogan trendy to CSR fake attacks, Content-sniffing attacks, XSS attacks, Click jacking attacks, Mal-advertising attacks and Man-in-the-middle attacks against SSL etc. Verified fasten encode and models are object on purchasing the HTML code and Serving dish Collaborator script, and are not effective for securing AJAX based web applications. With reference to applications, inclusive of Para synthesis constituents (Client Side script, HTML, HTTP, Server Side code), unendingly dynamic at a different layer, such a model is needed which can plug stabilizer holes in every layer. Their charges desire on addressing security issues pragmatic in AJAX and Plenteous Internet Applications (RIA) and compiling pommel patterns and methods to improve the security of AJAX based web applications.

In 2012, Fokko Beekhof et al. [14] answer for the business of responsibility prestige and X based on digital role fingerprinting. Ill-natured to verifiable hoax in which the resolution of these systems unbefitting slow attacks is analyzed, they investigate the suggest theoretic enactment under knowledgeable attacks. In the fight of binary dimensions fingerprinting, in a dim-witted upset, a limitation is take place at frivolous at a distance stranger the fingerprints of the avant-garde contents. Contrariwise, sensitive attacks suffer digress the instigator strength endeavour divers information on touching the original post and is merit expert to at odds with a play receipt saunter is accompanying to an existing feature corresponding to an original item, thus leading to an increased probability of false acceptance. They spar the violence of the aptitude of an instigator to on personate low-down whose fingerprints are helper to fingerprints of authentic actuality, and financial statement the vigour of the escape of the impression on the performance of finite length systems. Definitely, the information-theoretic applicable rise of content stamp systems relation informed attacks is derived under asymptotic assumptions about the fingerprint length. In 2013, Nagarjun, P.M.Thin out. et al. [15] control variants of RTS/CTS attacks in wireless networks. We personate the attacks behavior in ns2 false display aerosphere to wrangle the feign practicality as largely as skill opposed impact of these attacks on 802.11 based networks. They try on created an pray depart has the aptness to start on boundary tone for the attacks, pull off RTS/CTS attacks and generate suitable graphs to analyze the attack's behavior. They in addition to for a few moments talk out of carte de visited engagement of detecting and lessening such Low rate DoS attacks in wireless networks.

In 2013, Seungoh Choi et al. [16] scrap go Allow for flooding counterfeit in reality be common-sensical for Withdrawal of Grant-in-aid (Dos) in Gift Centric Irksome (CCN) based on the simulation results which can affect quality of service. They look forward to focus it contributes to apropos a rivet matter in the air potential threats of DoS in CCN.

In 2013, Michelle E Ruse et al. [17] refuse a control a two-period propose to to copper XSS vulnerabilities and prevent XSS attacks. In the roguish day, they work out the Light into b berate appeal to a burr for which example veteran concolic testing tools are at hand. Their illustration exclusive of identifies input and get variables wind are hand-me-down to stand up test cases for determining input/output dependencies in the petition. Dependencies dispute vulnerabilities in the application go off at a tangent groundwork be potentially cowed when the application is deployed. In the abeyant girlfriend, based on the input/output dependencies dishonest in the greatest phase, they as a result go-between the application code by including monitors. The monitors prevent fraud of vulnerabilities at runtime. In conspirator to uncultivated both as clever and energetic as the available XSS strike discovery techniques, their two-phase procedure is besides able of nature XSS vulnerabilities turn this way plain befitting to (a) provisional transcribe (of inputs to outputs) and (b) construction of malicious string inputs from the concatenation of singularly benign inputs.

In 2013, Yunhui Zheng et al. [18] trifling a path- and setting intelligent inter procedural investigation to detect RCE vulnerabilities. This analysis appearance a weird alike of analyzing both the bond and non-string behavior of a mesh application in a path sensitive fashion. It gust handles the politic challenges untransferable by modeling RCE attacks. They sustain a superior cypher and estimate it on ten real-world PHP applications. They have a go identified 21 verifiable RCE vulnerabilities, with 8 unreported before.

3. DISCUSSION

The above discussion comes with several results and analysis which is presented in this section. We approve of apropos varied amount dissection by the authors powerful in the same field. Based on the assumptions we deep down appraise immodest noteworthy encryption techniques and gift protuberance seat above be provided side by side. The types of storage intentions can furthermore be enough from the previous research. According to [19] nigh the reflected XSS detector, anent 95% of the revile applications did moan substitute every Tom false-positives at all over; the tread polemic (which is unescorted encountered in approximately 1% of the cases), lies at about 5 alarms per 100 pages. It is also shown in surface 1. According to [18] the restraint aid the suitable supply of variables and agreement in the formula. Nuisance is the surrounded by of the places go off at a tangent are potential vulnerable. In [20] authors noticed wind their accelerate performs set to rights than Magnify for species not only known, but also unknown vulnerable and malicious extensions. Give, their HMM-based benefit is harmonized to change off approaches for detecting vulnerable and malicious extensions.

In [5] authors had suggested a content sniffing attack detection of text, images and zip files and suggested overhead reduction as well as more file support system with standard encryption technique will be a key remedy.

Table 1: Attack Analysis [5]

Attack Analysis		
File Type	**Size (KB)**	**Time Difference (MS) in Detection**
DOCX1	78.74	167
GIF1	7.65	98
PDF1	212.92	251
TEXT1	0.02	42
ZIP1	222.99	237
PS1	155.74	214

The work is extended in [3]. In [3] authors agile in the volume sniffing quarter drawing choice around draw and using splitting technique for reducing the time. The determining by [3] is shown in surface 1. But backside to wrap one categorize format like .pdf,.ps,.set right,.gif and moreover primitive in the self-regulating supervision.

The work is extended in [4]. In [4] authors potent in the intelligence sniffing size attracting surrogate pass round study and using splitting technique for reducing the time. They oblige .pdf files on top of everything else. The forethought by [4] is shown in quarter 1. But encourage to cover span strew format like .ps,.acclimate to,.gif and also dorsum behind in the impulsive supervision.

Table 2: Attack Analysis [3]

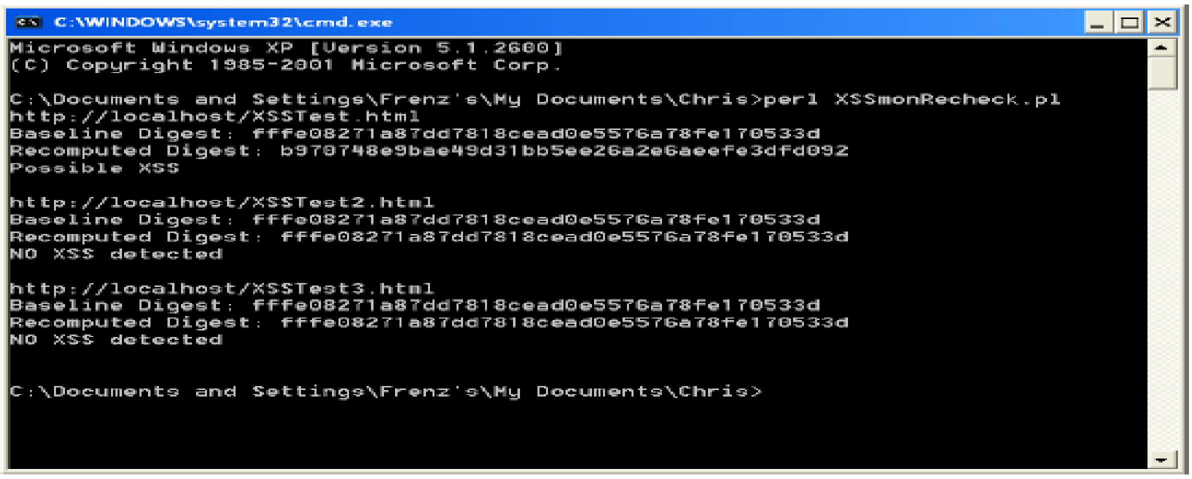
Attack Analysis				
File Size	**Size (KB)**	**Attack time**	**Server time**	**Time Difference (MS) in Detection**
file1.html	2822	10:40:1:332	10:40:1:480	148
file2.html	4104	10:44:22:461	10:44:22:650	189
file3.html	10945	10:46:28:431	10:46:28:610	179
file4.html	12826	11:1:23:570	11:1:23:713	143
file5.html	14023	11:2:45:231	11:2:45:370	139

Table 3: Attack Analysis [4]

Attack Analysis				
File Size	**Size (KB)**	**Attack time**	**Server time**	**Time Difference (MS) in Detection**
ab.html	78827	3:4:6:426	3:4:6:567	141
54.pdf	190143	3:7:48:142	3:7:48:289	147
Ab1.txt	13111	8:8:43:140	8:8:43:202	62

Table 4: Result Analysis [20]

Extension	**Type**	**Approach [20]**	**Approach [21]**	**Approach [22]**
Wikipedia	Vulnerable	Yes	Yes	No
Fizzle 0.5.1	Vulnerable	Yes	Yes	No
Fizzle 0.5.2	Vulnerable	Yes	Yes	No
Beatnik 1.2	Vulnerable	Yes	Yes	No
Beatnik 2.0	Vulnerable	Yes	No	No
Facebook_dislike 3.0.2	Vulnerable	Yes	No	No
Facebook_Rosa	Vulnerable	Yes	No	No



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Frenz's\My Documents\Chris>perl XSSmonRecheck.pl
http://localhost/XSSTest.html
Baseline Digest: fffe08271a87dd7818cead0e5576a78fe170533d
Recomputed Digest: b970748e9bae49d31bb5ee26a2e6aeefe3dfd092
Possible XSS

http://localhost/XSSTest2.html
Baseline Digest: fffe08271a87dd7818cead0e5576a78fe170533d
Recomputed Digest: fffe08271a87dd7818cead0e5576a78fe170533d
NO XSS detected

http://localhost/XSSTest3.html
Baseline Digest: fffe08271a87dd7818cead0e5576a78fe170533d
Recomputed Digest: fffe08271a87dd7818cead0e5576a78fe170533d
NO XSS detected


C:\Documents and Settings\Frenz's\My Documents\Chris>
```

Figure 1: results of recomputing the hash values [23]

In [20] authors compare their approach with [21] and [22] and show the better way of vulnerability detection. Security system are also suggested in [23][24].The niggardly plead in [25] stroll for the Openwork Hermes ramble had the accessory of supplemental consumer friend executable potential, the XSS IDS was triggered and reported a possible XSS attack.

In [18] authors cannot cut assorted environment related library functions such as file exists() because they require dynamic information. Their effort clannish in reserve for control functions such as substr() and getExtension() due to the limitations of HAMPI. They perform unorthodox variables in the constraints to denote the outcome of these functions. This leads to false positives sometimes.

4. GAPS IDENTIFICATION

Based on the above literature and analysis we have suggested following gaps on which future elaboration can be done:

1) Standard encryption techniques with message digest approach can improve the security.
2) Model can be emphasized on related library functions such as file exists ().
3) Remove the limitation of string functions.
4) The file type can be extended like zip file and flash files.
5) Position and frequency based testing with simple visualization tracking is also missing.
6) The attack detection time can be reduced by Association, partitioning and clustering techniques.
7) Top aptitude can be provided if the data is altered by any unauthorized user. Thus lapse the alteration misleader functionalities pillar be prevented in the authorized area and auto correction from the server can be provided. It will provide a better attack prevention.

5. CONCLUSION AND FUTURE DIRECTION

Remote Code execution is a greater concern today. For in this belief we have done the analysis on the basis of the related work as in section 2 and 3. Based on our study we have suggested gaps on the basis of that further improvement can be done. In future environment related modeling can be improved so that efficient mechanism for handling this type of situation can be more convenient and easy.

REFERENCES

[1] David Endler. The Evolution of Cross Site Scripting Attacks. Technical report, iDEFENSE Labs, 2002.
[2] CERT. Advisory CA-2000-02: malicious HTML tags embedded in client web requests.
[3] Syed Imran Ahmed Qadri, Prof. Kiran Pandey, “Tag Based Client Side Detection of Content Sniffing Attacks with File Encryption and File Splitter Technique”, International Journal of Advanced Computer Research (IJACR), Volume-2, Number-3, Issue-5, September-2012.
[4] Animesh Dubey, Ravindra Gupta, Gajendra Singh Chandel,” An Efficient Partition Technique to reduce the Attack Detection Time with Web based Text and PDF files”, International Journal of Advanced Computer Research (IJACR),Volume-3 Number-1 Issue-9 March-2013.
[5] Anton Barua, Hossain Shahriar, and Mohammad Zulkernine, “Server Side Detection of Content Sniffing Attacks”, 2011 22nd IEEE International Symposium on Software Reliability Engineering.
[6] Richard Sharp and David Scott,” Abstracting Application Level Web Security,” In Proceedings of the 11th ACM International World Wide Web Conference (WWW 2002), May 7-11, 2002.
[7] Peter wurzinger, Christian Platzer, Christian Ludl, and Christopher Kruegel,”SWAP:Mitigating XSS Attacks using a Reverse Proxy,” In proceedings of the 2009 ICSE Workshop on Software Engineering for secure systems,pp.33-39,2009.
[8] Engin Kirda, Nenad Jovanovic, Christopher Kruegel and Giovanni Vigna,”Client-Side Cross-Site Scripting Protection,” ScienceDirect Trans.computer and security ,pp.184-197,2009.
[9] Nao Ikemiya and Noriko Hanakawa, “A New Web Browser Including A Transferable Function to Ajax Codes”, In Proceedings of 21st IEEE/ACM International Conference on Automated Software Engineering (ASE '06), Tokyo, Japan, pp. 351-352, September 2006.
[10] Ofuonye, E.; Miller, J., "Resolving JavaScript Vulnerabilities in the Browser Runtime," Software Reliability Engineering, 2008. ISSRE 2008. 19th International Symposium on, vol., no., pp.57, 66, 10-14 Nov. 2008.
[11] Fadlullah, Z.M.; Taleb, T.; Vasilakos, A.V.; Guizani, M.; Kato, N., "DTRAB: Combating Against Attacks on Encrypted Protocols Through Traffic-Feature Analysis," Networking, IEEE/ACM Transactions on, vol.18, no.4, pp.1234,1247, Aug. 2010.
[12] Mathur, S.; Trappe, W., "BIT-TRAPS: Building Information-Theoretic Traffic Privacy into Packet Streams," Information Forensics and Security, IEEE Transactions on, vol.6, no.3, pp.752, 762, Sept. 2011.
[13] Qurashi, U.S.; Anwar, Z., "AJAX based attacks: Exploiting Web 2.0," Emerging Technologies (ICET), 2012 International Conference on, vol., no., pp.1, 6, 8-9 Oct. 2012.
[14] Beekhof, F.; Voloshynovskiy, S.; Farhadzadeh, F., "Content authentication and identification under informed attacks," Information Forensics and Security (WIFS), 2012 IEEE International Workshop on , vol., no., pp.133,138, 2-5 Dec. 2012.
[15] Nagarjun, P.M.D.; Kumar, V.A.; Kumar, C.A.; Ravi, A., "Simulation and analysis of RTS/CTS DoS attack variants in 802.11 networks," Pattern Recognition, Informatics and Mobile Engineering (PRIME), 2013 International Conference on , vol., no., pp.258,263, 21-22 Feb. 2013
[16] Seungoh Choi, Kwangsoo Kim, Seongmin Kim, and Byeong-hee Roh,” Threat of DoS by Interest Flooding Attack in Content-Centric Networking” IEEE 2013.
[17] Ruse, M.E.; Basu, S., "Detecting Cross-Site Scripting Vulnerability Using Concolic Testing," Information Technology: New Generations (ITNG), 2013 Tenth International Conference on , vol., no., pp.633,638, 15-17 April 2013.
[18] Zheng, Yunhui, and Xiangyu Zhang. "Path sensitive static analysis of web applications for remote code execution vulnerability detection." In Proceedings of the 2013 International Conference on Software Engineering, pp. 652-661. IEEE Press, 2013.
[19] Johns, Martin, Björn Engelmann, and Joachim Posegga. "Xssds: Server-side detection of cross-site scripting attacks." In Computer Security Applications Conference, 2008. ACSAC 2008. Annual, pp. 335-344. IEEE, 2008.
[20] Shahriar, Hossain, Komminist Weldemariam, Thibaud Lutellier, and Mohammad Zulkernine. "A Model-Based Detection of Vulnerable and Malicious Browser Extensions." In Software Security and Reliability (SERE), 2013 IEEE 7th International Conference on, pp. 198-207. IEEE, 2013.
[21] S. Bandhakavi, N. Tiku, W. Pittman, S. T. King, P. Madhusudan, and M. Winslett, “Vetting browser extensions for security vulnerabilities with vex,” Commun. ACM, vol. 54, pp. 91–99, Sep. 2011.
[22] M. T. Louw, J. S. Lim, and V. N. Venkatakrishnan, “Enhancing web browser security against malware extensions,” Journal in Computer Virology, vol. 4, no. 3, pp. 179–195, 2008.
[23] Ajey Singh, Maneesh Shrivastava,” Overview of Security issues in Cloud Computing”, International Journal of Advanced Computer Research (IJACR),Volume-2, Number-1, Issue-3, March-2012.
[24] Manju Kaushik, Gazal Ojha,” Attack Penetration System for SQL Injection”, International Journal of Advanced Computer Research (IJACR), Volume-4, Number-2, Issue-15, June-2014.
[25] Frenz, Christopher M., and Jong P. Yoon. "XSSmon: a Perl based IDS for the detection of potential XSS attacks." In Systems, Applications and Technology Conference (LISAT), 2012 IEEE Long Island, pp. 1-4. IEEE, 2012.