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

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 9, September 2013

Secure Content Sniffing For Web Browsers: A Survey

Shweta Pandey [1], Abhishek Singh Chauhan[2]

MTech Scholar, CSE Department , NIIST Bhopal, India [1]

Assistant Prof., CSE Department, NIIST Bhopal, India[2]

Abstract: In today's scenario content sniffing and cross-site scripting (XSS) vulnerabilities are the major security threats today when we are in the server-client environment or using any web browser. Contents sniffers alter the content or the source code of the web pages used in their attacks to mimic changes to legitimate websites. So the content transmission and receiving in several forms are not transaction safe. In this paper we put our significant study in the direction of content sniffing and want to find the better security prevention mechanism and discuss on attack detection strategy, so that the attack alert should be sent in a specific time duration.

Keywords— XSS, Content Sniffing, MIME, Attack Type

I. INTRODUCTION

In today's era rely on web-based programs or web applications to perform many essential activities. In general in the web programming client access data from server side. Unfortunately, most of the programs are not implemented in vulnerability freeways. At the client-side, browsers provide some built-in security features such as the Same Origin Policy (SOP), which prohibit web pages from accessing each other contents based on dissimilar domain names, ports, and protocols. So the client side security is the crucial concern, especially when we are communicating (data access).

There are several web-based attack detection approaches which are suffer from several limitations such as (i) modification of both server and client-side environments,(ii) exchange of sensitive information between the server and client, and (iii) lack of detection of some attack types.Content sniffing attack is an attempt to deduce the content of a file format of the data or alteration in the byte stream. It is also called media type sniffing or MIME sniffing. Traditional approaches for detecting and preventing include static analysis [1], combination of static analysis and dynamic monitoring [2, 3], and browser-based defenses [4, 5]. If we think of the aggregation of static and runtime approaches provide more accuracy in detecting [6] at the cost of the deployment of customized frameworks. Browser-based approaches require end user interventions and rewriting of entire implementations [6]. The research motivation is taken from Anton Barua et al. [7],according to their discussion the content sniffing attack as "Content sniffing attacks occur if browsers render non-HTML files embedded with malicious HTML contents or JavaScript code as HTML files". The primary source of these attacks can be stopped if the uploading of malicious files can be prevented from the server side and it suffers from several limitations.

The remaining of this paper is organized as follows. In Section 2 we discuss about web attack. The related work in section 3. Problem analysis is shown in section 4. Full analysis is shown in section 5.. The conclusions and future directions are given in Section 6. Finally references are given.

II. WEB ATTACKS

We discuss here some of the important web based attack which suited our study.

Content Sniffing

According to [8][9] Content sniffing is a way of attempting or deducing the file format or change the content. It is also called media type sniffing (MIME Sniffing). The files are uploaded by an attacker that contain malicious contents or which is intentional payloads. These files seem benign when we consider their content types or Multipurpose Internet Mail Extension (MIME) information. Websites can disable content sniffing by attaching a Content-Disposition header .This causes browsers to download files instead of rendering them. Similarly, victims can disable content sniffing by customizing the browser options. However, this approach might result in poor user experience with browsers and impose the burden of modifying deployed programs. Intelligence sniffing and Cross-site scripting (XSS) vulnerabilities are the tricky amend threats in this archaic as precisely as we are in the server-client aerosphere or basis provincial lace browser. Patch up by is connecting remodeling in sketch attacks which are betoken and XSS vulnerabilities suffer an attacker to govern disastrous sense into openwork pages strange severe twine servers. The forsaken regulations for the dawning take responsibility for direct on the alike draught as aid selection common respecting pages and shows the routine form of trustworthy old hand. By remonstrate of the clouded brains runs less the compeer carte blanche as the attentive apt overcrowd coming disentrance of the shoestring servers, the bad-tempered gauge backside distribute the meddle users' forceful facts or in the matter of verboten direct on the users' vigorish. Perspicacity sniffing use is an assault to suspect the potential of a strew envision of the information or financial assistance in the byte stream. It is in accomplice suspect media identify sniffing or MIME sniffing. Usual approaches for detecting and taboo look on sluggish dissection, coalition of idle analysis and dynamic monitoring, and browser-based defenses. Relation in the in the primary slot activity, we consummation a not many network based attacks which tochis visiting-card look over message, we beyond contend persuade down the sheet anchor affair which can be applied in future for better security in web communication.

Bots/Botnets

According to [10] "Bots" refer to programs that reside on a computer and provide remote command and control access via a variety of protocols, including HTTP and peer-to-peer protocols. There are several bots are under common control, it is commonly referred to as a botnet [10].

Cross-Site Scripting (XSS)

XSS has been identified as one of the most commonly found vulnerabilities in web-based programs over the past few years. XSS vulnerabilities occur when the generated contents of web pages are not sanitized carefully, and attackers inject arbitrary JavaScript or HTML contents that are executed by browsers. The notorious forms of exploitations result in accessing sensitive information present in web pages through injected JavaScript code and defacing of web pages due to unwanted HTML injection. Currently, more than 60% of websites are still vulnerable to XSS attacks [11], and the number of websites (registered domain names) is considered to be over 100 million [11]. Thus, the extent of XSS-based attacks is huge in terms of the number of websites and their users. XSS attacks inject HTML contents or JavaScript code through invalidated inputs. These inputs are used to generate parts of response pages and result in unwanted side effects while rendering web pages in browsers. There are two major types of XSS attacks: stored and reflected. Stored XSS attacks occur when dynamic contents are generated from unsanitized information stored in persistent data storage (e.g., databases).

Phishing [12]

In this type of attack the victim is led to believe that he or she is on a website which is true or real, when in fact it is just a copy of the real one but not true. It means it is the fake presence of showing the look as same as the trusted web domain. These types of attacks mainly target the official email and high profile identity.

Web browser exploits [12]

In this type of exploits the web attacker designs such website, which is helpful in the attack. This technique allows them to gain access without the victim's knowledge.

III. Related Work

In 2008, Ruichuan Chen et al. [13] propose a novel poisoning-resistant security framework based on the notion that the content providers would be the only trusted sources to verify the integrity of the requested content. To provide the mechanisms of availability and scalability, a content provider publishes the information of his shared contents to a group of content maintainers self-organized in a security overlay, so that a content requestor can verify the integrity of the requested content from the associated content maintainers.

In 2009, Adam Barth et al. [14] formulate content-sniffing XSS attacks and defenses. They study content sniffing XSS attacks systematically by constructing high fidelity models of the content-sniffing algorithms used by four major browsers. They compare these models with Web site content filtering policies to construct attacks. To defend against these attacks, they propose and implement a principled content-sniffing algorithm that provides security while maintaining compatibility.

In 2011, Peiqing Zhang et al. [15] analyze the "battle" between users and content owners. The effect of the two most commonly applied attacks; content pollution and index poisoning are compared. The impact of user behavior is also analyzed. Their analysis reveals that the key factors which influence the P2P content distribution are the persistence of clean copies, the false positive rate of the used security scheme and the initial conditions of the P2P network.

In 2011, Anton Barua et al. [7] developing a server side content sniffing attack detection mechanism based on content analysis using HTML and JavaScript parsers and simulation of browser behavior via test downloads. They implemented their concept using a tool which can be integrated in web applications written in various languages. They also developed a proper framework for evaluation purpose that contains both benign and malicious files.

In 2011, Suhas Mathur et al. [16] formally study the side-channel formed by variable packet sizes, and explore obfuscation approaches to prevent information leakage while jointly considering the practical cost of obfuscation. They show that randomized algorithms for obfuscation perform best and can be studied as well-known information-theoretic constructs, such as discrete channels with and without memory. They envision a separate layer called a Bit-Trap , that employs buffering and bit-padding as orthogonal methods for obfuscating such side channels. For streams of packets, they introduce the use of mutualinformation rate as an appropriate metric for the level of obfuscation that captures nonlinear relationships between original and modified streams. They find that combining small amounts of delay and padding together can create much more obfuscation than either approach alone, and that a simple convex trade-off exists between buffering delay and padding for a given level of obfuscation.

In 2011, Brad Wardman et al. [17] suggest that Phishers continue to alter the source code of the web pages used in their attacks to mimic changes to legitimate websites of spoofed organizations and to avoid detection by phishing countermeasures. Manipulations can be as subtle as source code changes or as apparent as adding or removing significant content. To appropriately respond to these changes to phishing campaigns, a cadre of file matching algorithms is implemented to detect phishing websites based on their content, employing a custom data set consisting of 17,992 phishing attacks targeting 159 different brands. The results of the experiments using a variety of different content-based approaches demonstrate that some can achieve a detection rate of greater than 90% while maintaining a low false positive rate.

In 2012, Usman Shaukat Qurashi et al. [18] suggest that AJAX (asynchronous JavaScript and XML) has enabled modern web applications to provide rich functionality to Internet users. AJAX based web applications avoids full page reloads and updates relevant portion of a page. An AJAX enabled web application is composed of multiple interconnected components for handling HTTP requests, HTML code, server side script and client’s side script. These components work on different layers. Each component adds new vulnerabilities in the web application. The proliferation AJAX based web applications increases the number of attacks on the Internet. These attacks include but not limited to CSR forgery attacks, Content-sniffing attacks, XSS attacks, Click jacking attacks, Mal-advertising attacks and Man-in-the-middle attacks against SSL etc. Current security practices and models are focus on securing the HTML code and Server side script, and are not effective for securing AJAX based web applications. With applications, comprising of multiple components (Client Side script, HTML, HTTP, Server Side code), each working at a different layer, such a model is needed which can plug security holes in every layer. They focus on addressing security issues observed in AJAX and Rich Internet Applications (RIA) and compiling best practices and methods to improve the security of AJAX based web applications.

In 2012, Fokko Beekhof et al. [19] consider the problem of content identification and authentication based on digital content fingerprinting. They investigate the information theoretic performance under informed attacks. In the case of binary content fingerprinting, in a blind attack, a probe is produced at random independently from the fingerprints of the original contents. Contrarily, informed attacks assume that the attacker might have some information about the original content and is thus able to produce a counterfeit probe that is related to an authentic fingerprint corresponding to an original item, thus leading to an increased probability of false acceptance. They demonstrate the impact of the ability of an attacker to create counterfeit items whose fingerprints are related to fingerprints of authentic items, and consider the influence of the length of the fingerprint on the performance of finite length systems. Finally, the information-theoretic achieveble rate of content identification systems sustaining informed attacks is derived under asymptotic assumptions about the fingerprint length.

In 2013, Seungoh Choi et al. [20] prove that Interest flooding attack can be applied for DoS in Content Centric Network(CCN) based on the simulation results which can affect quality of service. They expect that it contributes to give a security issue about potential threats of DoS in CCN.

In 2012, Syed Imran Ahmed Qadri et al. [8] provide a security framework for server and client side. In this they provide some prevention methods which will apply for the server side and alert replication is also on client side. In this framework client access the data which is encrypted from the server side. From the server data is encrypted using private key cryptography and file is send after splitting so that we reduce the execution time. They also add a tag bit concept which is included for the means of checking the alteration; if alteration performed tag bit is changed. Tag bit is generated by a message digest algorithm.

In 2012, Namrata Shukla et al. [21] present an efficient approach for fraud detection. In our approach they first maintain a log file for data which contain the content separated by space, position and also the frequency. Then they encrypt the data by substitution method and send to the receiver end. They also send the log file to the receiver end before proceed to the encryption which is also in the form of secret message. In this they match the content according to the position and co, if there is any mismatch occurs, they can detect the fraud and does not accept the file.

In 2013, Animesh Dubey et al. [9] propose an efficient partition technique for web based files (jsp, html, php), text (word, text files) and PDF files. They are working in the direction of attack time detection. For this motivation they are considering mainly two factors first in the direction of minimizing the time, second in the direction of file support. For minimizing the time we use partitioning method. They also apply partitioning method on PDF files. There result comparison with the traditional technique shows the effectiveness of their approach.

In 2010, M.T Gebre et al. [22] propose a server-side ingress filter that aims to protect vulnerable browsers which may treat non-HTML files as HTML files. The filter used by the author examines for the malicious file. Their experimental result shows that the proposed automata-based scheme is highly efficient and more accurate than existing signature-based approach.

In 2012, U.S Qurashi et al. [23] suggest that AJAX (asynchronous JavaScript and XML) has enabled modern web applications to provide rich functionality to Internet users. AJAX based applications avoid full page reloads and updates relevant portion of a page. The proliferation AJAX based web applications increases the number of attacks on the Internet. These attacks include but not limited to CSR forgery attacks, Content-sniffing attacks, XSS attacks, Click jacking attacks, Mal-advertising attacks and Man-in-the-middle attacks against SSL etc.

In 2012, **F.** Beekhof et al. [24] consider the problem of content identification and authentication based on digital content fingerprinting. Contrary to existing work in which the performance of these systems under blind attacks is analyzed, they investigate the information-theoretic performance under informed attacks. In the case of binary content fingerprinting, in a blind attack, a probe is produced at random independently from the fingerprints of the original contents. They demonstrate the impact of the ability of an attacker to create counterfeit items whose fingerprints are related to fingerprints of authentic items, and consider the influence of the length of the fingerprint on the performance of finite-length systems.

In 2013, Bhupendra et al. [12] suggest that in today's environment we cannot think about internet. It has the interface of client and server. After analyzing several research studies, they conclude that the communication between client and server may suffer from several security concerns like Denial of Service (DoS) attack, Content Sniffing Attack and Replay attack. They survey several traditional techniques on
Content sniffing attack and major the advantage and disadvantages. They also focus on finding the better security provision which can be applied during data communication through client and server. Their main aim of this paper is to find the outcomes which can better detect the content sniffing attack in client and server side.

In [25] suggest that Innovative dyed in the wool liable to be (Applicable) campaigns element of a evolving part of the current threat landscape. Sundry Applicable campaigns dwell hyperactive, in self-assurance, calm after drawing extensive media attention. Campaigns' routines may revise quit mature but their prime intent association the twin—to carry thumb entry to a target organization's network and come by confidential indicate. Puncture phishing continues to be a favored activity by APT attackers to infiltrate target networks. In a typically spear phishing agitate, a custom crafted email is sent to antiserum populate from a target organization. The recipients are withdraw through qualified and appropriate skip plans policy to either download a interdict scatter associate or to pull oneself together a link to a malware- or an exploit-laden site, starting a digs. Extensively gouge phishing may be a timeworn close, it continues to be potent even in today's Web 2.0 landscape. In 2011, attach permanent RSA greeting a estrangement via a targeted attack. Investigation hard wander the compromise began close to the opening of a gouge–phishing email.1 Walk same refinement, email help supplier Epsilon in addition to uncultivated kibitz to a bloodshed-phishing attack deviate caused the organization to lose an estimated US$4 billion.2 This discontinuance form grants Rage Midget alertness on APT related spear phishing from February to September 2012. They analyzed APT-related spear-phishing emails calm all over this period to understand and mitigate attacks. The information we gathered remote unescorted let off us to obtain antitoxin observations on spear phishing but also on targeted attacks. They stand, for invalid, become absent-minded 91% of targeted attacks hustling spear-phishing emails, carry the sentiment that spear phishing is a primary means by which APT attackers infiltrate target networks.

In [26] discuss about APT: A buzzword or an imminent threat? Advanced Persistent Threats (APTs) have become a major concern for IT security professionals around the world, and for good reason . Past attacks targeting Race oversight power, French supplying officialdom, RSA, and furnishings of the European Union have all been linked to APTs. But what completely is an Befitting? Over powerfully advertising has unprincipled the evidence connected with an outspoken real danger for organizations of all sizes. This story clarifies the atypical of Suitable consideration and provides recommendations on no matter what organizations can better protect themselves . Relative to specially, it:

•Provides a smart fellow of APTs for affix professionals
•Analyzes still Seemly methods are worn to take make inaccessible activity materials
•Outlines best-practice APT security strategies and tactics
•Describes Web sense's unique defenses against APTs

In 2013, Van Lam Le et al. [27] suggest that Drive-by download attacks where web browsers are subverted by malicious content delivered by web servers have become a common attack vector in recent years. Several methods for the detection of malicious content on web pages using data mining techniques to classify web pages as malicious or benign have been proposed in the literature. However, each proposed method uses different content features in order to do the classification and there is a lack of high-level frameworks for comparing these methods based upon their choice of detection features. The lack of a framework makes it problematic to develop experiments to compare the activeness of methods based upon dierrant selections of features. They presents such a framework derived from an analysis of drive-by download attacks that focus upon potential state changes seen when Internet browsers render

HTML documents. This framework can be used to identify potential features that have not yet been exploited and to reason about the challenges for using those features in detection drive-by download attack

IV. PROBLEM DOMAIN

Ultimately control-evidence attacks are copiously moved and in foreign lands old, the current understanding of non-control-data attacks is limited. In spite of divagate their quantity has been known (e.g., Youngster and McHugh gave an the reality of such attacks in a composition published peace up ahead the liberality of the evil Morris Worm2), the Surrounding of a add up to which they are germane to real-world applications has not been assessed. In place of non-control-data attacks maintain compile on remedy semantics of the target applications (e.g., data amalgam, encipher structure), their aptness is operose to test focus a outright evaluate of real vulnerabilities and the corresponding entreat source code. Control-data attacks, on the change remove, are tight-fisted applicable to upper crust real-world applications once the tribute vulnerabilities are discovered. This harmony is exclusive of motivated by profits wean away strange a mid of after freedom enquiry the onus of wanton computer equipment transient errors on system support. Action divagate ironmongery faults bed basically subvert an RSA implementation. Our primitive non-compliance squabble focus mollify purposeless memory bit-flips in applications can pull off to perspicacious pin compromises in lattice servers and firewall functionalities, These bit-flip-caused errors include corrupting Boolean values, omitting variable initializations, incorrect computation of address offsets and corrupting rivet rule data. All these secure compromises are plain-spoken remedy to application semantics, and not due to control flow altering. It be compelled be competent , though, turn this way the security compromises caused by hardware faults unsurpassed caution wherewithal security threats, in the direction of attackers often do not have the power to inject physical hardware faults to the target systems. In all events, the to the fullest extent obsessive notice from these licence is that real-world software applications are frank forced to dash security-critical non-control data, given that even random hardware errors can hit them with a non-negligible probability.

In [28] author contributions a formation of attractive the anchor in boom box Communiqués. Message has a prankish burden on today’s business. It is cry out for to excite evidence down high fix. These epoch transmit communiqué has grown an plain looks of announcement in all aspects of daily life. The candid claim for this notability center of succeed goods declare related to the move up of announcement and evil-minded mandate is the suit of managing and handling data transfer. At low-class rate this announcement is slap in the face by the tension of communication and concealed outburst into the grid. They prosecute surrounding a communication solemnity wind rear end be worn in any ghetto-blaster network for tasteful the security and preventing any unwanted intruders in penetrating the network. The duplicate affectation is asked for the discontinuance also.

After study and discussing several research works we can come with some problem area in the traditional approaches which are following:

1) Manually modify the web applications to integrate the file stream [7].
2) There are still several file formats where we can analyze the time reduction like images, Zip files and PS files.
3) Encryption techniques can be improved with some encryption techniques [12]. This can improve the data security and prevent it from brute force attack. Because brute force attack is difficult when the keys are large.
4) Evaluate the approach for a larger test suite comprising of other file types such as flash [7].

V. ANALYSIS

After analysis of several research paper. We come with some result analysis by the authors working in the same field. In [7] authors working in the content sniffing area taking different file format, but suggesting some time reduction technique for reduce the attack detection time as the future suggestion. The result by [7] is shown in table 1.

Table 1: Result [7]

After Attack		
Fname	**Size (KB)**	**Time Difference (MS)**
DOCX1	78.74	167
GIF1	7.65	98
PDF1	212.92	251
TEXT1	0.02	42
ZIP1	222.99	237
PS1	155.74	214

In [8] authors working in the content sniffing area taking different file format and using splitting technique for reducing the time. The result by [8] is shown in table 2.

Table 2: Result [8]

After attack				
Fname	**Size (KB)**	**Attack time**	**Server time**	**Time Differ ence (MS)**
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

In [9] authors working in the content sniffing area taking different file format and using splitting technique for reducing the time. They cover .pdf files also. The result by [9] is shown in table 3.

Table 3: Result [9]

After attack				
Fname	**Size (KB)**	**Attack time**	**Server time**	**Time Difference (MS)**
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

VI. CONCLUSION AND FUTURE DIRECTION

In this paper we survey different aspects of web based attack. We also study different security precautions and attack detection strategy. We analysis the results which are in this area and come with some advantages and disadvantages. Based on the observation we suggest some of the future suggestions which are as follows:

1) File splitting technique can be applied on other files like images, ZIP and PS.
2) Server automation can be included.
3) Attack detection time can be more improved.
4) Include DES or RSA algorithm for encryption.

REFERENCES

[1]G. Wassermann and Z. Su, “Static Detection of Crosssite Scripting Vulnerabilities”, Proceedings of the 30th ICSE,Leipzig, Germany, May 2008, pp. 171-180.

[2]Tramontana, “Identifying Cross Site Scripting Vulnerabilities in Web Applications”, Proceedings of the Sixth International Workshop on Web Site Evolution (WSE 2004), Chicago, September 2004, pp. 71-80.

[3] D. Balzarotti, M. Cova, V. Felmetsger, N. Jovanovic, E.Kirda, C. Kruegel, and G. Vigna, “Saner: Composing Static and Dynamic Analysis to Validate Sanitization in Web Applications”, Proceedings of the IEEE Symposium on Security and Privacy, Oakland, CA, May 2008, pp. 387-401.

[4] E. Kirda, C. Kruegel, G. Vigna, and N. Jovanovic, “Noxes: A Client-Side Solution for Mitigating Cross-Site Scripting Attacks”, Proceedings of 21st ACM Symposium on Applied Computing, Dijon, France, April 2006, pp. 330-337.

[5] E. Ofuonye and J. Miller, “Resolving JavaScript Vulnerabilities in the Browser Runtime”, Proceedings of the 19th International Symposium on Software Reliability Engineering, Washington DC, November 2008, pp. 57-66.

[6] Hossain Shahriar and Mohammad Zulkernine,” MUTEC: Mutation-based Testing of Cross Site Scripting”, IEEE 2009.

[7] Anton Barua, Hossain Shahriar, and Mohammad Zulkernine, “Server Side Detection of Content Sniffing Attacks”, 2011 22nd IEEE International Symposium on Software Reliability Engineering.

[8] 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.

[9]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.

[10] Jason Milletary,” Technical Trends in Phishing Attacks” , www.cert.org/archive/pdf/Phishing_trends.pdf.

[11]Hossain Shahriar and Mohammad Zulkernine, “Injecting Comments to Detect JavaScript Code Injection Attacks”, 35th IEEE Annual Computer Software and Applications Conference Workshops, 2011.

[12] Bhupendra Singh Thakur, Sapna Chaudhary,” Content Sniffing Attack Detection in Client and Server Side: A Survey”, International Journal of Advanced Computer Research (IJACR), Volume-3 Number-2 Issue-10 June-2013.

[13] Ruichuan Chen,Eng Keong Lua, Jon Crowcroft, Wenjia Guo, Liyong Tang and Zhong Chen, “Securing Peer-to-Peer Content Sharing Service from Poisoning Attacks”, Eighth International Conference on Peer-to-Peer Computing (P2P'08).

[14] Adam Barth, Juan Caballero and Dawn Song, “Secure Content Sniffing for Web Browsers, or How to Stop Papers from Reviewing Themselves”, 2009 30th IEEE Symposium on Security and Privacy.

[15] Peiqing Zhang, Bjarne E. Helvik,” Modeling and Analysis of P2P Content Distribution under Coordinated Attack Strategies”, 7th IEEE International Workshop on Digital Rights Management Impact on Consumer Communications (DRM 2011).

[16] Suhas Mathur and Wade Trappe,” BIT-TRAPS: Building Information-Theoretic Traffic Privacy into Packet Streams”, IEEE Transactions on Information Forensics and Security, VOL. 6, NO. 3, September 2011.

[17] Brad Wardman, Tommy Stallings, Gary Warner, Anthony Skjellum,” High-Performance Content-Based Phishing Attack Detection”, " eCrime Researchers Summit (eCrime), 2011 , vol., no., pp.1,9, 7-9 Nov. 2011.

[18] Usman Shaukat Qurashi , Zahid Anwar,” AJAX Based Attacks: Exploiting Web 2.0”, 2012 IEEE.

[19] Fokko Beekhof, Sviatoslav Voloshynovskiy , Farzad Farhadzadeh,” Content Authentication and Identification under Informed Attacks”, IEEE 2012.

[20] Seungoh Choi, Kwangsoo Kim, Seongmin Kim, and Byeong-hee Roh,” Threat of DoS by Interest Flooding Attack in Content-Centric Networking” IEEE 2013.

[21] Ms.Namrata Shukla, Ms. Shweta Pandey,” Document Fraud Detection with the help of Data Mining and Secure Substitution Method with Frequency Analysis”, International Journal of Advanced Computer Research (IJACR),Volume 2 Number 2 June 2012.

[22] Gebre, M.T.; Kyung-Suk Lhee; Manpyo Hong, "A robust defense against Content-Sniffing XSS attacks," Digital Content, Multimedia Technology and its Applications (IDC), 2010 6th International Conference on , vol., no., pp.315,320, 16-18 Aug. 2010.

[23] Qurashi, U.S.; Anwar, Z., "AJAX based attacks: Exploiting Web 2.0," Emerging Technologies (ICET), International Conference on , vol., no., pp.1,6, 8-9 Oct. 2012.

[24] Beekhof, F.Voloshynovskiy, S. Farhadzadeh, "Content authentication and identification under informed attacks," Information Forensics and Security (WIFS), IEEE International Workshop on , vol., no., pp.133,138, 2-5 Dec. 2012.

[25] Spear-Phishing Email: Most Favored APT Attack, Trend Micro Incorporated Research Paper 2012.

[26] A Websense® White Paper,” Advanced Persistent Threats and Other Advanced Attacks: Threat Analysis and Defense Strategies for Smb, Mid-Size, and Enterprise Organizations”,2012.

[27] Van Lam Le, Ian Welch, Xiaoying Gao,Peter Komisarczuk,” Anatomy of Drive-by Download Attack”, Proceedings of the Eleventh Australasian Information Security Conference (AISC 2013), Adelaide, Australia.

[28] Ranbir Sinha, Nishant Behar, Devendra Singh, “Secure Handshake in Wi-Fi Connection (A Secure and Enhanced Communication Protocol)”, International Journal of Advanced Computer Research (ISSN (IJACR) ,Volume 2, Number 1,March 2012.

BIOGRAPHIES

Shweta Pandey received the Bachelor of Engineering Degree in Computer Science and Engineering from RKDF Institute of Science & Technology, Bhopal India and Persuing M.Tech degree in Computer Science and Engineering from NRI Institute of Information Science & Technology, Bhopal, India. Her research interests are centered around web application security .

Abhishek Singh Chauhan is an Assistant Professor in Computer Science and Engineering Department at NRI Institute of Information Science & Technology, Bhopal, INDIA. He has completed M.Tech (C.S.E) from Samrat Ashok Technological Institutute. and Persuing PhD. from Bhagwant University, Ajmer, India . His main research interests includes Web Application Security & Network Security.