A Study on XSS Attacks: Intelligent Detection Methods

V S Stency1 and N Mohanasundaram 2

1Research Scholar, Department of CSE, Karpagam Academy of Higher Education, Coimbatore - 641021, Tamil Nadu, India.
2Professor, Department of CSE, Faculty of Engineering, Karpagam Academy of Higher Education, Coimbatore - 641021, Tamil Nadu, India.

E-mail: stenz.denz@gmail.com

Abstract: Cross-site scripting is one of the standard web application attacks vulnerable to the application layer. The attacker handles malicious scripting for trusted websites and inject the script. There are numerous types of XSS scripting vulnerable to attack websites incredibly open web applications. The attacker can load or redirect to the malicious webpage. The XSS is susceptible to attack significant websites like medical, e-commerce, banking, etc. The detection and prevention of XSS attacks are still complicated. Plenty of research has been carried out to control the XSS based attack. This paper analyses the XSS attack detection methods by various performance metrics. Numerous works issued in the widespread journals between 2019 and 2020 are reviewed in this paper to accomplish these requirements. The reviewed articles are compared concerning algorithms’ simplicity, the type they belong, and the performance metrics. The work assumed that the movement in the application of elementary methods to detect XSS attacks is better than the recommendations that custom some artificial-intelligence techniques.

1. Introduction

At present website makes the meaningful environment in all field, there has been plenty of websites developed last few decades. The web-based application does business quickly and smartly [1]. The web-based application is the part of intelligent technological development in specific. The web delivers a technique for vendors to know the individuals visiting the sites and communicate directly based on business communication [2, 3]. The Web application makes good relations with customer-to-customer and customer-to-the supplier. The enhancement has been made with subscription, order, likes, rating about the product, etc. [4, 5].

There has been plenty of advancement applied to a web-based application. The data mining concept has been applied to mine the particular web rating, pattern-based sales, customer review analysis [6, 7, 8, 9].

The considerable development involved with some threats, one of the most developed threats is web-based attacks [10, 11]. There has been plenty of web-based attacks been able to threaten the activity of web applications (hijacking, flooding, slowloris, XSS., etc.))[12]. The attackers are a
vulnerable and easy way to attack the web without any specified interest. The open web applications are more comfortable to attack [13].

If the attacker has little knowledge about web development and scripting enables attackers to inject client-side scripts into web pages viewed by other users on the website is called a Cross-site Scripting Attack. XSS attacks have been a kind of injection; the malicious scripting has been injected to the website mislead to page or redirect to the attacker site [14]. This can happen through cookies, session token, etc. The HTML page content can also be rewritten with an XSS-based attack [15].

In this framework, the work's primary goal is to detect an efficient tool and detection method for XSS in various web applications. We have searched various research articles from different journals. The search is based on XSS detection and prevention mechanism, especially based on deep learning and artificial intelligence technique based on XSS detection and classification. The various methods have been compared with different performance metrics. The comparison also conveys the limitations of the work. There has been plenty of surveys that have been carried out for XSS based attack detection methods [16,17,18,19,20].

The paper structured as follows, section 1 convey the introduction of the work with study methodology, section 2 describes the basic about the XSS and vulnerability, section 3 gives the review of intelligent XSS detection methods. Section 4 describes the other detection tools and techniques, section 5 discusses the review process, and section 6 concludes the work process.

2. XSS Attack Basics

XSS's vulnerability has been specified by ref [1,2], persistent, non-persistent, and DOM-based. The classification is also defined in the server-side and client-side. Both persistent and non-persistent XSS has a server-side XSS, and DOM is the subset of client XSS. There has been some defense against the XSS, the Context-sensitive server-side output encoding, and safe JavaScript APIs for client XSS. Figure 1 represents the process of XSS attack. The attacker sends the script injected link to the web user. The user clicks the link for a legitimate website request. The request is then sent to the web browser and responds with legitimate websites' loading and executed malicious code. Finally, the attacker gets the private user data.

![Figure 1. Process of XSS attack](image)

2.1 Performance metrics

The precision, recall, and Accuracy has been the most evaluated performance metrics in the attack detection method.
Precision = \frac{T_p}{T_p + F_p} \tag{1}

Recall = \frac{T_p}{T_p + F_n} \tag{2}

Accuracy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \tag{3}

3. XSS- SMART WAY OF DETECTION

The smart way of XSS based attack has been detected and prevent using deep learning techniques called DeepXSS. In this method, the XSS has been tested and trained with some features. There has been plenty of research carried out. The following table 1 representsts the comparison of different intelligent algorithms for XSS detection.

| Author                        | year | Methods                        | Performance                          | Result                                      |
|-------------------------------|------|--------------------------------|--------------------------------------|---------------------------------------------|
| Xueqin Zhang                  | 2020 | MCTS, GAN, Deep learning       | Recall rate (TPR) Precision          | Suitable only for adversarial examples      |
| Yue Zhou                      |      | CICIDS2017 dataset              |                                      |                                             |
| Songwen Pei                   |      |                                |                                      |                                             |
| Jingjing Zhuge                |      |                                |                                      |                                             |
| Jiahaosh Chen[30]             |      |                                |                                      |                                             |
| Stanislav Abaimov             | 2019 | CNN, With Deep Learning         | Accuracy Recall Precision            | 94% accuracy, 99% precision, and a 93% recall value. |
| Giuseppe Bianchi[29]          |      | Real-time data set SQL and XSS  |                                      |                                             |
|                               |      | Pre-pressing: filter noise      |                                      |                                             |
|                               |      | GitHub                         |                                      |                                             |
| Fawaz Mahiuob, Mohammed       | 2019 | ANN, MLP, DFE dynamic-features extraction technique | Accuracy overall Misclassification Rate, precision Detection Rate (DR) or Recall | high precision and low complexity |
| Mokbal, Wang Dan, Azhar Imran, Lin Jiuchuan, Faheem Akhtar, And Wang Xiaoxi[28] | | | | |
| Wenchuan                       | 2019 | GRU, malicious URL detection, network attack, character-level embedding, CNN FCNN | Precision Recall Accuracy F-measure | Efficient method |
| Yang, Wen Zuo, And Baojiang Cui[27] | | | | |
| Yun Zhou *, Peichao           | 2019 | Ensemble learning method Bayesian network | Average classification | Not suitable for outside attack |
Wang[35]  Binary classification accuracy
    problem
Cyber thread intelligent
Bagging and majority voting
GitHub

Yong Fang
Yang Li
Liang Liu
Cheng Huang[31]

2018  word2vec Precision  DL reaches a precision rate
DeepXSS Recall  99.5% and a recall rate 97.9%
Long Short-Term Memory (LSTM) RNN

In 2020 the ref [30] experimented deep learning-based XSS detection with the CICIDS2017 dataset. The work modelled the Generative Adversarial Network (GAN) for prone to misjudging adversarial. The Monte Carlo tree search (MCTS) algorithm generates a model for adversarial for train and testing. The performance has been carried out with the performance metrics of recall rate and precision. The result shows that Adversarial detection has been improved with the MCTS algorithm, and the detection time has been optimized with GAN. The limitations of this model have been designed only for detecting adversaries.

In 2019 the ref [29] experimented XSS attack with SQL injection code detection in web application. The CDNN convolutional deep neural network has been used for the pre-processing stage. Noise filtered by encoding and training CDNN to remove SQL and XSS special symbols. GitHit dataset has been used for experimentation. The performance has been evaluated by the metrics of accuracy, recall, and precision. The result shows that the processing time has been reduced with the encoding technique and enhanced the effect of DCNN.

In 2019 the ref [28] deals with the XSS detection by integrating Multilayer perceptron with ANN. In this work, the dynamic-feature extraction technique has been applied for extracting the XSS attack feature. The dynamic behavior has been trained and tested with the dataset feen into the artificial neural network. The performance has been calculated with Accuracy's metrics overall, Misclassification Rate, precision Detection Rate (DR), or Recall. The result shows that high precision and low complexity are noted in the proposed model.

In 2019 the ref [27] experimented Convolution neural network-based XSS detection with URL. The gate-recurrent unit has been used to collect time dimensional malicious keyword in URL in the original pooling layer with feature acquisitions. The feature extraction module has extracted the collected keywords. Finally, the fully connected neural network has been classifying and detect features. The performance has been calculated with the metrics of Accuracy, Recall, precision, and f-measure. The result shows that the model is efficient with FCNN.

Based on the above review, the deep learning-based XSS detection has been started from 2018-2020. Various methods and algorithms are integrated to produce efficient detection mechanisms, RNN, CNN, CDNN, and RNN used for smart detection of an XSS attack.

4. Other XSS detection methods

There have been other methods also able to detect XSS attacks. In 2020 the ref [26] experimented Gray box mechanism instead of a black box. This method used HTML output with context-sensitive XSS flaws based on the HTTP request. OWASP Zed Attack Proxy is used for experimentation. Severity Improper Sanitizations Performance Statistics evaluated the performance on Response Injection Granularity. The result shows that intercept traffic to non-opensource databases.
### Table 2. Other XSS Detection methods

| Author                        | Year | Techniques used                                                                 | Performance metrics                  | Result and analysis                                      |
|-------------------------------|------|---------------------------------------------------------------------------------|--------------------------------------|----------------------------------------------------------|
| A. Steinhauser and P. Tůma[26]| 2020 | gray box mechanism HTML output cooperates with the database context-sensitive XSS flaws HTTP e OWASP Zed Attack Proxy | Severity Incorrect Sanitizations Performance Statistics on Response Injection Granularity | intercept traffic to non-opensource databases             |
| Shashank Gupta, B.B. Gupta, Pooja Chaudhary[32] | 2017 | DOM-based XSS attack Mobile cloud computing context-sensitive sanitization HTTP request Js files | Precision Recal F-measure Response time | very low and acceptable false-positive and false -negatives |
| Ran Wang, Guangquan Xu*, Xianjiao Zeng, Xiaohong Li, Zhiyong Feng[34] | 2017 | DOM-based XSS attack Taint tracking analysis | dynamic analysis method | lower rate of false-positive and false-negative |
| Shashank Gupta, B.B. Gupta[22] | 2016 | Java Script injection attacks PHP web application BlogIT client-side JavaScript library | analysis method | acceptable runtime overhead |
| Shashank Gupta, B.B.Gupta[25] | 2016 | reflected XSS-attack HTTP request JS files Virtual cloud server | Precision Efficiency Transparency Dynamic Reaction deployability | detects the XSS-attack by low false-positives and false -negatives |
| Shashank Gupta, B. B. Gupta[23] | 2016 | HTML5-CSS hidden vulnerable injection points i | Precision Recal F-measure | a high rate of true-positives and low rate of false-negatives and false-positives. |
| M.I.P. SalasE. Martins[21]    | 2014 | Penetration Testing and FaultInjection SoapUI SecurITree HTTP | Vulnerability analysis | limited in the tools commonly used for security testing |
| L.K. Shar H.B.K. Tan[33]      | 2012 | tainted information flow graph, seven Java-based web applications XSS defence feature HTML output | Vulnerable statements reported, #False positives | does not address DOM-based XSS as it would require the analysis of client-side scripts |

In the ref[32,22,25,23] experimented by the same author, mobile cloud computing, PHP web application, virtual cloud server, and HTML scripting. The method [32] DOM-based attack in mobile script injection attacks. The ref[22] experimented by the same author, mobile cloud computing, PHP web application, virtual cloud server, and HTML scripting. The method [22] DOM-based attack in mobile script injection attacks.
cloud computing. The work used context-sensitive sanitization with HTTP requests. Precision Recal F-measure and Response time have evaluated the performance. The result shows that very low acceptable false positive and false negatives have been noted. The method [22] has been used for PHP web applications with BlogIT to detect JavaScript injection attacks. The experiment has been carried out with the client-side JavaScript library. The performance shows that this method worked with acceptable runtime overhead. The process [25] has been used to detect the reflected XSS attack in a virtual cloud server based on HTTP requests with experimented using java scripting files. The performance has been evaluated with Precision Efficiency Transparency; Dynamic Reaction deployability. The result shows that this method detects XSS-attack by a low false-positive rate and false- negatives.

The other detection-based method can employ only a particular task, mostly for HTTP requests from the above review. The javascript has been vulnerable to attack by XSS. The cloud environment has also experimented with an XSS attack in the study. The standard method is not suitable for intelligent detection methods for future smart web development.

5. Discussion
Deep learning-based XSS detection plays a significant role in developing efficient intrusion detection against XSS. The XSS features have been extracted and classified depends on the training and testing purpose. DeepXSS detects the attack more efficiently with other integrated mechanisms like ANN, MCTS, GAN, CID, DFE, GRU, etc. The other tool called CNN, CDN, RNN, and FCNN mechanism improved the precision-recall and Accuracy detection ratio. The dataset has been more critical in the training process, includes GitHut, CICIDS2017, word2vec etc.

The work's academic consequences presented the various detection and prevention methods for XSS and the accompaniment of the work illustrated different Deep learning techniques with artificial intelligence techniques. This work will guide researchers in this field and experts who are enthusiastic about investigating XSS attacks. The practical suggestions of this work allow new integrated algorithm development with ANN with XSS attack detection. The work also guides other resent technology development like cloud, IoT, and smart web applications.

Deep learning utilizes fell layers in an order structure to perform information preparing, which brings about huge outcomes in spaces of solo element learning and example acknowledgment. Propelled by execution of deep learning techniques, we accept deep learning is significant for field of organization security, in order to survey the current deep learning strategies for XSS-attack discovery. We break down recent strategies, characterize them as indicated by various deep learning procedures, and pack the exhibition of the most delegate techniques.

6. Conclusion
Various cross-site scripting detection methods have been analyzed in the direction to find out the tendency of exploitation tools or approaches to detect XSS attacks. Over the variety of performance metrics, each method has been evaluated. The results of this research show Deep learning-based XSS detection plays a significant role in developing efficient intrusion detection against XSS. The XSS features have been extracted and classified depends on the training and testing purpose. The result of the study guides many researchers to present the various XSS based proposal. As per the above examination, we hold a conviction that this outline is an advantage for the individuals who have thoughts to improve the exhibition of attack recognition as far as exactness; our survey will give direction and word references to additional exploration in this field of deep learning based XSS-attack detection. Also, the future direction of the work leads to implement strong artificial intelligence-based tools and techniques against the XSS attack.
References
[1] Saborido, Rubén, and Enrique Alba 2020 Software systems from smart city vendors. Cities 101: 102690.
[2] Kahle, Júlia Hofmeister, Érico Marcon, Antonio Ghezzi, and Alejandro G. Frank 2020 Smart Products value creation in SMEs innovation ecosystems. Technological Forecasting and Social Change 156: 120024.
[3] Chen, Renee Rui, Robert M. Davison, and Carol Xiaojuan Ou 2020 A symbolic interactionism perspective of using social media for personal and business communication. International Journal of Information Management 51: 102022.
[4] Molinillo, Sebastian, Rafael Anaya-Sánchez, and Francisco Liebana-Cabanillas 2020 Analyzing the effect of social support and community factors on customer engagement and its impact on loyalty behaviors toward social commerce websites. Computers in Human Behavior 108: 105980.
[5] Rese, Alexandra, Lena Ganster, and Daniel Baier 2020 Chatbots in retailers’ customer communication: How to measure their acceptance? Journal of Retailing and Consumer Services 56: 102176.
[6] Barenji, Ali Vatankhah, Hanyang Guo, Yitong Wang, Zhi Li, and Yiming Rong 2020 Toward blockchain and fog computing collaborative design and manufacturing platform: Support customer view. Robotics and Computer-Integrated Manufacturing 67: 102043.
[7] Shokouhyar, Sajjad, and Sanaz Amirmokhtar Radi 2020 Toward Consumer perception of cellphones sustainability: A Social media analytics. Sustainable Production and Consumption.
[8] Wei, Jie, Yue Wang, and Jinghui Lu. 2020 Information sharing and sales patterns choice in a supply chain with product's greening improvement. Journal of Cleaner Production: 123704.
[9] Paschen, Jeannette, Matthew Wilson, and João J. Ferreira. 2020 Collaborative intelligence: How human and artificial intelligence create value along the B2B sales funnel. Business Horizons.
[10] Nguyen, Van-Linh, Po-Ching Lin, and Ren-Hung Hwang 2019 Web attacks: defeating monetisation attempts. Network Security 2019, no. 5: 11-19.
[11] Mansoori, Masood, and Ian Welch 2020 How Do They Find Us? A Study of Geolocation Tracking Techniques of Malicious Websites. Computers & Security: 101948.
[12] Liang, Jingxi, Wen Zhao, and Wei Ye 2017 Anomaly-based web attack detection: a deep learning approach. Proceedings of the 2017 VI International Conference on Network, Communication and Computing, pp. 80-85. 2017.
[13] Chiew, Kang Leng, Kelvin Sheng Chek Yong, and Choon Lin Tan 2018 A survey of phishing attacks: Their types, vectors and technical approaches. Expert Systems with Applications 106: 1-20.
[14] Marashdih, Abdalla Wasef, and Zarul Fitri Zaaba 2017 Cross site scripting: removing approaches in web application. Procedia Computer Science 124: 647-655.
[15] Rodríguez, Germán E., Jenny G. Torres, Pamela Flores, and Diego E. Benavides 2020 Cross-site scripting (XSS) attacks and mitigation: A survey. Computer Networks 166: 106960.
[16] Sarmah, Upasana, D. K. Bhattacharyya, and Jugal K. Kalita 2018 A survey of detection methods for XSS attacks Journal of Network and Computer Applications 118 : 113-143.

[17] Hydara, Isatou, Abu Bakar Md Sultan, Hazura Zulzalil, and Novia Admodiastro 2015 Current state of research on cross-site scripting (XSS)–A systematic literature review Information and Software Technology 58 : 170-186

[18] Cui, Yanpeng, Junjie Cui, and Jianwei Hu 2020 A Survey on XSS Attack Detection and Prevention in Web Applications Proceedings of the 2020 12th International Conference on Machine Learning and Computing, pp. 443-449.

[19] Liu, Miao, Boyu Zhang, Wenbin Chen, and Xunlai Zhang 2019 A Survey of Exploitation and Detection Methods of XSS Vulnerabilities IEEE Access 7 : 182004-182016.

[20] Faghani, Mohammad Reza, and Uyen Trang Nguyen 2013 A study of XSS worm propagation and detection mechanisms in online social networks IEEE transactions on information forensics and security 8, no. 11 : 1815-1826.

[21] Salas, Marcelo Invert Palma, and Eliane Martins 2014 Security testing methodology for vulnerabilities detection of xss in web services and ws-security Electronic Notes in Theoretical Computer Science 302 : 133-154.

[22] Gupta, Shashank, and B. B. Gupta 2016 Automated discovery of JavaScript code injection attacks in PHP web applications Procedia Computer Science 78 : 82-87.

[23] Gupta, Shashank, and B. B. Gupta 2016 CSSXC: Context-sensitive sanitization framework for Web applications against XSS vulnerabilities in cloud environments Procedia Computer Science 85 : 198-205.

[24] Kaur, Gurpreet, Bhavika Pande, Aayushi Bhardwaj, Gargi Bhagat, and Shashank Gupta 2018 Efficient yet robust elimination of XSS attack vectors from HTML5 web applications hosted on OSN-based cloud platforms Procedia Computer Science 125 : 669-675.

[25] Gupta, Shashank, and B. B. Gupta 2016 Enhanced XSS defensive framework for web applications deployed in the virtual machines of cloud computing environment Procedia Technology 24 : 1595-1602.

[26] Steinhauser, Antonin, and Petr Tůma 2020 Database Traffic Interception for Graybox Detection of Stored and Context-Sensitive XSS arXiv preprint arXiv:2005.03322.

[27] Yang, Wenchuan, Wen Zuo, and Baojiang Cui 2019 Detecting malicious urls via a keyword-based convolutional gated-recurrent-unit neural network IEEE Access 7 : 29891-29900.

[28] Mokbal, Fawaz Mahiuob Mohammed, Wang Dan, Azhar Imran, Lin Jiuchuan, Faheem Akhtar, and Wang Xiaoxi 2019 MLPXSS: An Integrated XSS-Based Attack Detection Scheme in Web Applications Using Multilayer Perceptron Technique IEEE Access 7 : 100567-100580.

[29] Abaimov, Stanislav, and Giuseppe Bianchi 2019 CODDL: Code-injection detection with deep learning IEEE Access 7 : 128617-128627.

[30] Zhang, Xueqin, Yue Zhou, Songwen Pei, Jingjing Zhuge, and Jia Hao Chen 2020 Adversarial Examples Detection for XSS Attacks Based on Generative Adversarial Networks IEEE Access 8 : 10989-10996.

[31] Fang, Yong, Yang Li, Liang Liu, and Cheng Huang 2018 DeepXSS: Cross site scripting detection based on deep learning Proceedings of the 2018 International Conference on Computing and Artificial Intelligence, pp. 47-51.
[32] Gupta, Shashank, Brij Bhooshan Gupta, and Pooja Chaudhary 2018 Hunting for DOM-Based XSS vulnerabilities in mobile cloud-based online social network *Future Generation Computer Systems* 79: 319-336.

[33] Shar, Lwin Khin, and Hee Beng Kuan Tan 2012 Auditing the XSS defence features implemented in web application programs *IET software* 6, no. 4 : 377-390.

[34] Wang, Ran, Guangquan Xu, Xianjiao Zeng, Xiaohong Li, and Zhiyong Feng 2018 TT-XSS: A novel taint tracking based dynamic detection framework for DOM Cross-Site Scripting *Journal of Parallel and Distributed Computing* 118 : 100-106.

[35] Zhou, Yun, and Peichao Wang 2019 An ensemble learning approach for XSS attack detection with domain knowledge and threat intelligence *Computers & Security* 82 : 261-269.