ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 4, Issue 12, December 2016

A Survey on Automated Prediction of Cross Site Scripting Vulnerabilities in Web Applications

Bhavana D. Dhobale[1], Prof. Priya Pise[2]

M. E Student, Dept. of Computer, Indira College of Engineering, Pune, Maharashtra, India[1]

Professor, Dept. of Computer, Indira College of Engineering, Pune, Maharashtra, India[2]

ABSTRACT: Recently, machine-learning based mostly vulnerability prediction models are gaining quality in internet security house, as these models offer an easy and efficient thanks to handle internet application security problems. Existing state-of-art Cross-Site Scripting (XSS) vulnerability prediction approaches don't contemplate the context of the user-input in output-statement, that is incredibly vital to spot context-sensitive security vulnerabilities. During this paper, we have a tendency to propose a completely unique feature extraction algorithmic rule to extract basic and context options from the ASCII text file of internet applications. Our approach uses these options to create varied machine-learning models for predicting context-sensitive Cross- web site Scripting (XSS) security vulnerabilities. Experimental results show that the projected options based mostly prediction models will discriminate vulnerable code from non-vulnerable code at a really low false rate. In proposed system it will predict cross-site scripting as well as SQL injection attacks using reverse proxy server.

KEYWORDS: web application security, cross-site scripting vulnerability, machine learning, context-sensitive, input validation

I. INTRODUCTION

A large variety of individuals area unit betting on internet applications that get used for social communications, health services, monetary transactions and type of alternative functions. These internet Applications handle great deal of user's personal knowledge. The importance of this data like personal data interests the assaulter in internet application. However, the presence of security vulnerabilities limits the utilization of those applications as malicious user will steal sensitive data, send felonious protocol requests, direct benign user to malicious websites, install malware, and perform numerous alternative malicious operations. the most reason of XSS vulnerabilities is weakness within the ASCII text file which allows the utilization of user-input in internet server output statement with none validation.

It's seen that if computer code metrics obtained from ASCII text file and development history area unit prognostic of vulnerable code locations then security consultants will use computer code metrics for prediction purpose. During this computer code metrics and static code attributes area unit necessary options for building of machine learning model for predicting security vulnerabilities [1]. To predict vulnerable computer code the approach that is employed supported text mining the ASCII text file of element. During this options get extracted to ascertain whether or not the parts contain vulnerability that is incredibly necessary for prediction of XSS vulnerabilities [3].

It's seen that a unique approach to extract basic and context options from ASCII text file that build machine learning primarily based vulnerability prediction model. To the simplest of our information, this can be the primary approach to use context data for predicting XSS vulnerabilities. The projected approach has enforced in an exceedingly paradigm tool for automatic extraction of those options from PHP ASCII text file.

Internet Application:
Three tier internet applications comprises presentation logic, business logic and knowledge logic. Presentation logic is wherever computer programme (UI) is developed victimisation that users initiate internet requests. Business logic is

DOI: 10.15680/IJIRCCE.2016. 0412085

wherever the validations and internet service functionalities area unit written. Knowledge logic is said with all the info queries generated as a results of internet requests.

Cross website Scripting Attacks (XSS):
In XSS attacks, assaulter tries to inject a shopper aspect scripting to the remote server. XSS usually attacks the hypertext markup language of the net page being loaded. ActiveX, JavaScript is littered with this attack. XSS will reveal cookie data. The scripts is hosted somewhere by the assaulter. Assaulter provides a link to the users that appears real however has malicious script code. Once user reaches the link script run on client’s machine permitting assaulter to realize very important data.

a. Non-persistent XSS attack: During this form of attack the attack script is unbroken in other places instead of the info server of internet application. Such malicious scripts area unit created accessible to users in such some way that to user it looks legit. Once this script is touched hacker will gain access to user data.

b. Persistent XSS attack: Because the name counsel, persistent XSS attack stores the ambiguous script on secondary devices of internet application like info of three tier internet application. This sort of attack is feasible wherever internet application permits user to store data like comments, diary sections etc. That is additionally visible to alternative users. Session hijacking is done victimisation persistent XSS.

II. LITERATURE SURVAY

1. Evaluating complexity, code churn, and developer activity metrics as indicators of software vulnerabilities.
Authors:J. Anvik, L. Hiew, and G. C. Murphy
Security scrutiny associate degreed testing need consultants in security WHO suppose like an offender. Security consultants ought to apprehend code locations on that to focus their testing and scrutiny efforts. Since vulnerabilities ar rare occurrences, locating vulnerable code locations is a difficult task. we have a tendency to investigated whether or not package metrics obtained from ASCII text file and development history ar discriminative and prophetical of vulnerable code locations. If so, security consultants will use this prediction to prioritise security scrutiny and testing efforts. The metrics we have a tendency to investigated comprise 3 categories: quality, code churn, and developer activity metrics. we have a tendency to performed 2 empirical case studies on giant, wide used ASCII text file projects: the Mozilla Firefox applications programme and therefore the Red Hat Enterprise UNIX operating system kernel. The results indicate that twenty four of the twenty eight metrics collected ar discriminative of vulnerabilities for each comes. The models victimization all 3 forms of metrics along foretold over eighty % of the best-known vulnerable files with but twenty five % false positives for each comes. Compared to a random choice of files for scrutiny and testing, these models would have reduced variety|theamount|the quantity} of files and therefore the number of lines of code to examine or take a look at by over seventy one and twenty eight %, severally, for each comes.

2.Using complexity, coupling, and cohesion metrics as early indicators of vulnerabilities
Author:H. P. Luhn
Software security failures square measure common and also the drawback is growing. A vulnerability could be a weakness within the code that, once exploited, causes a security failure. It's tough to notice vulnerabilities till they manifest themselves as security failures within the operational stage of code, as a result of security issues square measure typically not addressed or famed sufficiently early throughout the code development life cycle. Varied studies have shown that quality, coupling, and cohesion (CCC) connected structural metrics square measure vital indicators of the standard of code design, and code design is one in every of the foremost vital and early style choices that influences the ultimate quality of the package. Though these metrics are with success used to point code faults normally, there aren't any systematic pointers on the way to use these metrics to predict vulnerabilities in code. If cardinal metrics may be wont to indicate vulnerabilities, these metrics might aid within the conception of additional secured design, resulting in additional secured style and code and eventually higher code. During this paper, we have a tendency to gift a framework to mechanically predict vulnerabilities supported cardinal metrics. To through empirical observation validate the framework and prediction accuracy, we have a tendency to conduct an oversized empirical study on 52 releases of Mozilla Firefox developed over a amount of 4 years.

3. Predicting vulnerable software components via text mining
Authors:J. Anvik and G. C. Murphy
This paper presents associate approach supported machine learning to predict that elements of a code application contain security vulnerabilities. The approach is predicated on text mining the ASCII text file of the elements. Namely, every part is characterised as a series of terms contained in its ASCII text file, with the associated frequencies. These options square measure accustomed forecast whether or not every part is probably going to contain vulnerabilities. In associate exploratory validation with twenty mechanical man applications, we tend to discovered that a dependable prediction model will be designed. Such model may be helpful to rank the validation activities, e.g., to spot the elements needing special scrutiny.

4. Software vulnerability prediction using text analysis techniques. Proceedings of the 4th International Workshop on Security Measurements and Metrics
Author:C. C. Aggarwal and P. Zhao,
It is with nice pleasure that we tend to welcome you to the eighth International Workshop on Security Measurements and Metrics (MetriSec 2012). This year's MetriSec are going to be to a small degree totally different from previous years, and that we ar terribly excited to ascertain however it'll end up. Whereas we tend to after all have a paper-presenting session and a oratory, we tend to even have for the primary time a give-and-take. Our mission of providing a venue wherever we tend to share new concepts on a way to get, use, deploy, and appraise security metrics has not modified, and that we needed to undertake a additional interactive workshop now. Betting on the results, we'd continue this vogue next year, therefore take care to inform United States what you're thinking that.
The call for papers attracted nine submissions from Asia, Europe, and also the u. s.. The program committee accepted four papers that cowl a spread of topics, as well as vulnerability prediction, security image, and attack surfaces.
In addition, the programme has two different highlights. First, there'll be a oratory by Peter Gutmann, entitled From Revenue Assurance to Assurance: The Importance of measuring in laptop Security. Peter has been performing on each software package security and security software package, and his book cryptological Security design, style and Verification (Springer Verlag 2003) ought to get on each developer's and each security researcher's shelf.

5.Automatic detection and correction of web application vulnerabilities using data mining to predict false positives.
Author:K. Balog, L. Azzopardi, and M. de Rijke
Web application security is a vital drawback in today's net. a serious explanation for this standing is that a lot of programmers don't have adequate data concerning secure writing, so that they leave applications with vulnerabilities. associate approach to unravel this drawback is to use ASCII text file static analysis to search out these bugs, however these tools ar famed to report several false positives that build exhausting the task of correcting the appliance. This paper explores the employment of a hybrid of ways to sight vulnerabilities with less false positives. when associate initial step that uses taint analysis to flag candidate vulnerabilities, our approach uses data processing to predict the existence of false positives. This approach reaches a trade-off between 2 apparently opposite approaches: humans writing the data concerning vulnerabilities (for taint analysis) versus mechanically getting that data (with machine learning, for information mining). Given this additional precise variety of detection, we have a tendency to do automatic code correction by inserting fixes within the ASCII text file. The approach was enforced within the WAP tool associated an experimental analysis was performed with an outsized set of open supply PHP applications.

III.PROPOSED SYSTEM

It is seen that in this approach. First, it sends request to server. Then, request redirect to proxy server and proxy server validate request. Then, valid request sent to database through web application and database sent data in the form of response through web application.
SQL injection vulnerability is one of the most serious problems to the database applications. It may allow an attacker to gain complete access to their databases. A SQL Injection identifies weaknesses in the applications.
Reverse Proxy is a technique which is used to sanitize the users’ inputs that may transform into a database attack. In this technique a filter program redirects the user’s input to the proxy server before it is sent to the application server. At the proxy server, data cleaning algorithm is triggered using a sanitizing application. Also it will detect Cross-Site

DOI: 10.15680/IJIRCCE.2016. 0412085

Scripting Vulnerability which is one of the most common attacks. In this malicious script get generated to access personal information.

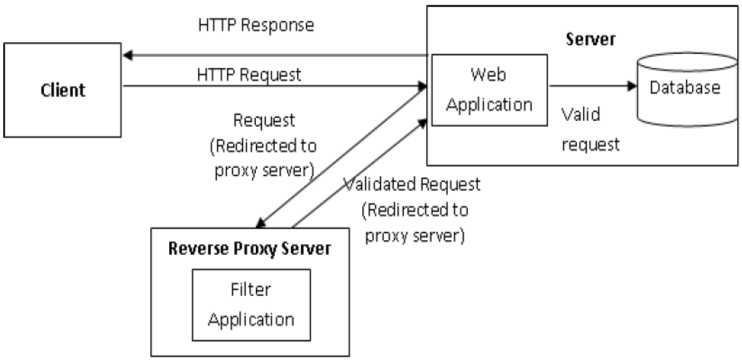

Fig1. Architecture Diagram

OBJECTIVE:

Cross-Site Scripting vulnerabilities are represented mutually of the foremost serious threats to the net applications. Net applications that square measure liable to XSS could enable associate degree offender to realize complete access to their personal data.
It's seen that to extract basic and context options from the ASCII text file of net applications a unique feature extraction rule get used. To create numerous machine-learning models for predicting context-sensitive Cross- website Scripting (XSS) security vulnerabilities associate degree approach uses these options.Also to detect SQL injection attacks using reverse proxy.
The project aims to work out a good approach to predict and curb the XSS attacks in net applications victimization feature extraction rule.

IV.CONCLUSION

It is seen that vulnerability prediction is a crucial task in securing the net applications before their unleash. Insecure net applications could reason behind stealing personal and crucial user data. It's seen that a completely unique approach to extract relevant options that classify vulnerable ASCII text file from benign one. An approach uses these options to create numerous machine-learning models for predicting context-sensitive Cross- website Scripting (XSS) security vulnerabilities. It is also seen that proposed system protects the application from SQL injection attack using reverse proxy.

ACKNOWLEDGMENT

We might want to thank the analysts and also distributers for making their assets accessible. We additionally appreciative to commentator for their significant recommendations furthermore thank the school powers for giving the obliged base and backing.

REFRENCES

[1] WhiteHatSecurity. Web statistics report. https://whitehatsec.com/categories/statistics-report, 2013. Accessed: 2013-06-26.
[2] IsatouHydara, Abu Bakar Md. Sultan, HazuraZulzalil, and NoviaAdmodisastro. Current state of research on cross-site scripting a systematic literature review. Information and Software Technology, 58(0):170 – 186, 2015.
[3] Yonghee Shin, A. Meneely, L. Williams, and J.A. Osborne. Evaluating complexity, code churn, and developer activity metrics as indicators of software vulnerabilities. IEEE Transactions on Software Engineering, 37(6):772–787, Nov 2011.
[4] Istehad Chowdhury and Mohammad Zulkernine. Using complexity, coupling, and cohesion metrics as early indicators of vulnerabilities. Journal of Systems Architecture, 57(3):294 – 313, 2011. Special Issue on Security and Dependability Assurance of Software Architectures.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 4, Issue 12, December 2016

[5] J. Walden, J. Stuckman, and R. Scandariato. Predicting vulnerable components: Software metrics vs text mining. IEEE 25th International Symposium on Software Reliability Engineering (ISSRE), pages 23–33, Nov 2014.
[6] LwinKhinShar and HeeBengKuan Tan. Predicting sql injection and cross site scripting vulnerabilities through mining input sanitization patterns. Information and Software Technology, 55(10):1767 – 1780, 2013.
[7] R. Scandariato, J. Walden, A. Hovsepyan, and W. Joosen. Predicting vulnerable software components via text mining. IEEE Transactions on Software Engineering, 40(10):993–1006, Oct 2014.
[8] LwinKhinShar and HeeBengKuan Tan. Automated removal of cross site scripting vulnerabilities in web applications. Information and Software Technology, 54:467–478, 2012.
[9] PrateekSaxena, David Molnar, and Benjamin Livshits. Scriptgard: Automatic context-sensitive sanitization for large-scale legacy web applications. Proceedings of the 18th ACM Conference on Computer and Communications Security, pages 601–614, 2011.
[10] LwinKhinShar, HeeBengKuan Tan, and Lionel C. Briand. Mining sql injection and cross site scripting vulnerabilities using hybrid program analysis. Proceedings of the 2013 International Conference on Software Engineering, pages 642–651, 2013.
[11] Aram Hovsepyan, Riccardo Scandariato, WouterJoosen, and James Walden. Software vulnerability prediction using text analysis tech- niques. Proceedings of the 4th International Workshop on Security Measurements and Metrics, pages 7–10, 2012.
[12] LwinKhinShar and HeeBengKuan Tan. Predicting common web application vulnerabilities from input validation and sanitization code patterns. Proceedings of the 27th IEEE/ACM International Conference on Automated Software Engineering, pages 310–313, 2012.
[13] Ibéria Medeiros, Nuno F. Neves, and Miguel Correia. Automatic detection and correction of web application vulnerabilities using data mining to predict false positives. Proceedings of the 23rd International Conference on World Wide Web, pages 63–74, 2014.
[14] Bertrand STIVALET Aurelien DELAITRE. Php vulnerabilities test suite. https://github.com/stivalet/PHP-Vulnerability-test-suite , 2014. Accessed: 2014-07-13.
[15] Peter ReutemannEibe Frank, Mark Hall and Len Trigg. Weka: Data mining tool. http://www.cs.waikato.ac.nz/ml/weka, 2013. Accessed: 2013-06-26. [16] Ian H. Witten, Eibe Frank, and Mark A. Hall. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011.