International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT217155

A Secure Software Specification Development Strategy for Enterprises : A Case Study Approach

Sifat Ali Sathio[1], Dr. Isma Farah Siddiqui*[2], Dr. Qasim Ali Arain[3]

[1]Department of Software Engineering, Institute of Information and Communication Technology, Mehran University of Engineering and Technology, Jamshoro, Pakistan

[1]sifat26.ali@gmail.com

[2]Associate Professor, Department of Software Engineering, Mehran University of Engineering and Technology, Jamshoro, Pakistan

[2]Isma.farah@faculty.muet.edu.pk

[3]Associate Professor, Department of Software Engineering, Mehran University of Engineering and Technology, Jamshoro, Pakistan

[3]qasim.arain@faculty.muet.edu.pk

Article Info
Volume 7, Issue 1
Page Number: 260-266

Publication Issue :
January-February-2021

Article History
Accepted : 20 Feb 2021
Published : 27 Feb 2021

ABSTRACT

Although Security is a non-functional requirement, it is a very essential requirement for software systems, to achieve secure software specification development for enterprises we need to find and fix vulnerabilities in the early phase of SDLC. For the successful achievement of secure software specification development in the software enterprise, the security of software application plays a very vital role. During the software development lifecycle, improper security can lead to thoughtful and serious consequences in any enterprise. In this paper, the case study approach is followed regarding the achievement of a secure web application, finding and fixing vulnerabilities in the early software development lifecycle, and applying the re-engineering process on a developed web application using the best security assessment model considering the literature review. Also, validation of the developed application is done with the help of Penetration testing.

Keywords : Requirements engineering, Software requirements, Software engineering, Software specifications, Secure Software Specification, Secure Software Development Life Cycle, Re Engineering, Vulnerability, Penetration Testing

I. INTRODUCTION

Software development is a rapidly expanding field since the 21st century [1]; the security of the software is a major issue for any organization. Initially, software applications used by very few users, as the passage of time, the number of users increased in the web application field. Due to this the security concern also increased.

Security of software application is an essential requirement for enterprises [2]. The security violated

through the internet network by cyber attacker is known as Cyber Security. There are several types of cyber-attacks, including Phishing, Man in Middle, Password attack, Denial of Service, Distributed Denial of Service, Malware, etc. through which cyber attackers breaches security of software application. To achieve quality software application, it is necessary to follow proper principles and standards introduced by various organizations such as ISO, IEEE, and CMM standards, etc. Using these standards, we can specify the functional and nonfunctional requirements, including security in the software development lifecycle phase.

To build security requirements, security needs and knowledge of vulnerabilities must be specified at the SDLC phase. The security of software application isn't addressed from the very beginning of SDLC [3] and it is compulsory to ensure that the software applications are secure by considering the security requirements at the phases of SDLC [4]. Mostly security handled with a reactive approach which is costly and may leave vulnerabilities, while a cost-effective proactive approach in which security is addressed throughout SDLC in software application like attack trees, misuse cases and UMLSec are used in the phases of software development lifecycle to analyze vulnerabilities.

Case study was conducted to find and fix the security risk i.e. vulnerabilities in the early software development lifecycle, finding and to propose the best security model based on the literature review, design and develop application, re-engineering with the proposed security model on the developed application and applying validation using Penetration testing.

The arrangement of this study is accordingly: In Section 2 Related Work was discussed, Section 3 describes Research Methodology, Section 4 contains the Results of this case study, Section 5 is regarding Discussion and the Section 6 presents Conclusion, and finally Section 7 shows the References.

II. RELATED WORK

In the field of software engineering research, several works are done by researchers related to secure software specification development, in which functional and non-functional requirements in software engineering [5], including non-functional characteristics i.e. security are identified. Based on these requirements [6] analysis security requirements for developing secure software applications are performed. Several software security approaches followed in SDLC [7] in order to find out vulnerabilities, [8] to evaluate the physical characteristics of a security requirement specification methodology according to the standards of enterprise. Security requirements [9] can be specified as the properties of the system and applied formal approach to the secure software specification model. Most of the researchers proposed the different approaches to achieve security, but still lots of work can be done in this non-functional requirement i.e. security.

III. RESEARCH METHODOLGY

Case studies are used to explore the knowledge from an environment, and explaining the working mechanism process [10]. To achieve secure software specification including functional and non-functional requirement i.e. security in a web application, we should follow a secure software development lifecycle, and after the deployment phase a penetration testing process performed to find the vulnerabilities at SDLC. Secure software development lifecycle an iterative process of requirement specification, design, coding and deployment of software application as shown in Figure 1: SDLC Phases.

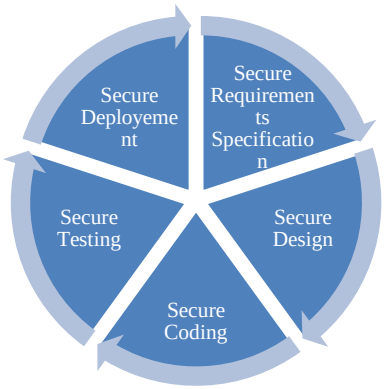


Figure 1 : SDLC Phases

Several phases of secure software development lifecycle, in Secure Requirements specification phase, functional and non-functional requirements specified. In secure design phase secure use cases developed while Implementation phase consist of identification of vulnerabilities and security modules developed. In testing phase several types of testing occurred i.e. Unit Testing, Functional Testing, and Penetration Testing. The last phase of secure SDLC is secure deployment which consists monitoring of requirements.

A case study approach followed in the methodology section to achieve a secure software specification development which includes several sections, including the i) Literature Review, ii) Identifying Security Assessment Model, iii) Web Application Development and iv) Penetration Testing, as shown in Figure 2: Methodology.

Figure 2 : Methodology

IV. Literature Review

Literature Review is the best technique to analyze the security risk in software requirement engineering i.e. vulnerabilities in web applications. For this case study a Literature Review conducted as described in Section II: Related Work. In this case study several research papers studied, the summary of literature review is described below.

Security is a decisive characteristic of software application; to develop secure software is to incorporate security from the very early phase of SDLC [3]. Several software security approaches followed in SDLC [7] in order to find out vulnerabilities [8] to evaluate the physical aspects of a security requirement specification methodology according to the standards of enterprise.

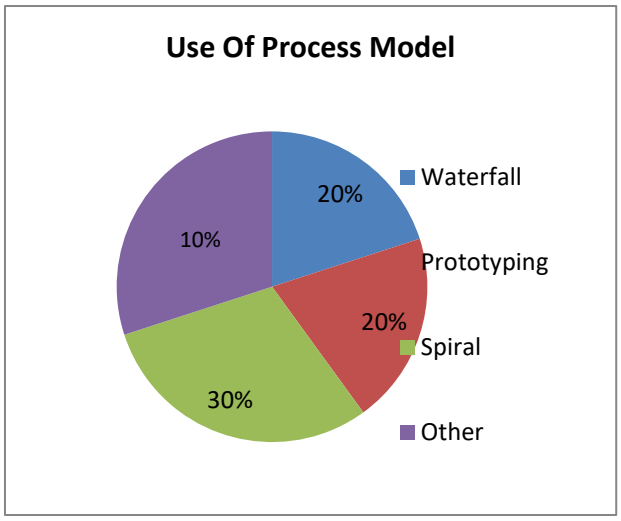
Security requirements [9] can be specified as the properties of the system and applied formal approach to the secure software specification model. Security risk i.e. vulnerabilities in different phases of SDLC while developing a software application should be removed at the early stage of SDLC otherwise security risk and cost factors affect the product. User

requirements i.e. information must be kept confidentially and data integrity, to verify the security in SDLC to eliminate the security vulnerabilities.

3.1 Identifying Security Assessment Model

Various SDLC models are created by software developers, universities and standards organizations [3]. Software development Secure software specification development needs to follow the best security assessment model. Several software models used in software application development in enterprises, i.e. Waterfall, Rapid Prototyping also called Prototyping, Spiral and other process models.

The Waterfall model is sequential step by step process model from an initial phase to the last phase of SDLC. While the Rapid Prototyping model from the requirement gathering phase, next to a prototype



application is built and presented to the clients by enterprises. If the client approves tested prototype, then the next prototype is built by software developers. In Rapid Prototyping the security of application can't be achieved due to fast development. Spiral Process Model follows an iterative approach in SDLC; the developer's starts with a small piece of requirements and goes through each phase of SDLC for those set of requirements via a risk analysis process. Developer's implements functionality for additional requirements in spiral until application is ready for deployment. Most of the enterprises use Spiral Process Model for SDLC in web application development because it is iterative in nature, and it provides risk analysis process.

Figure 3 : Process Models

3.2Web Application Development

With the help of Literature Review and identified security assessment model, the next step of our Methodology Section is Web Application Development. Here we develop a software application to perform the testing process using Penetration Testing for finding the vulnerabilities using OWSAP Tool.

3.3 Penetration Testing

Penetration Testing or also called Pen testing is a process of discovering the vulnerabilities to determine the loop holes in a software application. Penetration Testing phases include Planning, Scanning, Execution, Risk Analysis and Output. Penetration testing has various types i.e. targeted testing, external testing, internal testing, blind testing, and double blind testing. Pen testing can be done using Black Box Testing, Gray Box Testing and White Box Testing.

Penetration Testing can be performed using various tools i.e. Metasploit, NMap, Nessus Vulnerabilities Scanner, THC Hydra, John the Ripper (JTR), Social Engineering Toolkit, WireShark, The Web Application Attack and Audit Framework (W3AF) and the Open Web Application Software Project (OWASP).

V. RESULTS

In this section, we discuss about the aims and objectives of research paper, the experimental work of a case study are described below:

For secure software specification development it is necessary to follow the proper principle and standards that are introduced by various organizations some of them are ISO, IEEE and CMM. Most commonly waterfall, spiral and prototyping security assessment models used for web application development, but results shows that the spiral process model is mostly use for development, as shown in Figure 3: Process Models.

4.1 Security Assessment Model

Several assessment models are used by various enterprises for identification of vulnerabilities as already discussed in method in section 3.4 Penetration testing. Nowadays mostly enterprises, prefers the Open Web Application Software Project (OWASP) Framework over other tools.

The OWASP Framework is a java based framework that contains thousands of test programs used analysing the web application for identifying vulnerabilities [11]. OWASP provides Top Ten (10) list of threats i.e. vulnerabilities which can affect the security of a web application. These vulnerabilities are listed below:

1. SQL Injection (SQLi)
2. Broken Authentication
3. Sensitive Data Exposure
4. XML External Entities (XXE)
5. Broken Access Control
6. Security Misconfiguration
7. Cross-Site Scripting (XSS)
8. Insecure Deserialization
9. Using Components with Known Vulnerabilities
10. Insufficient Logging & Monitoring

These threats affect the security of a web application, so with in testing phase of SDLC using penetration testing these vulnerabilities are identified by ZAP software tool provided by OWASP organization in the software application. Threats can be of any type, just like potential danger of user input in SQL statement, as shown in Figure 4: Vulnerability Attack

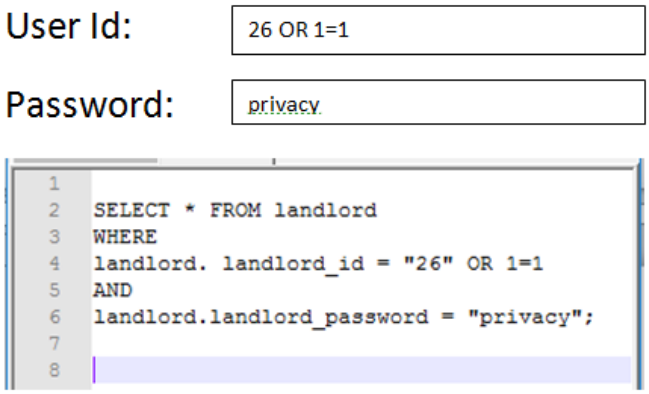


Figure 4 : Vulnerability Attack

We can fix vulnerabilities by using different techniques to make a secure software application, just like applying proper encryption methods, fixing the vulnerabilities which are identified by OWASP framework by tactics, makes a secure software application.

VI. DISCUSION

This case study examines that secure software specification development strategy for enterprises needs to handle at the SDLC phase of software application. The literature reviews statistics shows that for a secure development model most of the organizations follows spiral process model for secure requirement specification. An OWASP framework used for identification of vulnerabilities, a penetration testing is performed at the testing phase of SDLC using ZAP software application tool in the software application development. The main purpose of this research is to find out and fix the loopholes in a software application.

VII. CONCLUSION

This research identifies the various secure requirement specification in the development of the software application. Mostly, enterprises use the spiral process model in SDLC phases for secure software development. Literature review examined for finding and fixing of vulnerabilities and a case study investigated regarding the security risk i.e. vulnerabilities in web applications.

To achieve research aims and objectives, an Open Web Application Software Project (OWASP) tool for identification of vulnerabilities and validated to achieve the security into SDLC phases of software application. With the incorporation of the proposed security assessment model, security can be incorporated into phases of SDLC, so that security risk reduced at the early stage. Hence, the secure requirement specification is a vital process which must be implemented into different phases of SDLC in software applications.

In this case study, we proposed an OWASP-based security tools to identify and fix the vulnerabilities in software specification within enterprises. In future researchers can also explore this framework for security risk analysis in software application and suggest additional tools in the field of cyber security.

VIII. REFERENCES

[1]. Mamdouh Alenezi, Amir Shahab, Muhammad Nadeem & Raja Asif Wagan (2020). An automated approach to fix buffer overflows. Int J Elec & Comp Eng, Vol. 10, No. 4.

[2]. Qian, K., Parizi, R. M., & Lo, D. (2018, December). OWASP Risk Analysis Driven Security Requirements Specification for Secure Android Mobile Software Development. In 2018 IEEE Conference on Dependable and Secure Computing (DSC) (pp. 1-2). IEEE.

[3]. Khan, M. U. A., & Zulkernine, M. (2008, July). Quantifying security in secure software development phases. In 2008 32nd Annual IEEE International Computer Software and Applications Conference (pp. 955-960). IEEE.

[4]. Karim, N. S. A., Albuolayan, A., Saba, T., & Rehman, A. (2016). The practice of secure software development in SDLC: an investigation through existing model and a case study. Security and Communication Networks, 9(18), 5333-5345.

[5]. Chung, L., Nixon, B. A., Yu, E., &Mylopoulos, J. (2012). Non-functional requirements in software engineering (Vol. 5). Springer Science & Business Media.

[6]. Salini, P., & Kanmani, S. (2016). Effectiveness and performance analysis of model-oriented security requirements engineering to elicit security requirements: a systematic solution for developing secure software systems. International Journal of Information Security, 15(3), 319-334.

[7]. Mohammed, N. M., Niazi, M., Alshayeb, M., & Mahmood, S. (2017). Exploring software security approaches in software development lifecycle: A systematic mapping study. Computer Standards & Interfaces, 50, 107-115.

[8]. Sravani Teja Bulusu, Romain Laborde, Ahmad Samer Wazan, Francois Barrere & Abdelmalek Benzekri. (2018). Applying a Requirement Engineering Based Approach to Evaluate the Security Requirements Engineering Methodologies. Researchgate.

[9]. Rouland, Q., Hamid, B., Bodeveix, J. P., & Filali, M. (2019, November). A Formal Methods Approach to Security Requirements Specification and Verification. In 2019 24th International Conference on Engineering of Complex Computer Systems (ICECCS) (pp. 236-241). IEEE.

[10]. Abdul Karim, Nor & Albuolayan, Arwa & Saba, Tanzila & Rehman, Amjad. (2016). The

practice of secure software development in SDLC: an investigation through existing model and a case study: The practice of secure software development in SDLC. Security and Communication Networks. 10.1002/sec.1700.

[11]. Burato, E., Ferrara, P., & Spoto, F. (2017). Security analysis of the OWASP benchmark with Julia. Proceedings of ITASEC, 17.

Cite this article as :

Sifat Ali Sathio, Dr. Isma Farah Siddiqui, Dr. Qasim Ali Arain, "A Secure Software Specification Development Strategy for Enterprises : A Case Study Approach", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7 Issue 1, pp. 260-266, January-February 2021. Available at doi : https://doi.org/10.32628/CSEIT217155
Journal URL : https://ijsrcseit.com/CSEIT217155