Big Data Analytics in Secure Software Development Life Cycle

Sithara Sasidharan[1] **Geethumol P V**[2]
[1,2]Assistant Professor
[1,2]Department of Computer Science & Engineering
[1,2]Mount Zion College of Engg Kadammanitta, India

Abstract— A central and critical feature of the computer security problem is a software problem. To improve software security is to provide security process in SDLC processes .In each phase of secure software development life cycle, security testing is needed. This paper proposes an integrated security testing frame work SSDLC in Big data analytic approach. In our proposed framework, system can integrate various security testing tools and supports secure activities. Using this frame work can provide quality, stable service and security in big data.

Key words: Big Data, SSDLC, Software Development Life Cycle

I. Introduction

In most corporations and large organizations, security is the domain of the people who set up and retain firewalls, intrusion detection systems; today, most projects and project managers are responsible for system operation respond to the Internet-based attacks. Threats use defects of system and cause risk to attack a system [4]. The software development life cycle (SDLC) is a framework defining tasks performed at each step in the software development process. Small changes in the software development life cycle can improve security [1].

Many security testing tools are used to detect security defects and vulnerabilities. In this paper, we propose an integrated security testing framework for secure software development life cycle. It is also relevant for developers and managers looking for information on existing software development life cycle processes that address security. An organization can compare their practices to the model to identify potential areas for improvement. An organization that wants to obtain or develop a particular type of security product defines their security needs using a protection profile.

Once the software is thought secure enough for use, it can be implemented to test real-world usability, and enters the maintenance phase. The maintenance stage allows the application to be adjusted to organizational, systemic and utilization changes. Two or more data sets collected using a mix of methods is analyzed separately. The findings are then combined or integrated.

Requirement analysis is the most important and fundamental stage in SDLC. Once the requirement analysis is done the next step is to clearly define and document the product requirements [8]. More than one design approach for the product architecture is proposed. If the design is performed in a detailed and organized manner, code generation can be accomplished. The testing activities are mostly involved in all the stages of SDLC. Big data analytics examines large amounts of data to uncover hidden patterns, correlations and other insights.

Advances in Big Data Analytics Data- driven information security detects fraud detection and anomaly based intrusion detection systems. Big data tools are also particularly appropriate to become fundamental for advanced persistent threat detection.

II. Related Work

The Microsoft Security Development Lifecycle (Microsoft SDL) [9] is a software development process used and proposed by Microsoft to reduce software maintenance costs and increase reliability of software concerning software security related bugs. Microsoft's methodology is maybe one of the most used in the commercial area. Microsoft SDL involves modifying a software development organization’s processes by integrating measures that lead to improved software security.Fig.1 shows the key activities in Microsoft SDL.

Training	Require-ments	Design	Impleme-ntation	Verifi-cation	Release	Response
Core security training	Create quality gates	Threat modeling	Static analysis	Dynamic analysis	Final security review	Execute hidden response plan

Fig. 1: The Microsoft Security Development Lifecycle

The Open Web Application Security Project (OWASP) [10] is an online community which creates freely-available articles, methodologies, documentation, tools and technologies in the field of web application security. The OWASP Software Assurance Maturity Model (SAMM) [1] is a usable framework to help organizations prepare and implement a strategy for application security to the specific business risks.

III. Proposed Framework

In the software development life cycle, security plays a very important role. This paper proposes an integrated security testing framework for SSDLC. It also proposes big data analytics in evaluation.

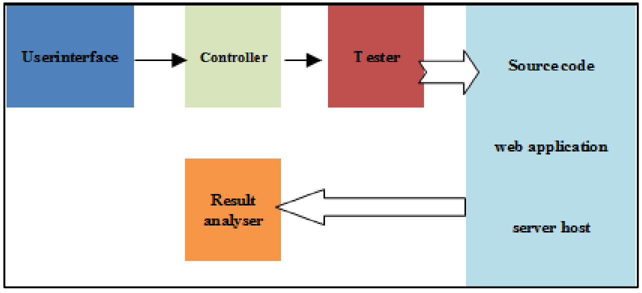


Fig. 2: Implementation architecture of prototype system

In the above fig, contains the phases of the framework. In phase 1, requirement development, to analyze the activities and practices of SSDLC and develop corresponding parameter for security issue. By the definition of SSDLC, to construct security parameter by identifying the security issue need to implement. In phase 2, test case construction, design test case and test plan to meet security guideline of SSDLC.

In phase 3, tool integration, build up a common interface to integrate different security testing tools and test will be executed by using corresponding automatic testing tools. In phase 4, result analysis, big data analytics data model to represent testing results from several testing tools and improve testing results analysis.

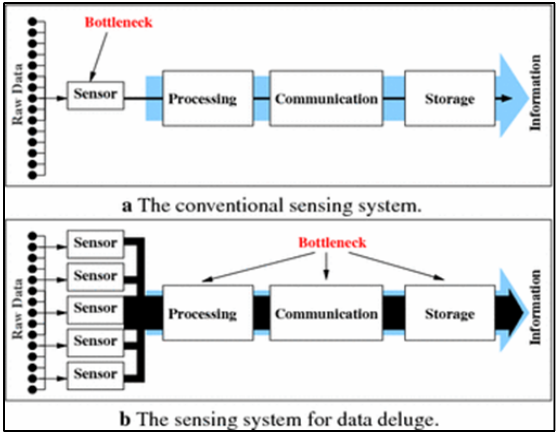
After testing the result size will be large, and the data will be occurred in multiple types and captured from different sources, respectively. The selection operator, usually plays the role of knowing which kind of data was required for data analysis and select the relevant information from the assembled data or databases; thus, these assembled data from different data resources will need to be integrated to the target data. The preprocessing operator, plays a different role in dealing with the input data which is intended at detecting, cleaning, and filtering the unnecessary, inconsistent, and incomplete data to make them the useful data. By the transformation operator. The methods for reducing the complexity and reducing the data scale to make the data useful for data analysis.

Most data analysis methods [7], have limitations for big data. Most data analysis methods are not for large-scale and complex dataset. Most traditional data analysis methods cannot be dynamically used to for different situations. The traditional data mining algorithms assume that the format of the input data will be the same.

IV. Big Data Analytics

Nowadays, the data that need to be analyzed are not just large, but they are composed of various data types, and even including streaming data. Although it seems that big data [6], makes it possible for us to collect more data to find more useful information, the truth is that more data do not necessarily mean more useful information. It may contain more uncertain or irregular data. Therefore, several new issues for data analytics come up, such as privacy, security, storage, fault tolerance, and quality of data.

The overflow of input data is the primary thing that needs to look because it may paralyze the data analytics. The bottleneck of big data analytics will be shifted from sensor to processing, communications, storage of sensing data, as shown in Fig. 3.



a The conventional sensing system.

b The sensing system for data deluge.

Fig. 3: Comparison between traditional data analysis and big data analysis

Sensors can collect much more data, but when uploading such large data to upper layer system, it may produce bottlenecks everywhere. The incoming data may use different types or have unsatisfactory data, how to handle them also take another issue. By using domain knowledge to design the preprocessing operator is a feasible solution for the big data analytics.

Sampling and compression [8], are two representative data reduction methods for big data analytics because reducing the size of data makes the data analytics computationally less expensive. To make it possible for the compression method to efficiently compress the data, a capable solution is to apply the clustering method to the input data to divide them into several different groups and then compress these input data according to the clustering information. In *Clustering algorithms* the characteristics of big data still brought up several new challenges for the data clustering issues.

The big data clustering is divided into two categories: single-machine clustering and multiple-machine clustering. This means that traditional reduction solutions can also be used in the big data age because the complexity and memory space needed for the process of data analysis will be decreased by using sampling and dimension reduction methods.

V. Conclusion

Several studies attempt to present an efficient or effective solution from the perspective of system .In recent years; SSDLC is widely discussed in developing secure software. There are many activities and practices are proposed to achieve security goals of project. This paper proposed an integrated security testing structure for secure software development life cycle. The results were evaluated using Big Data analytics. In the future studies, will analyze in depth about developing performance of programmers and security tools.

References

[1] A. D. Kent, “Comprehensive, multi-source cyber-security events,” Los Alamos National Laboratory, 2015.

[2] N. Adams and N. Heard, Data analysis for network cyber-security. Imperial College Press, 2014.

[3] A. D. Kent, “Cyber security data sources for dynamic network research,” in Dynamic Networks in Cyber security. World Scientific, 2016.

[4] I. Friedberg, F. Skopik, G. Settanni, and R. Fiedler, “Combating advanced persistent threats: From network event correlation to incident detection.” Computers and Security, vol. 48, pp. 35–57, 2015.

[5] https://www.microsoft.com/en-us/sdl/ [Last accessed 1 May, 2016] "Microsoft Security Development Lifecycle",

[6] McGraw, G., & Viega, J. (2001). Building Secure Software. Addison Wesley.

[7] Tung, Yuan-Hsin, Chen-Chiu Lin, and Hwai-Ling Shan. "Test as a Service: A framework for Web security TaaS service in cloud environment." Service Oriented System Engineering (SOSE), 2014, IEEE 8th International Symposium on. IEEE, 2014.

[8] Wakchaure, Mr Manoj Ashok, and Shashank D. Joshi. "A Framework to Detect and Analyze Software Vulnerabilities: Analysis Phase in SDLC." Journal of Modern Electronics 4.1-2 (2015).

[9] Tondel, I. A., Jaatun, M. G., & Meland, P. H. (2008). Security Requirements for the Rest of Us: A Survey. IEEE Software, 20-27.