International Journal of Computer & Information Technologies (IJOCIT)
www.ijocit.ir & www.ijocit.org
ISSN = 2345-3877

An Approach for Assessment Ensuring the Development of Secure Software in Prototyping Process

Ali Taati[1], **Hossein Erfani**[2]

MSc, Department of Electrical, Computer & IT, Zanjan Branch, Islamic Azad University[1]

University of Science and Culture Rasht Branch[2]

Ali.taati.ch@gmail.com[1] , *erfani@usc.ac.ir*[2]

Abstract:

Nowadays, network security evaluation is not an only scanning of open ports, and explores the behavior of software as a key component of the system is essential and critical. Much software beneficial after production and in the process of applying thinks about the security of their applications and often it is sufficient to perform a penetration test. This increases the cost of fixing security flaws in the publication process, case and sporadic clashes with security at the application level. Inaccurate and incomplete understanding of the security requirements of software stakeholders, lack of proper management of changes, and security policies cause reduces the level of confidence. Security assessment should be much deeper than penetration testing and at the application layer even include functional security testing measures. In this paper, first we need to evaluate the safety and security of the software and a method will be provided for evaluating the safety level of the software. Using this methodology, the software security requirements of stakeholders are carefully identified and then according to the application and secure software development check list metrics, an acceptable level of reliability is achieved.

Keywords: Software Security Assessment, Software Reliability, Software Reliability Metrics

1. Introduction

In this paper, first software security and also Software vulnerabilities are described especially in the design phase. To this point some reports are expressed about the cost based on the time of the error detection and

vulnerability of the software and hardware is expressed and after that the proposed method for evaluating the expected reliability of organization is stated. For the purpose of comparing, the level of trust and real confidence level of the checklist is drawn and the valuation on each organization's software security metrics explained and finally the advantages of this method are discussed with respect to the organizations' view.

2. The Importance of Application Security Assessment

What is more importance in the field of information technology is the security key because information and communication without security parameters can make a lot of risks.

Especially in applications that their information is highly confidential to organizations and continues their business depends on this information. In this regard a study is done by the NIST Institute; Figure 1 shows the relationship between the costs of the security error detection based on its discovery time in each developmental stages.

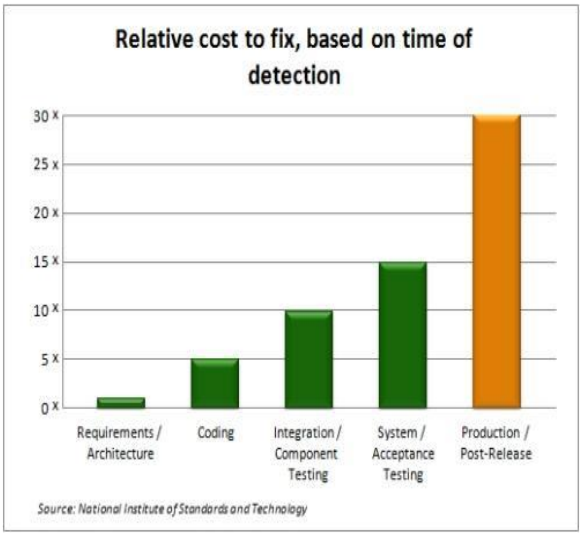

Figure1. The Cost is based on the Error Identification

So it is necessary that software which is used in organizations nowadays is assessed with high accuracy in the software development process and their actual confidence level fulfill the organizations' expected target level of confidence.

Here in confirming the above text; facts and figures are expressed till the importance of the affair gets clear more:

• More than 75 percent of today's influential and security attacks are due to software vulnerabilities

• Over 70 percent of security vulnerabilities are in the user level and not at the network level

• About 64 percent of software developers fail to develop security.

• If only 50 percent of vulnerabilities remove in the software before deployment, 75% of software development costs will be reduced.

2.1. Software Vulnerabilities

Figure 2 shows the percent of software vulnerabilities that are related to the software. Generally there are two types of software vulnerabilities, the first one related to the fundamental weaknesses of the applications that these security weaknesses cannot be eliminated. Therefore, to deal with them first it is necessary to identify them and then by using a layered defense strategy, they reduce the probable abuse. There is another type of security weaknesses in the design and implementation of software. Usually these kinds of security weaknesses are preventable .This is largely achieved in compliance with security requirements during the design, implementation and testing of software. It is necessary for those vulnerabilities, which are hidden during the installation of software and can be identified after software production; instantly relevant amendments create and install the patch immediately and otherwise strategies should be presented for reducing the risk until the relevant amendments to be prepared. For this reason, installing on time amendments is one of the security requirements for the software operation. Then, vulnerabilities in software design are described.

2.2. Vulnerabilities of Design

Vulnerabilities in software design stem from a fundamental or an oversight infrastructure in the design of software. If there is a flaw in the design, the software won't be safe definitely. Because the software does what it is designed for and in this case they are designed to do the wrong thing. These types of defects are due to environment in which the software is running. These vulnerabilities are typically known as high-level vulnerabilities, architectural defects or problems in the requirements and limitations of the program. For example, the TELNET protocol has been designed to connect to a remote device. In terms of design, this protocol with respect to relying on insecure communications has been vulnerable. This protocol is reliable only in environments where network infrastructure is secure, but in environments such as the Internet can be very dangerous.

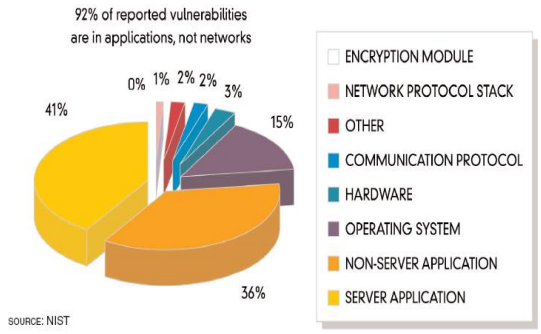

Figure 2. Software Vulnerabilities Report

3. The Proposed Method

The proposed method in figure 3 is based on two very important principles, for understanding the role of organizations' software security applications and believes that secure software should be defined based on the organizations' framework and software framework. This means that the target level of an organization's security of different software related to the business and its importance in the specified field.

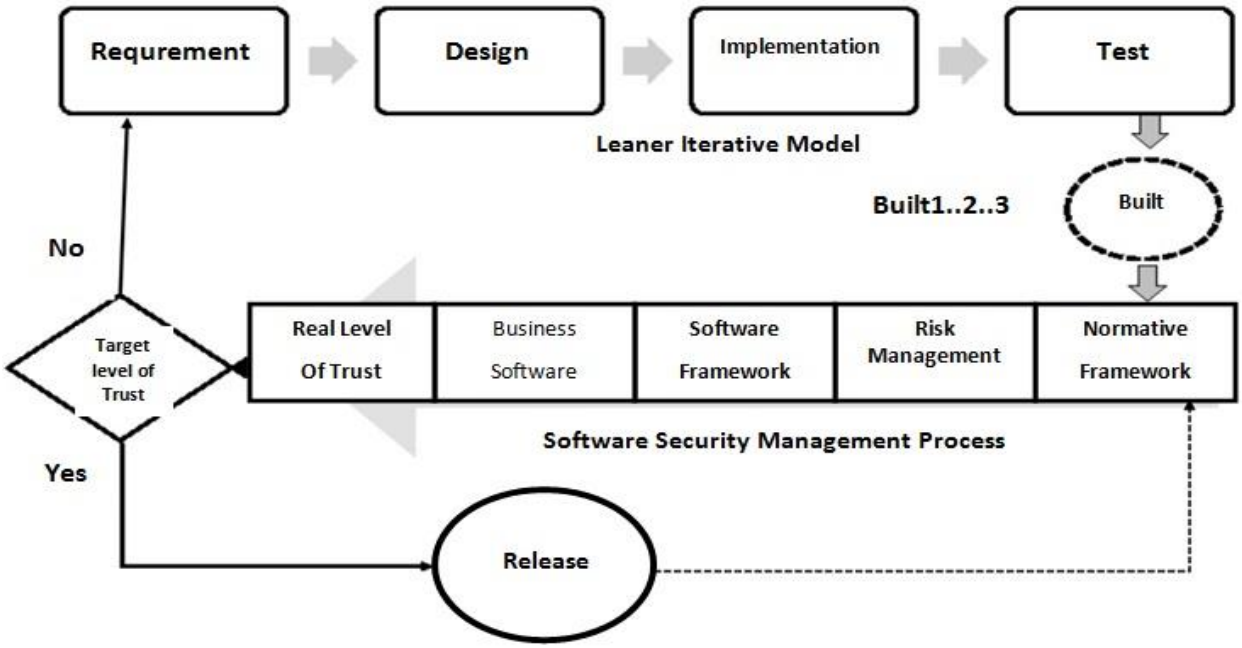

Figure 3. Suggestion Method

As you see in the proposed method (in Figure 3) when the applications' preparations were prepared, first the functional requirements of software is prepared and security requirements are documented as well as them. When software design is defined according to the requirements specified by the developers, the implementation stage and testing is done based on a model that finally produced

prototype software. The software development team must deliver documents and sample software to the organization's security management, that this team-based on specified process and metrics and the rules of software development, evaluated the software and compare the actual confidence level of application with the confidence level of the target (based on a checklist designed), If the expected confidence in this assessment is met, the software will enter to the distribution phase and otherwise prototype for future produce is returning to the development team with evidence of pathology.

3.1. Evaluation of the Actual Level of Security

At this stage, which is in the design phase of the business project, Software Security Assessment Committee will decide whether the application re-enter new construction or should be sent to the execution phase. This process is based on the ISO 27034 which is describes software security application in two main sections:

A: organizations' framework: consists of a set of rules and regulations and organizations' site in the business.

B: organizations software Framework: include organizations' software rules and software components that should be considered in the organization.

3.2. Check List of Software Security Assessment

The prepared check list in the form of Table 1, is according to securing the proposed method. In this check list each of the software prevention security vulnerabilities are rated.

			0	1	3		
		Score				Vulnerability assessment	
A MinimumScore Goals By The Target Level of Trust		1		v		The correct definition is based on the politics of identity	1
		0	v			Prepare a list of security requirements for software	2
		1		v		Determine the need for fault tolerance list	3
Risks		0	v			The definition requires secure communication between components of the system software	4
Down		0	v			Definition lists the system requirements for data encryption	5
Average		0	v			Design policies lockout feature	6
above		0	v			Software requirements management of security incidents	7
		1		v		Software requirements for data consistency check mechanism	8
		0	v			Develop a plan to deal with the needs of the vulnerable security policy	9
		3			v	List of access policy management requirements	10
		1		v		Members need to input validation security software	11
		0	v			Producers need to assess security software list	12
		0	v			Define information exchange requirements encryption algorithm	13
		0	v			Defining the need for secure network communications	14
		0	v			Model of the application is approved threats	15
		0	v			Respect to the design of software architecture	16
		0	v			The use of appropriate algorithms encryption	17
		0	v			Allow unlimited attempts to implement authentication	18
		0	v			Delivery mechanisms and the establishment of secure software	19
		3			v	Mechanisms for user groups with different access levels	20
		1		v		Mechanisms for dealing with error messages unmanaged	21
		0	v			Protect sensitive information during transmission application	22
		0	v			Software security coordination with the Organization for Security Policy	23
		0	v			Developing mechanisms for data encryption software	24
		1		v		User access control management mechanisms	25
Total minimum rating target level of Trust=51		0	v			Inspection mechanism Security Software	26
	51	12				Total scores (real confidence level)=12	
Result	Rejection	Inclusion/ exclusion					

Table 1: List of Software Evaluation Assurance Level

In this checklist in Table 1 in the case of compliance with any instance of the security measures green indicator is checked and 3 points awarded and if there is a defect in it the yellow color appears with a rating and if the indicator is not met, the red color with a zero point awarded. It is noted that the rating and metrics valuation can vary depending on the kind of the application and organization. For example, negative score is considered to red. According to organizations' security targeted level, each phase is marked with a minimum rating, and the software must obtain the least security score. Earn points by the software is the real confidence level of the application.

In check list in Table 1 the elements of a security assessment of the requirements and design is examined that it can be variable

based on the type of software and business environment.

4. The Advantages of This Method

- Application security assessment of the organization's security management based on the confidence level of the target

- Design organizational security software compatible with policies of the organization's software

- Manage time and cost within a Defined framework

- Existence of high correlation between the Applications and the necessity of their recognition

- Increased safety due to the commitment of Software Security Assessment

- Avoid the careless security of software development team

- make confidence in the end-user in use of a software system

5. Conclusion

So in this article we come to an important fact that any organization to have secured software in order to sustain their business should have the organizational software security management, with creation an appropriate software security by the software development team and doing security evaluation according to the presented checklist a secure software development life cycle can be achieved. As a result, by evaluation and control of software security vulnerabilities and enjoying proper threat modeling according to the target confidence level can close the actual level of software security to the desired target level.

Resources

[1] T.lanowitz, (2005), “security at the application level”,gartner research .
[2] Microsoft development research http://download.microsoft.com/download/f/4/a/f4a67fc8-c499-461d-a025 [Accessed on January 2015]
[3] S.Sima,(2004),”Security at the Next Level , Are Your Applications Vulnerable”SPI dynamics.
[4] Reavis Consulting Group, LLC, (2013)" The Emergence of Software Security Standards: ISO/IEC 27034-1:2011 And Your Organization", http://www.microsoft.com/global/eu/RenderingAssets/pdf/The%20emergence%20of%20software%20security%20standards.pdf [Accessed on January 2015]
[5] Kakali Chatterjee • Daya Gupta • Asok De, (2013)," A framework for Development of Secure Software’’", CSIT (June 2013), Vol. 1, Issue 2: pp.143–157

[6] Muhammad Shakeel Faridi, Tasleem Mustafa and Fahad Jan,(2012)," Human Persuasion Integration in Software

Development Lifecycle (SDLC)", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 4.
[7] Malik Imran Daud, (2010),"Secure Software Development Model: A Guide for Secure Software Life Cycle", proceeding of the international multi-conference of engineers of computer scientists IMECS Hong Kong.

