



Stability: Abstract Roadmap of Software Security

Kavita Sahu¹, Raj Shree²

^{1,2}Department of Information Technology

Babasaheb Bhimrao Ambedkar University, Lucknow (U.P.), INDIA

Abstract: *With the advent of new technologies and software, software industry is becoming more competitive. The users are pressurizing software developers to deploy a fully secured system. Increasing requirements of software security often leads to the development of many successive versions of software. Software products are growing more sophisticated, the writing of latest versions has become complex and time consuming activity. For software security reasons it is necessary to understand the nature of security and the best way to do it is to identify new security factors and analyze them. In this paper stability is discussed as a security factor of software security. Predicting software stability with enhanced new version is a key factor for software security. Stability, like most security factor, is a complex phenomenon and its prediction is a real challenge.*

Keywords: *software security, security attributes, stability, design phase.*

I. Introduction

Computer and internet are being very popular and basic need for day to day life. These needs include sharing of data globally, transactions etc. With the wide use of computer, software becomes complicated and large-scale. Complicated software results more security problems. Software security is the ability of software to prevent system from failures when it is attacked [1]. It is evident from literature survey that the plenty of work has been done in the area of software security. In the following section, some relevant works have been discussed with their findings in order to strengthen the fact that need and importance of having security is continuously increasing at the time of development [4]. In 2008, Beznosov and Chess presents a report on security for U.S. An industry perspective on the secure software challenge. This report was based on the state of the practice and recent advances in engineering secure software for the wide range of industrial application domains. Author discussed the needs [4] and importance of security requirements analysis and threat analysis and suggested that developer needs to properly called and analyze security requirements [7].

In software economic and environment factors may affect it, stability and security as well. Software security requirements mainly include data confidentiality, integrity, availability, authentication, authorization, and access control. To develop a fully secured system, all phase contributes the same in security hence this is necessary to secure each and every phase of a software. In the context of all this investigation we considered stability as a security factor.

The approach in this work has been taken for software security and security factors. As security factors effects the security of software directly hence enhancing the factors will improve the quality of software. Stability as a security factor effects the security which is defined later in this paper. Stability is the ability of secured software of being stable for a long time period.

II. Software security

Security decisions can be incredibly complex, even when we look at a relatively small and simple problem. Software security is about developing software to behave as preventive in the presence of various types of attacks, even though in the real world, software failures usually happen unexpectedly that is, without intentional mischief [3]. Software security feature can't be added through the addition of sets of features, it must be designed and integrated with the every phase of the software development life cycle.[4]. Undesirable threats take advantage of the weaknesses of the software and hardware and affect the factors of security as privacy, authentication, availability and integrity. Software industries are suffering from these undesirable malicious attacks before software deployment. Hence it is necessary to invigilate the each and every factor of security during the design phase.

III. Security factors

Security is considered as part of software's non-functional requirements. Such software quality attributes include security, maintainability, performance, reusability, and reliability [2]. Most of these attributes have been

studied and measured extensively. The typical model adopted for security has only three security factors which are also called the three pillars of software security these are confidentiality, integrity, availability. These are the three concerns of most endeavors towards providing secure software. These three factors are described as follows.

A. Confidentiality

Confidentiality is an application means that the private and sensitive data handled by the application read by anyone who is not explicitly authorized to view it. It also limits the access of information in right direction and prevention of disclosure of information to unauthorized users. Confidentiality is broader concept of privacy which limits access to individual's personal information which requires a trusted binding mechanism of design and its total supporting services and related components. It ensures that there is no chance of leakage of information [5].

B. Integrity

Integrity means that the data processed by an application is modified by any unauthorized channels. In integrity effect the whole information is also validated which insure the changes having not any effect of the integrity. As more and more monetary exchanges occur over the internet, integrity is crucial [8].

C. Availability

Availability speaks to the software's ability to remain operational even in the face of failure or attack. This is typically the hardest attribute to guarantee because there are forces at work that are beyond the control of the system itself. Availability is also related to maximum strength of protection because the information can be traced by intruders to violate security [9].

IV. Stability as a security factor

When any change to the security of software is done than its original design is preserved or not, that is called stability. Stability is the state of being steady or ability to maintain a firm position for a time-period. It is usually considered to be the most important dimensions of dependability. If software is unreliable, it is difficult to ensure software safety or security as they may be compromised by software failures. When the stability will be applied to the software it will be auto-stable. Changes to the software are in frequent. Controlled, security, if changes to the software should not invalidate automated. When any software updated then some attributes of that security are changed but some attributes does not change their behavior, because they are stable when they were constructed. Stability is the software attribute that reflects the software's ability to operate, normally or abnormally without threatening people or the environment. When stability is an important and essential attribute of security which defines all other attributes of software security and components [10].

V. Stability in context of other software attributes

A. Stability in context of authorization

The process of determining that a person who is requesting for an authorization is allowed to receive a stable service or performance operation. If it is not getting the stable service that means that service will not be secured for a long duration.

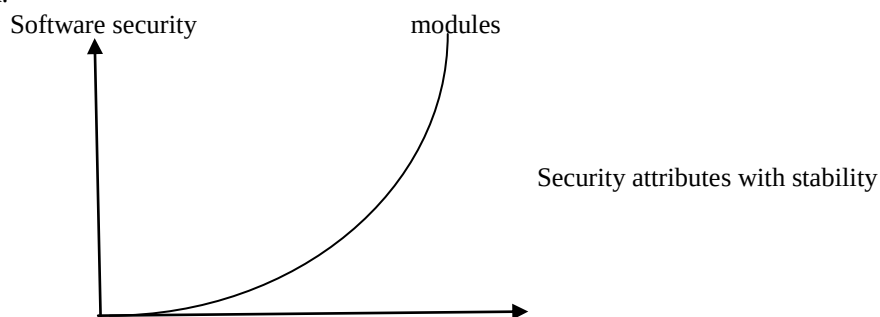


Fig. 1: Security and security attributes with modules

B. Stability in context of availability

Assuring information of data and communication s services will be stable for use when expected. Its information of data must be kept available stable to authorized persons when they need it.

C. Stability in context of reliability

Reliability ensures that application performs stable operation without failure in specific environment. Reliability factor is very important in term of stability because as much the system's reliability increases its security also increases.

D. Stability in context of survivability

Survivability ensures that application of software providing stable and essential resources to users even in the presence of an attack. Application can't have failure points with stability and it will survive for a long duration.

VI. Impact analysis of security factors

Take an example of medicines, as antibiotic medicines prevents the viruses to increase, they do not kill them as well as stability as a security factor gives the security stability. To make the all attributes stable in software security, it should be considered as a factor [11]. As it gives all the security factors stability hence it is complement of every attribute of software security which belongs to C.I.A as showed in the given figure.

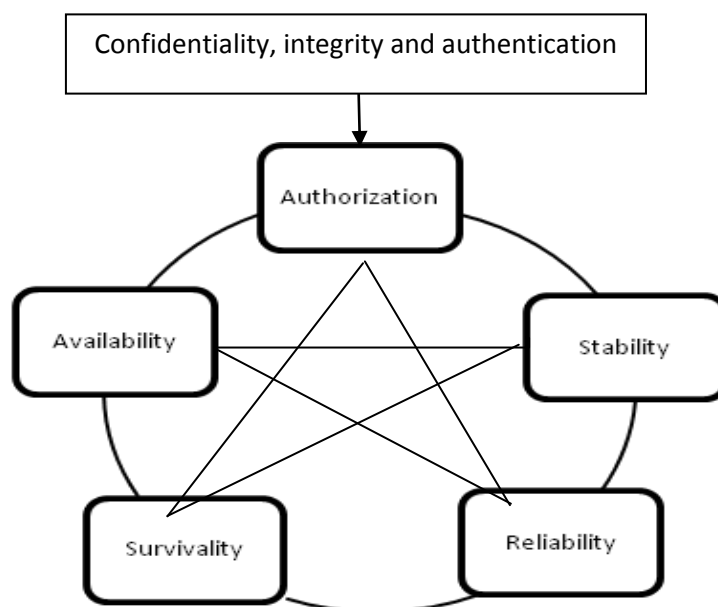


Fig. 2: Relationship between Security attributes.

VII. Relation between quality, security and stability

The demand for quality software continues to intensify due to our society's increasing dependence on software and internet. So the software security is being also very important phase in software development. Goal of software development is to deliver high quality software which can be achieved only by measuring and estimating its security factors. Stability as a quality factor relates with its estimation during development phase of a software. As early the software security factors are identified and measured the security of software is assured that much early. Hence it states stability factor plays an important role in defining security as well as quality of a software.

VIII. Discussion

Development of security design and architecture for software is not a onetime construction process, it is based on reuse of existing security specifications and sub factors. [6] There is common agreement between researchers and security practitioners to integrate security at the preliminary stage of software development life cycle in order to develop secured software. Security estimation and upgrading is possible with the help of finding new security attributes which directly affects the security features of software. Security attributes are found out and its applications are responsible for the successful delivery of secured software.

IX. Conclusion

Software security is a complicated issue. Security attributes play an important role in software security estimation. This research paper studies the report on a security factor which is stability, which helps a developer to understand the security in SDLC to develop more secure software. This paper concludes that development of secured software depends on the software attributes and it finally affects the quality of software. Stability as a security factor effects the quality of a software positively if it is considered in the early stage of development.

References

- [1] Gary McGraw, Bruce Potter. "Software Security Testing". IEEE Security & Privacy, 2004, 2(5):81-85.
- [2] A. Sachitano, R. O. Chapman, and J. A. Hamilton, "Security in software architecture: a case study," in Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop, 2004, pp. 370-376.

- [3] Gary McGraw, "Software Security: Building Security In", Addison Wesley Software Security Series, ISBN-13: 978-0321356703, Edition: 1, 2006.
- [4] S. Chandra and R.A. Khan, 2008. Object Oriented Software Security Estimation Life Cycle-Design Phase Perspective. Journal of Software Engineering, 2: 39-46.
- [5] G.H. Walton, T. A. Longstaff, R.C. Linder, Computational Evaluation of Software Security Attributes, IEEE, 1997.
- [6] C. Wang and W. A. Wulf, "A framework for security measurement" Proc. Of National Information System Security Conference, 7-10 Oct 1997, pp: 522-533.
- [7] Konstantin (Kosta) Beznosov and Brian Chess, "Security for the Rest of Us: An Industry Perspective on the Secure-Software Challenge", Journal IEEE Software Volume 25 Issue 1, January 2008 Pages 10-12.
- [8] R.A.Khan and S.A.Khan, "A Roadmap For security", IJCSIT, VOL 1, ISSUE 1, JUNE 2010.
- [9] S. A. Khan, R. A. Khan, "Securing Object Oriented Design: A Complexity Perspective", International Journal of Computer Applications (0975-8887) Volume 8- no.13, October 2010.
- [10] Charles B. Haley, Jonathan D. Moffett, Robin Laney and Bashar Nuseibeh, "A framework for security requirements engineering". In SESS'06, Shanghai, China, ACM, May 2006.
- [11] C. Wang and W. A. Wulf, "A framework for security measurement" Proc. Of National Information System Security Conference, 7-10 Oct 1997, pp: 522-533.