Security Estimation Model: Fault Perspective

Anshul Mishra[1], Dr. Devendra Agarwal[2], Dr. M. H. Khan[3]

[1]*Research Scholar, School of Computer Application, BBDU, Lucknow, India*
[2]*Director (Engg.) at BBDNIIT, BBDU, Lucknow, India*
[3] *Professor, Department of Computer Science & Engineering, I.E.T, Lucknow, India*

***Abstract*- The issue of security is essential for secure software. Measuring security of software system at design phase may help software developers to improve the security of software system. A security estimation model for object oriented design fault perspective has been developed in this paper. The proposed model correlates the Object Oriented Design constructs with Fault and Security. The security estimation process can be achieved by controlling the fault issues at design phase. This paper presents a multivariate linear regression for establishing the security estimation model in terms of Confidentiality, Integrity and Availability as attributes of security criteria to evaluate security of class diagram. Security estimation model is empirically validated and statistical significance of the study considers the high correlation for model acceptance.**

***Keywords*- Security Factors, Object Oriented Design Characteristics, Security Estimation, Fault factors**

I. INTRODUCTION

A successful software development process includes the details of each and every step of the process which are called as phases of development process. In general design phase, the designer/developer decides on the topology of the architecture of the project. In this whole process, there will be several challenges that the developer has to overcome to reach the final completion stage followed with the delivery of the project. Security can be measured in terms of confidentiality, integrity and availability to understand the impact in the project at early stage. The effectiveness of a security mechanism, however, depends on both users and technology "doing the right thing" [6]. Recent research on usability and security has focussed on user problems and needs [7, 20, 21]. The security of a software system can be measured by using a security attribute of software system. Some standards have proposed different general models of security, but there is no generally accepted set of security concepts and definitions [8]. Both known and unknown vulnerabilities can be exploited to compromise security attributes - confidentiality, integrity, availability, authenticity or non-reputability of information used by organization [14]. Software security provides many effective techniques to protect high secure software from illegal access and malicious occurrences at software development life cycle. Security essentials are expected methods ant techniques that provide the uncracked methodologies [22, 17]. Software security assessment is an integral part of security management of software projects. Security in software industries is considered to be a chance of occurrence of failure or probable failure. Security is a necessary factor for the quality software development and at the same time project may be unsuccessful by its impact.

Security refers to the high level protection of software projects from unauthorized access, modification and damage. Therefore, security specification is currently a major concern of software system and it is generally recommended to take care of security prior to software development life cycle [4, 13]. Security improvement strategies are exceedingly desirable for improving internal structure, design ease, flexibility, effectiveness simplicity, or other features of application software's [18, 10]. Security is parallel to the concept of safety, fault, and quality attributes. Number of security drawback and vulnerabilities exists due to the defects of security structural design and security mechanism. Hackers and attackers do not create security loopholes; rather they target the weaknesses in the software system and exploit them. "Fault" and "Security" are considered to be the most important attributes for secured software. This paper focuses on an empirical evaluation of security assessment in software system with three security attributes.

II. RELATED WORK

Study of security, experts says that fault, stability, complexity and reliability are an essential attributes which impinge on the depth of security [19]. A recent survey of parallel software products from different software developers found that the least secure software system carried a 6 times higher business risk than the most secure one, highlighting the fact that the high security quality of a software product can vary significantly depending on who considered and implemented it [1]. According to CERT, the number of security vulnerabilities in systems is increasing rapidly (from 2437 in 2001 to 4129 in 2002) [16]. Shalini et al., presented a security estimation framework using multivariate regression line technique [3]. Bharat B. Madan et. al. worked on model and assessment of security attributes of software systems [15]. Anders Bond & Nils Påhlsson presented a quantitative assessment framework for module security in distributed information systems [9]. ISO/IEC 27002:2005 standard provides eleven main security clauses encapsulating over one hundred control objectives. Control objectives contain high-level implementation guidelines for particular security

controls [12]. In Ekelhart et. al. proposed a security ontology for organize knowledge on threats, safeguards, and assets. This work constructs categorization for each of these groups and creates a method for quantitative risk analysis, using its own framework [5]. There is also a paper criticizing the quantitative security assessment approach stating that there is a require of validation and evaluation among these methods against observed data [11]. Marcelo Masera and Igor Nai Fovino present the abstract basis for and a simple example of quantitative assessment of the software security of multifaceted systems. It describes a set of security definitions, for then proposing a set of parameters, indicators theme and indexes values for computing and propagation security evaluations [2].

III.SECURITY ESTIMATION MODEL

Security quantification model have been considered as a basis to develop the metric based estimation model for considering complexity issues at design stage [10]. **Figure 1 shows** involve subsequent steps for security quantification. **Figure 2 shows** correlation establishment among, Security, Security Factors, Fault Factors, Design Metrics, and describes the estimation process of estimation model.

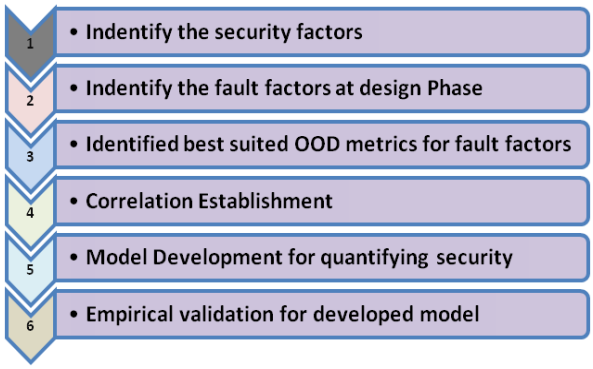

Fig1. Security Estimation Process

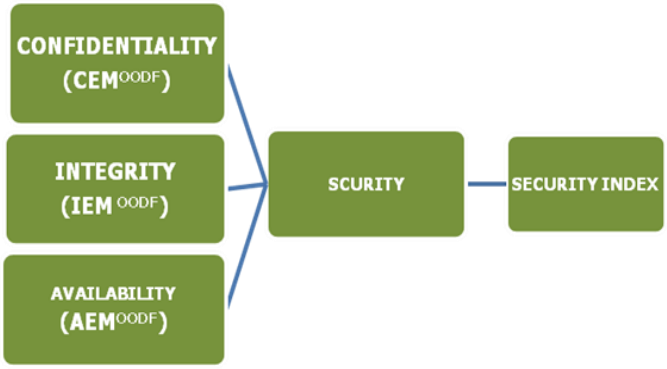

Fig 2. Correlation between security factors and security

The relative significance of individual identified security factors that have major impact on security estimation at design phase is weighted proportionally. The values of these design metrics can be identified by class diagram. In order to set up a metric based model for security estimation, a multiple regression technique has been used to get the coefficients of regression variables and regression intercept, shown in equation **5**. Identified security factors will take part in the role of independent variables while security will be taken as dependent variable. Estimation of security is very helpful to get security index of software design for high quality product. Multivariate regression equation is given in Equation (1) which is as follows

Where

- Y is dependent variable
- X1, X2, X3 ... Xn are independent variables.
- $\alpha_1, \alpha_2, \ldots \alpha_n$ are the regression coefficient of the respective independent variable.
- α_0 is the regression intercept.

$$Y=\alpha_0 + \alpha_1 X_1 + \alpha_2 X_2 + \alpha_3 X3 + \ldots\ldots \alpha_n X_n \quad (1)$$

It has been broadly reviewed and discussed in Section 1; Confidentiality, Integrity and Availability are the major factors affecting software security estimation considering with fault issue at design phase. Therefore, these identified major security factors were addressed well in advance while incorporating security at design phase. By applying the regression method, study already developed **Confidentiality Estimation Model [24]**, **Integrity Estimation Model [25] and Availability Estimation Mode**l [**26]** that is given in Equations (**2), (3) and (4)** respectively. The model of Confidentiality, Integrity and Availability forms the strong basis for development of Security Estimation Model.

$$\text{Confidentiality} = 0.3220 - 0.2140 * \text{DAM} + 0.2000 * \text{AVG\_CC} + 0.1920 * \text{CAM} \quad (2)$$

$$\text{Integrity} = 0.457 - 0.0420 * \text{DAM} + 0.00004 * \text{CE} + 0.144 * \text{MFA} \quad (3)$$

$$\text{Availability} = 0.860 + 0.00823 * \text{CE} + 0.0196 * \text{CAM} \quad (4)$$

It was observed that Object Oriented Design fault metrics affect security factor. Design metrics namely Inheritance **(MFA: Measure of Functional Abstraction), Encapsulation (DAM: Data Access Metrics), Cohesion (CAM: Cohesion among Methods), Complexity (Cyclomatic Complexity) and Coupling (CE: Coupling Efferent)** are used to address the key security factors namely Confidentiality, Integrity and Availability. These three identified factors are further used to measure security index of object oriented software at design stage in development life cycle. **Figure 2** gives an over-view of the main idea. In order to establish a model for software security estimation, a multiple regression method discussed in **Equation (1)** has been applied.

The data used for developing model has been taken from [26] that have been collected through large commercial object oriented systems. The data essential for accepted security values is being used from [3, 18].Which consists of six commercial software projects with around 10 to 22 number of classes. The values of design metrics namely, **Inheritance (MFA: Measure of Functional Abstraction), Encapsulation (DAM: Data Access Metrics), Cohesion (CAM: Cohesion among Methods), Complexity (AVG_CC: Cyclomatic Complexity) and Coupling (CE: Coupling Efferent) and the values of "Confidentiality, Integrity and Availability" have been used from** [17, 23, 27, 28]. Using SPSS, math work software correlation coefficients are calculated and model of Security Estimation is thus formulated as given in **Equation (5).**

Table I

Security Calculation Table

Project	Standard Confidentiality	Standard Integrity	Standard Availability	Standard Security
P_1	0.455	0.354	0.504	0.461
P_2	0.454	0.395	0.519	0.47
P_3	0.462	0.333	0.528	0.519
P_4	0.489	0.375	0.534	0.518
P_5	0.545	0.46	0.522	0.519

Security = - 0.478 + 0.763* Confidentiality - 0.396* Integrity + 1.46* Availability (5)

In **Table 2** the result of Model Summary is very important when performing multiple regressions. In this table, "R" is the multiple correlation coefficient that is used to know how strongly multiple independent variables are confidentiality, integrity, availability related to dependent variable as security. **"R square"** gives encouraging coefficient of determination.

The descriptive statistics of the output **Table** 3 gives the important proof of statistics that are mean, standard deviation and number of software projects selected for each of the dependent variable (security) and independent variable (confidentiality, integrity and availability).

Table II
Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.999[a]	.998	.998	.00676
Predictors: (Constant) Confidentiality, Integrity, Availability				

Table III
Descriptive Statistics

	Mean	Std. Deviation	N
Calculate Security	.8653	.15252	22
Confidentiality	.448126	.1319833	22
Integrity	.5145	.05478	22
Availability	.8264	.13411	22

IV. EMPIRICAL VALIDATION

Empirical validation process of work proves that how significant developed model, where metrics and model are able to estimates the security index of design stage. This validation is a vital stage of research to quantify the developed model for appropriate implementation and high level suitability. It is also the fine approach for claim the model recognition. To justify claiming for acceptance of developed model, an experimental validation of the developed security estimation model at design phase consider with fault has been carried out using tryout data [24, 25, 26]. In order to validate developed model, the value of metrics are available by using above data set for following projects in **Table 4**. Through experiment with statistical analysis, security index value of the projects has been computed using the developed model, followed by the estimates of security.

Table IV
Security Data Table

Project	Confidentiality	Integrity	Availability	Calculate Security	Standard Security
P_1	.4000	.527	.915	.955	.937
P_2	.3693	.495	.938	.977	.948
P_3	.3000	.423	.866	.848	.829
P_4	.4093	.551	.901	.931	.934
P_5	.3704	.520	.924	.948	.948
P_6	.3164	.457	.877	.863	.861
P_7	.3185	.457	.888	.880	.874
P_8	.3624	.514	.947	.977	.971
P_9	.6850	.415	.662	.847	.535
P_{10}	.4480	.569	.627	.554	.619
P_{11}	.6380	.457	.652	.780	.555
P_{12}	.3890	.553	.674	.584	.672
P_{13}	.5960	.585	.474	.437	.429
P_{14}	.7970	.586	.711	.936	.681
P_{15}	.5590	.538	.764	.851	.746
P_{16}	.5410	.435	.831	.976	.764
P_{17}	.3683	.512	.939	.971	.961
P_{18}	.4038	.535	.934	.981	.963
P_{19}	.4424	.594	.935	.989	.995
P_{20}	.4356	.574	.917	.966	.963
P_{21}	.3632	.537	.936	.954	.973
P_{22}	.3462	.487	.869	.832	.866

It is necessary to assessment the validity of proposed model for acceptance. A 2 sample paired t test analysis apply for verify the significance between standard security and

calculated integrity. 2t-test is handy hypothesis tests in statistics when compare means.

Table V.
2t- test between Standard Security and Calculate security

	Mean	N	Std. Deviation	Std. Error Mean
Standard Security	.81921	22	.168845	.035998
Calculate Security	.86532	22	.152515	.032516

Null hypothesis (H0): There is no significant difference between Standard Security and Calculate Security.

H0: $\mu 1 - \mu 2 = 0$

Alternate hypothesis (HA): There is significant difference between Standard Security and Calculate Security.

HA: $\mu 1 - \mu 2 \neq 0$

In the above hypothesis μ1 and μ2 are treated as sample means of population. Mean value and Standard Deviation value have been calculated for specified two samples and represented in table 5. The hypothesis is tested with zero level of significance and 95% confidence level. The p value is 0.055. Therefore alternate hypothesis directly discards and the null hypothesis is accepted. The developed equation used for Security estimation is accepted.

V. CRITICAL FINDINGS and TRY OUTS

This Study developed the Security Estimation Model. The Model has been validated using the same set of try out data. An empirical justification of the developed model is also performed using try-out data. Some of the critical findings are as given below:

- Confidentiality, Integrity and Availability has been recognized as a key factor to security, addressed in design phase with fault issues of object oriented software development to produce high secure software.
- Depth level measures of each of the security factors may be obtained
- Software design constructs are most appropriate and power full for controlling software security factors in de-sign phase
- Security estimation is an important key contribution for high level secure product.

VI. CONCLUSION

Software security key factors namely confidentiality, integrity and availability are identified and their significance on security estimation at design phase with fault issue has been tested and justified. The developed **Security Estimation Model** to assess security of object oriented software is extremely reliable and correlated with object oriented design artefacts. Security Estimation Model has been validated theoretically as well as empirically using statistical test. That empirically validation study on this research work proves that developed security estimation model is highly acceptable and helps the software industry in project ranking.

REFERENCES

[1]. The Security of Applications: Not All Are Created Equal. http://www.atstake.com. 2002.
[2]. M. Masera and I. N. Fovino, "Parameters for Quantitative Security Assessment of Complex Systems", ", Citeseers, 2010.
[3]. S. Ahmed and R. A. Khan, "Security Improvement of Object Oriented Design using Refactoring Rules", I.J. Modern Education and Computer Science, Vol 2, pp 24-31, Feb 2015.
[4]. D. Pandey, U. Suman and A. K. Ramani, "Security Requirement Engineering Issues In Risk Management", International Journal of Computer Applications, Foundation Of Computer Science, USA, ISBN:978-93-80747-89-4, Vol. 17, No. 5, Pp.11-14, 2011.
[5]. A. Ekelhart, S. Fenz, M. Klemen, and E. Weippl, "Security Ontologies: Improving Quantitative Risk Analysis," in HICSS '07 Proceedings of the 40th Annual Hawaii International Conference on System Sciences, 2007, pp. 1–7
[6]. Viega, J., McGraw and G. "Building Secure Software", 2002.
[7]. Adams, J. Risk. 1995. UCL Press.
[8]. M. Masera and I. N. Fovino, "Parameters for Quantitative Security Assessment of Complex Systems", IJCE, Apr 2010.
[9]. A. Bond and N. Påhlsson, "A Quantitative Evaluation Framework for Component Security in Distributed Information Systems", Thesis, Institute of Technology, link opening university, 2004.
[10]. M. Al-Kuwaiti and S. Hussein, "A Comparative Analysis of Network Dependability, Fault Tolerance, Reliability, Security and Survivability", IEEE, Vol. 11, No. 2, 2009.
[11]. V. Verendel, "Quantified security is a weak hypothesis: a critical survey of results and assumptions," in Proceedings of the 2009 workshop on New security paradigms workshop, ser. NSPW '09. New York, NY, USA: ACM, 2009, [Online].Available: http://doi.acm.org/10.1145/1719030.1719036
[12]. ISO, ISO/IEC Std. ISO 27002:2005, Information Technology - Security Techniques - Code of Practice for Information Security Management. ISO, 2005.
[13]. A. Agrawal, R. A. Khan and S. Chandra, "Software Security Process: Development Life Cycle Perspective", CSI communications, pp. 39-42, August 2008.
[14]. J. Breier and L. Hudec, "New Approach in Information System Security Evaluation", IEEE, 2012.
[15]. B. B. Madan, K. and K. S. Trivedi, "Modelling and Quantification of Security Attributes of Software Systems", Proceedings of the International Conference on Dependable Systems and Networks (DSN'02), IEEE, pp.505-514, 2002.
[16]. CERT. http://www.cert.org.
[17]. S. A. Khan and R. A. Khan, "Integrity Quantification Model for Object Oriented Design", ACM SIGSOFT Software Engineering Notes, Vol 37, No.2, Mar 2012.
[18]. S A Khan and R A Khan, "Security Quantification Modell", International Journal of Software Engineering, ISSN: 2090-1801, Vol. 6, No. 2, pp: 75-89, 2013.
[19]. A. Mishra, Dr. D. Agarwal and Dr. M. H. Khan, "A Critical Review of Fault Tolerance: Security Perspective", International Journal of Computer Science and Information Technologies, Vol. 8 (1), 132-135, 2017.

[20]. Weirich and Sasse "A. Pretty Good Persuasion: A first step towards effective password security in the real world", New Security Paradigms Workshop 2001.
[21]. Whitten, and A. Tygar "Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0", Proceedings of the 8th USENIX Security Symposium, August 1999.
[22]. PPT Network Security, Available at: http://boardreader.com/tp/ppt%20network%20security.htmllast visit Oct 15 (2014).
[23]. M. Jureczko and L. Madeyski, "Towards identifying software project clusters with regard to defect prediction", IEEE, 2010.
[24]. A. Mishra, D. Agarwal and M. H. Khan, "Confidentiality Estimation Model: Fault Perspective" International Journal of Advanced Research in Computer Science (IJARCS), Volume.8 Issue. 4, June 2017.
[25]. A. Mishra, D. Agarwal and M. H. Khan, "Integrity Estimation Model: Fault Perspective", International Journal on Recent and Innovation Trends in Computing and Communication, Vol 5, Issue 5, pp 1246-1249, May 2017
[26]. A. Mishra, D. Agarwal and M. H. Khan, "Availability Estimation Model: Fault Perspective", International Journal of Innovative Research in Science, Engineering and Technology, Vol. 6, Issue 6, June 2017.
[27]. N. Parveen, M. R. Beg and M. H. Khan, "Model to Quantify Availability at Requirement Phase of Secure Software", American Journal of Software Engineering and Applications, Vol. 6, 2015.
[28]. S. A. Khan and R. A. Khan, "Confidentiality Estimation Model for Object Oriented Design", International Journal of Information and Education and Technology, Vol. 37, No. 2, March 2010.