International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 5 Issue: 5 1246 – 1249

Integrity Estimation Model: Fault Perspective

Anshul Mishra[1],
[1]School of Computer Application
BBDU, Lucknow, India

Devendra Agarwal[2],
[2]Director (Engg.) at BBDNIIT
BBDU, Lucknow, India

M. H. Khan[3]
[3] Department of Computer Science & Engineering,
I.E.T. Lucknow

***Abstract*—**An integrity estimation model for object oriented design fault perspective has been proposed in this paper. Integrity has been recognized as a major factor to software security, an importance is being drawn to measure integrity early in development life cycle. No such model has been available in the literature that estimates security of object oriented design by taking fault parameters into consideration. A suit of design metrics useful in measuring integrity of software has been recognized. It becomes more significant in the case of object oriented design fault perspective. In this study a metrics based **Integrity Estimation Model (IEM^{OODF})** for object oriented design has been developed and justifying the correlation with the help of experimental tryouts. Finally the developed model **"Integrity Estimation Model"** is empirically validated and contextual importance of the study shows the high correlation for proposed model acceptance.

Keywords-*Integrity, Fault attributes, Object Oriented Design Characteristics, Security attributes*

*****

I. INTRODUCTION

A systematic approach to security in the software development lifecycle is an important to protect the complete project [11]. An accurate evaluation of software security depends on security estimation, which in turn depends on the factor that can affect security. The development of security oriented software still leftovers a matter of proper guidelines, finest practices and undocumented expert knowledge. In the highly competitive software industry, customer pressure causes Companies to go faster the speed to deliver software products. Though Schedules are frequently tightly limited; developers are required to weight the importance of software security against the chance of missing deadlines. For meet the objectives to need the implementing security in earlier stage of development and importance of design phase with respect to security.

To minimize vulnerabilities and achieve objective level security, estimation of security is necessary. Unfortunately, quantify the estimation of security in design phase is largely missing. Design properties are tangible concepts that can be directly assessed by examining the internal, external structure, functionality of design component, methods and classes. As far as security estimation in general and particularly at design phases of the software development life cycle is concerned, it is very hard problem to be solved and has got attention lately [1]. The software security should show capability to secure itself and the system from the attacker's exploitation and misuse of software security loophole [12]. The studies reveal the truth that secure software knows how to divaricate significantly according to user requirement, updated design and its implementation [3, 7].

Of course due to the possibility of threats that are unknown, no system will ever be 100 % secure but the security estimation which provide an early indicator of security is helpful and needed [1, 4]. A series of tragedies and chaos caused by the insecure software revealed that an unwanted design fault is one of the responsible reasons for successful attacks [8]. Software security considerations when considered as part of software development life cycle process, results in a more secured product. It is observed that the security of software can be monitored systematically at design phase of software development process.

A fault is critical issue a software flaw that causes failure [6]. Each security attribute such as confidentiality, integrity and availability requires the use of one or more trusted mechanisms that are implemented in software components. If a module is not secure then the whole concept of module based software development fails. This paper is structured in such an approach that it initially describes the integrity: a key factor to security for estimation process and describes integrity estimation model with their correlation establishment and also highlights information for statistical analysis impact of the developed model that are significant for further study.

 ISSN: 2321-8169
1246 – 1249

II. OVERVIEW OF THE PROPOSED MODEL

A. *Integrity Security Factor*

Integrity is an assurance that information is accurate. Integrity is strongly related to security and constantly plays a key role to deliver high secure software. The security of software components is one of the important factors determining the quality of components. Integrity minimization of software is the only way to maintain the originality and protect from security attack. At design time, class hierarchies are involved to share the information according to functional potential of concerned methods, functions and attributes [8]. In object oriented design perspective, functional attributes affected by the design property including cohesion, coupling, inheritance and polymorphism. Loss of integrity, functional level will be affected. It is visibly evident from previous article [10] that Confidentiality, Integrity, Availability, Durability and Authorization are common accepted factors at design phase.

B. *Integrity Estimation Model*

We have developed an integrity Estimation model that demonstrates the Estimation method of software security. The proposed model is shown in figure 1. The model establishes an appropriate impact relationship between fault factors and integrity constructs and the associated design metrics. The accurate values of these metrics can be identified with the help of class diagram. The quantifiable evaluation is very supportive to get integrity index of software design for low cost security Estimation. Security attributes involves maintaining the consistency, accuracy, and dependability of data over its complete software development life cycle. To calculate security it is better to analyze requirement constructs by means of security attributes that can be considered as availability, confidentiality, and integrity [2, 10 and 13].

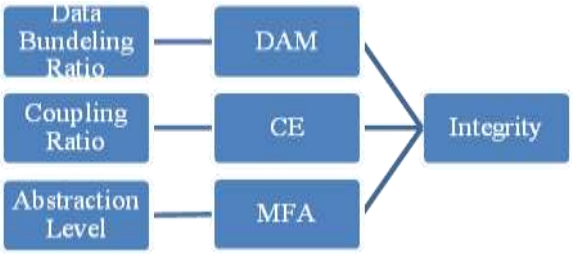

Figure 1. Correlations among Fault Factor and Security Factors

III. MODEL DEVELOPMENT FOR INTEGRITY ESTIMATION

It is understandable from in-depth literature survey that integrity is not a new word; rather it has been in conversation between the software professionals at different environment. The generic security models have been measured as a basis to develop the **"Integrity Estimation Model (IEM^{OODF})"** for object oriented design [9].

A. *Proposed framework implemented the following steps:*

- Indentify the integrity as a security factors
- Indentify the fault factors at design Phase
- Identified best suited OOD metrics for fault factors
- Correlation Establishment
- Model Development for quantifying Integrity
- Empirical validation for developed model

Use Correlation among fault factors and security factors has been established and shown in figure 1. In order to set up a model for integrity, multiple linear regression method has been used. Multivariate linear model is given below in "(1)" which is as follows.

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

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.

The data used for establishing integrity model is taken from [5] that have been collected through large commercial object oriented systems. The relationship between security factor integrity and object oriented fault design properties has been established as depicted in figure 1. The result is tabulated in table 1 in order to elucidate the correlation analysis for quantify integrity and developed the "(2)". The data essential for accepted integrity values is being used from [8]. Using "SPSS" math work software values of all design metrics, intercept, and coefficient of the respective design metrics are calculated. On the basis of this approach, the multiple linear regression confidentiality models have been developed that is given in "(2)". Identified independent variables, namely Coupling Efferent (CE), Measure of functional Abstraction (MFA), Direct Class Metric (DAM), the values of dependent variable "Y" can be found output by using the "Integrity Estimation Model of Object Oriented Design".

 ISSN: 2321-8169
1246 – 1249

Table I. Integrity Computed Table

Project	Standard Integrity	DAM	CE	MFA
P_1	**0.454**	**0.00000**	**9**	**0.00000**
P_2	**0.500**	**1.00000**	**4**	**0.70000**
P_3	**0.600**	**0.00000**	**1**	**1.00000**
P_4	**0.454**	**0.20000**	**4**	**0.00000**
P_5	**0.545**	**1.00000**	**5**	**0.80000**

Integrity = 0.457 - 0.0420* DAM + 0.00004* CE+ 0.144 * MFA (2)

III. STATISTICAL SIGNIFICANCE OF INDEPENDENT VARIABLES

The descriptive statistics of the output table 2 gives the valuable record of statistics that are mean, standard deviation and number of software projects selected for each of the dependent variable and independent variable.

The Integrity Estimation Model summary (table 3) of the output is most useful when performing multiple regressions. In this table "R" is the multiple correlation coefficient that used to know how strongly multiple independent variable are related to dependent variable. "R square" gives supportive coefficient of determination.

Table II. Descriptive Analysis of Project

Descriptive Statistics

	Mean	Std. Deviation	N
Calculate Integrity	**0.52242**	**0.062918**	**10**
DAM	**0.51779**	**0.374723**	**10**
CE	**6.2**	**3.084009**	**10**
MFA	**0.595928**	**0.4001272**	**10**

Table III. Model Summary for Confidentiality Model

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	**.999[a]**	**0.999**	**0.998**	**0.002769**

a. Predictors: (Constant), DAM, CE, MFA

IV. EMPIRICAL VALIDATION

One can easily find many research works in the literature of security in measures without proper empirical validation. Performing empirical validation with the metrics is fundamental in order to show their practical research. The empirical validation is a key stage of planned research to validate **Integrity Estimation Model** for high level acceptability. Empirical validation is the standard approach and practice to say the model acceptance. In order to validate proposed **Integrity Estimation Model** the value of metrics is available data sets for given ten projects taken from [5].

Table IV. Integrity Data Table

Project	DAM	CE	MFA	Standard Integrity	Calculate Integrity
P_1	1.000	2.000	.0000	.411	.415
P_2	.600	9.000	.9524	.572	.569
P_3	.000	1.000	.0000	.459	.457
P_4	.500	8.000	.8125	.534	.553
P_5	.250	9.000	.9583	.554	.585
P_6	.000	8.000	.8922	.584	.586
P_7	.471	7.000	.6957	.526	.538
P_8	.933	3.000	.1163	.412	.435
P_9	1.000	9.000	.9070	.552	.546
P_{10}	.424	6.000	.6250	.535	.540

It is compulsory to test the validity of proposed model for acceptance. A 2 sample t test applies for check the significance between standard integrity and calculated integrity. 2t-test is handy hypothesis tests in statistics when compare means.

Table V. 2t- test between Standard Integrity and Calculate Integrity

Paired Samples Statistics

		Mean	N	Std. Deviation	Std. Error Mean
Pair 1	**Calculate Integrity**	0.52242	10	0.062918	0.019896
	Standard Integrity	0.51398	10	0.063587	0.020108

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

H0: μ1-μ2 = 0

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

HA: μ1-μ2 ≠ 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. Correlation comes out to be 0.981, that shows the standard integrity and calculated integrity is highly correlated. The hypothesis is tested with zero level of significance and 95% confidence level. The p value is 0.06. Therefore alternate hypothesis directly discards and the null hypothesis is accepted. The developed equation used for integrity Estimation is accepted.

V. CONCLUSION

Software security is a typical task because no proper approach was existed for measuring this property. Here in this paper we developed an **Integrity Estimation Model (IEM^{OODF})** for quantify the integrity with fault attributes at early stage of software development life cycle. A multiple linear regression approach is used for quantify the integration. The statistical importance and validation of model has been done in this paper for acceptance.

REFERENCES

[1] I. A. Mir and S.M.K Quadri, "Analysis and Evaluating Security of Component Based Software Development: A Security Metrics Framework", I. J. Computer Network and Information Security, pp 21-31, 2012.

[2] P.Nikhat, S. Kumar and M. H. Khan, "Model to Quantify Integrity at Requirement Phase", Indian Journal of Science and Technology, Vol. 9, Aug 2016.

[3] Flechais, Sasse and H. SMV. "Bringing Security Home: A Process for developing secure and usable systems", ACM, pp 18-21, Aug 2003.

[4] Oman, P. Risley and A. Roberts, "Attack and Defend Tools for Remotely Accessible Control and Protection Equipment in Electric Power Systems", 55th Annual Conference for Protective Relay Enginers, Texas A&M University, April 2002.

[5] M. Jureczko and L. Madeyski, "Towards identifying software project clusters with regard to defect prediction", IEEE, 2010.

[6] A. P. Mittal, S. Singh and K. S. Kahlon, "Empirical Model for Fault Prediction using Object-oriented metrics in Mozilla Firefox", International Journal of Computer Technology & Research, pp. 151- 161, 2003.

[7] I. Flechais, M. Sasse and S M V Hailes, "Bringing Security Home: A Process for developing secure and usable systems", NSPW'03, ACM, pp:18-21, Aug 2003.

[8] S. A. Khan and R. A. Khan, "Integrity Estimation Model for Object Oriented Design", ACM SIGSOFT Software Engineering Notes, Vol 37, No.2, Mar 2012.

[9] Wang C & Wulf, "A Framweork for Security Measurement", Proc.of National Information Systems Security Conference, pp 522-533, Oct 1997.

[10] S. Chandra, R. A Khan and A. Agrawal, "Software Security Factors in DesignPhase"International Conference on Information Systems, Technology and Management, Springer Berlin Heidelberg, pp 339-344, March 2012.

[11] D. P. Gilliam, J.D. Powell and M. Bishop. "Application of Light weight Formal Methods to Software Security", 14th IEEE International Workshops on Enabling Technologies, 13-15 June 2005.

[12] G. Mc Graw, "Software Security", IEEE Security and Privacy, Vol 2,
pp. 80-83, 2004.

[13] Sterne and Daniel, "On the Buzzword "Security Policy", IEEE. 1991.