

Availability Estimation Model: Fault Perspective

Anshul Mishra¹, Dr. Devendra Agarwal², Dr. M. H. Khan³

Research Scholar, School of Computer Application, BBDU, Lucknow, India¹

Director (Engg.), BBDNIIT, BBDU, Lucknow, India²

Professor, Department of Computer Science & Engineering, IET, Sitapur Road, Lucknow, India³

ABSTRACT: Availability is one of the most important security criteria for early stage of software development life cycle. Early assessment of availability mainly at design phase helps to make use of its measurements more competently to enhance the security of software product. We elaborate a practical and formal availability model that enables security assessment using fault design factors. The design stage estimation of availability is more practicable for software developer team and protection inexpensively. This paper proposes an “**Availability Estimation Model**” (AEM^{ODF}). The proposed model for computing availability point out the possible influence of design properties and also observe the impact of software security computation with fault attributes at design stage.

KEYWORDS: Availability, Fault Attributes, Object Oriented Characteristics, Security Attributes

I. INTRODUCTION

Software development process is a critical task that involves a number of stages such as inception, initial design, detailed design and development, implementation testing and operation [3, 6, 17]. A most important challenge in software development process is to advance fault detection at early stages of software development life-cycle. The concept of security assessment is closely related to confidentiality, integrity and availability of assets [2]. Experts and Researchers are continuously concerning about complete secure software, whose primary ambitions is to implement a security model or framework, that functions properly under malicious use and that does not contain loopholes.

Software security means an unintended user are prevented to operate the system under any environment [11]. Any effort to breach security is called an attack; it can be in a large number of forms [6]. Taking security assessment early phase of a system development life cycle should have an impact on reducing many software faults. This study is focuses on a set of object oriented fault design metrics that can be used to quantify the availability. It has been reported that cost and attempt spent on software security is very high, approximately between 65% to 70% of total software development life cycle and support efforts [4]. The three basic security requirements confidentiality, integrity and availability namely CIA has been recognized. CIA being the cornerstone of security and it completely depends on authentication and authorization [8, 12].

The allow software designers to access the software design in early phase of software development, making changes that will reduce the fault and improve the security of the software. Sometimes tolerant minor fault always need efforts to reduce faults that might source them is in the centre of security. The survey highlighted the fact that security quality of software product can vary considerably depending on its design and implementation [1]. A present software expert gives consideration towards only normal behaviour with assumption that all faults can be removed during development. To develop secure software is the contributing process of different steps and reflections of each phase is the matter of study to measure the accurate impacts of security [7]. This research aims to propose a model to evaluate availability by using multiple regression method at early stage.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

II. AVAILABILITY: DESIGN PHASE

The majority of the mechanisms available for security quantification of objects oriented software usually are used in later phases of system development process. The theory of availability refers to the up time maintenance of all resources in software. The security factor is the capability of the software product to be secured. This means that all the software module and resources one have are functional all the time. For any type information or data when it is shared, it increases the availability of related information but some reason any vague or customized gathering information gets tampered [9, 5]. Availability models stay data and resources available for authorized use, especially during emergencies or disasters.

The availability of a system can be pictured as a set of software artifacts (classes) interacting with one another at design level. Inter-class links are assumed to have greater influence on availability than intra-class links. Software developers to avoid availability violation at design level it is mandatory to establish a systematic relation between the classes and entities. Availability is also correlated to highest strength of protection because the information can be traced by intruders to violate security quantification [10]. The internal and external security quantification provides an overview about the availability of the software product, and is dependent on its fault issues at design phase. It is visibly evident from previous article [13, 16] that Confidentiality, Integrity, Availability, Durability and Authorization are common accepted factors at design phase.

III. AVAILABILITY ESTIMATION MODEL

Availability has not been introduced as a single attribute of security. Security requirements mainly take in confidentiality, integrity, availability, authentication, authorization, validation, reliability and access control. Security quantification is an essential method for secured software and it must be taken at design level. Availability is strongly related to security and continuously acts a major role to deliver high class secured and reliable software within time. It is one of the most important concepts in design. It always supports for enhanced security of the software at initial stage of software development life cycle. That is to say at design phase that have positive impact on the overall security quantification cost and effort. We have developed an “**Availability Estimation Model**” that demonstrates the quantification method of software security at early stage of software development process. The model establishes an accurate impact relationship between fault factors and availability constructs and the associated design metrics.

IV. CORRELATION ESTABLISHMENT

Correlation establishment is most important step. Correlation is a statistical procedure that can explain whether and how strongly pairs of variables are related. In this step the recognized software security factors are to be correlated with the fault attributes at design level.

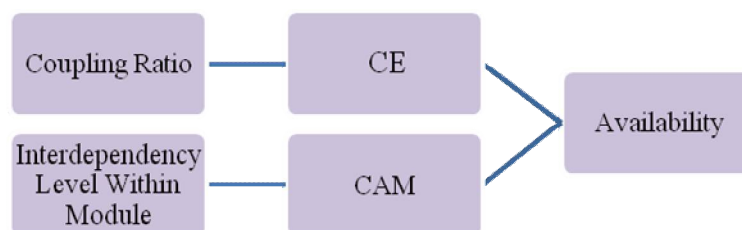


Fig 1. Correlation between faults attributes and security attributes

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

V. MODEL DEVELOPMENT FOR AVAILABILITY MEASUREMENT

The generic security model has been considered as a foundation to develop the Availability Estimation Model for object oriented software design [10, 12]. Estimation of class diagram's is prerequisite for the accurate availability estimation. Proposed model implemented the following steps are shown in figure 2.

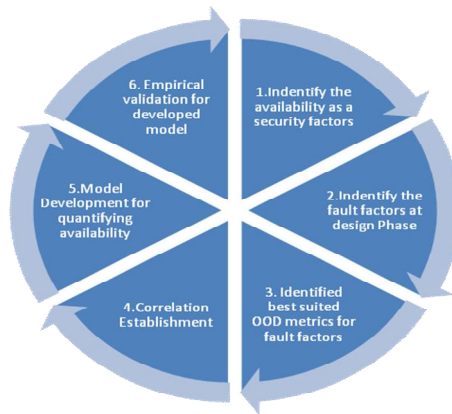


Fig 2. Research Framework for Availability Estimation Process

Correlation has been established and shown in figure 1. In order to set up a model for Availability, multiple linear regression method has been used. Multivariate linear model is given below in Eq (1) which is as follows.

$$Y = \alpha_0 + \alpha_1 X_1 + \alpha_2 X_2 + \alpha_3 X_3 + \dots + \alpha_n X_n \quad (1)$$

Where

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

The data used for developing model has been taken from [14] that have been collected through large commercial object oriented systems. The values of Coupling Metrics (CE), Cohesion Metrics (CAM) have been used for quantify the availability. The data essential for accepted availability values is being used from [12]. Cohesion refers to the internal availability within the parts of the design. A class is cohesive when its parts are extremely correlated. It should be complicated to divide a cohesive class. Cohesion can be used to recognize the badly designed classes. Coupling indicates the association or interdependency among modules. Coupling is a measure of interconnecting between modules in a software organization

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

Table I. Availability Computed Table

Project	Standard Availability	CE	CAM
P ₁	0.950	9	0.44444
P ₂	0.912	4	0.50000
P ₃	0.874	1	1.00000
P ₄	0.941	4	0.40625
P ₅	0.862	5	0.38889
P ₆	0.890	1	0.55556
P ₇	0.838	4	0.28421
P ₈	0.947	5	0.30000
P ₉	0.891	1	0.51852

Using SPSS, correlation coefficients are computed and model of availability measurement is thus formulated as given below in **equation (2)**. In the **equations (2)** the terms coupling efferent (CE) and cohesion among method (CAM) are independent and availability as dependent variables. Coupling and Cohesion is the best approach for predict the fault issues at design phase of software development [18]. Efferent Coupling is a metric solution in software development. It measures the number of parameters a class knows about. Cohesion is the kind of natural development of data hiding for example; set having all members in proof with a package having default visibility.

$$\text{Availability} = 0.860 + 0.00823 * \text{CE} + 0.0196 * \text{CAM} \quad (2)$$

VI. STATISTICAL ANALYSIS of AVAILABILITY ESTIMATION

Pearson’s coefficient of correlation technique was used for estimating the degree of correlation among variables. It can show a predictive relationship that can be exploited in practice. The Pearson correlation is +1 in the case of a perfect direct linear relationship (correlation), -1 in the case of a perfect decreasing linear relationship (anticorrelation), and some assessment in the open interval (-1 and 1) in all additional cases, signifying the measure of linear dependence between the variables. The value of correlation ‘r’ lies between ±1, positive value of ‘r’ in **table 2** is a sign of positive correlation between the two variables.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

Table II. Correlation between Independent Variables

		Calculated Availability	CE	CAM
Pearson Correlation	Calculated Availability	1.000	.973	.889
	CE	.973	1.000	.768
	CAM	.889	.768	1.000

Descriptive statistics are distinguished from inferential statistics, in that descriptive statistics aim to summarize a sample, rather than use the data to learn about the population that the sample of data is thought to represent. Descriptive statistics of the variables and the impact results were then calculated. We were also interested to know whether there is any relationship between a particular OO design metric and the impact for the given fault issues on that system. The descriptive statistics of the output **table 3** 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.

Table III. Descriptive Statistics of variable

Descriptive Statistics			
	Mean	Std. Deviation	N
Calculated Availability	.91809	.007798	10
CE	6.30000	.823273	10
CAM	.30349	.088747	10

The Availability Estimation Model summary (**Table 4**) 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 IV. Model Summary for Availability Estimation

Model Summary				
Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
	.998 ^a	.996	.995	.000562
a. Predictors: (Constant) CE, CAM				

VII. EMPIRICAL VALIDATION

Empirical validation is a vital phase of proposed research. Empirical validation is the standard approach to justify the model approval. Taking view of this truth, practical validation of the availability model has been performed using sample tryouts. In order to validate developed availability measurement model the data has been taken from [14, 15].

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

Table V. Availability Measurement Table

Project	CE	CAM	Calculated Availability	Standard Availability
P ₁	5.000	.300	.910	.840
P ₂	7.000	.382	.925	.921
P ₃	6.000	.199	.913	.906
P ₄	5.000	.162	.904	.865
P ₅	7.000	.375	.925	.923
P ₆	6.000	.197	.913	.907
P ₇	7.000	.400	.925	.918
P ₈	6.000	.300	.915	.888
P ₉	7.000	.333	.924	.930
P ₁₀	7.000	.388	.925	.920

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

Table VI. 2 t- test between Standard Availability and Calculated Availability

	Mean value	N	Std. Deviation	Std. Error Mean	T- Value	P- value	Correlation
Standard Availability	.91809	10	.007798	.002466	2.243	0.052	0.834
Calculated Availability	.90177	10	.029109	.009205			

Null hypothesis (H₀): There is no significant difference between Standard Availability and Calculated Availability.

H₀: $\mu_1 - \mu_2 = 0$

Alternate hypothesis (H_A): There is significant difference between Standard Availability and Calculated Availability.

H_A: $\mu_1 - \mu_2 \neq 0$

In the above hypothesis μ_1 and μ_2 are treated as sample means of population. Mean value and Standard Deviation value have been calculated for specified two samples and represented in table 6. The hypothesis is tested with zero level of significance and 95% confidence level. The p value is 0.052. Therefore alternate hypothesis directly discards and the null hypothesis is accepted. The developed equation used for availability estimation is accepted.

VIII. CONCLUSION

This study shows the importance of availability in general and as a key factor to software security for producing high class reliable and secure software within time and budget. Availability is clearly highly appropriate and significant in the perspective of software security. Availability model is developed with the help of multiple linear regression method on object oriented design properties. Statistical analysis shows that proposed model is statistically very much significance and acceptable. Availability estimation model has been validated contextually as well as empirically using experimental tryout. The practical validation on the “**Availability Estimation Model (AEM^{OODF})**” concludes that proposed model is highly acceptable and secure.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 6, June 2017

REFERENCES

- [1] Available at www.cert.org
- [2] M. Al-Kuwaiti and S. Hussein, "A Comparative Analysis of Network Dependability, Fault Tolerance, Reliability, Security and Survivability", IEEE, Vol. 11, No. 2, 2009.
- [3] S. Jain and M. Ingle, "A Review of Security Metrics in Software Development Process", International Journal of Computer Science and Information Technologies, Vol. 2 (6), 2011.
- [4] K. K. Aggarwal, Y. Singh and J. K. Chhabra, "An Integrated Measure of Software Maintainability", Proceedings Annual Reliability and Security Symposium, pp 235-241 2002.
- [5] R. Subramanyan and M.S. Krisnan, "Empirical Analysis of CK Metrics for Object-Oriented Design Complexity : Implications for Software Defects," IEEE Trans. Software Eng., Vol. 29, No. 4, PP 297-310, Apr. 2003".
- [6] L. Bass, P. Clements and R. Kazman, "Software Architecture in Practice", Second edition, Sei- Series in Software Engineering.
- [7] Dr. R. A. Khan and S. A. Khan, "A Roadmap for Security", International Journal of Computer Science & Emerging Technologies (IJCSET), Vol. 1, June 2010.
- [8] 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, August 2003.
- [9] N. Parveen, Md. Rizwan and Beg, et al. "Software Security Issues: Requirement Perspectives" International Journal of Scientific & Engineering Research, Vol. 5, pp 11-15, Issue-7, July 2014.
- [10] S. A. Khan and R. A. Khan, "Securing Object Oriented Design: A Complexity Perspective", International Journal of Computer Applications (0975-8887), Vol 8, No.13, October 2010.
- [11] Alshammari, Bandar and Fidge, Colin J. and Corney, Diane (2009) "Security metrics for object-oriented class designs" In: QSIC 2009 Proceedings of: Ninth International Conference on Quality Software, pp 24-25, Aug 2009.
- [12] 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.
- [13] A. Mishra, D. Agrwal & 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.
- [14] M. Jureczko and L. Madeyski, "Towards identifying software project clusters with regard to defect prediction", IEEE, 2010.
- [15] S.P. Kadam and S. Joshi, "Secure by Design Approach to Improve Security of Object Oriented Software", IEEE, 2015.
- [16] A. Mishra, D. Agrwal & M. H. Khan, "Confidentiality Estimation Model: Fault Perspective" International Journal of Advanced Research in Computer Science (IJARCS), Volume.8 Issue. 4, June 2017.
- [17] Mohammad Zunnun Khan, M Akheela Khana & M. H. K. "Requirement Modifiability Quantification Model of Object Oriented Software" , Special Edition on Applied Mathematics in Information and Communication Technology (AMICT), June 2017.
- [18] S. Praveen & R. A. Khan "fault Prone Model for object oriented software: Design Phase", Vol 2, ICCEA, 2009.