International Journal of Research in Advent Technology, Vol.6, No.10, October 2018
E-ISSN: 2321-9637
Available online at www.ijrat.org

Risk Impact Analysis of Software Vulnerability at Early Stage

Dr. Brijesh Kumar Bhardwaj,
Assistant Professor, Department of MCA
Dr. R. M. L. Avadh University, Faizabad

Abstract- Productive improvement of software project in any association is accomplished by guaranteeing that the conveyed framework meets the desires of the client. Risk thusly ought to be settled by minimize vulnerability which stays away from the event of risk or even by arranging the mitigation when its event is unavoidable. In this paper centres on dissecting the effect of risks across over vulnerability which are arranged as far as here and now and long projects. This risk analysis empowers the designer to grasp and along these lines deal with the risks all the more successfully.

Index Terms- Risk analysis, risk factors, software engineering, vulnerability, software development life cycle.

1. INTRODUCTION

A productive software advancement process incorporates the points of significance of every last advance of the procedure which are called as phases of improvement process [5]. When all is said in done, these phases can be characterized as requirement phase being the underlying one, where the client desires are changed over to business objectives that are thus characterized as far as project objectives [8, 2]. Design phase being the following, the designer/engineer chooses the topology of the engineering of the project. In the ensuing phase, this topology is coded by making utilization of the software details. Subsequently, in the Testing phase, designer will have the capacity to check or approve the code alongside the information against the business requirements [1]. In this entire procedure, there will be a few difficulties that the engineer needs to overcome to achieve the last culmination organize took after with the conveyance of the project. There are various factors that produce the software project risk. Vulnerability is basically negative factors that create hurdles during the execution of software project or risks are even those events that can threaten the software success. Risks can be handled or managed but it is necessary to minimize the vulnerability. Estimating the vulnerability are shown in [10], Vulnerability impact analysis is categorised in different level and generate the risk at three level with rule. The risk analysis is shown in figure 1 and dependent on the vulnerabilities index.

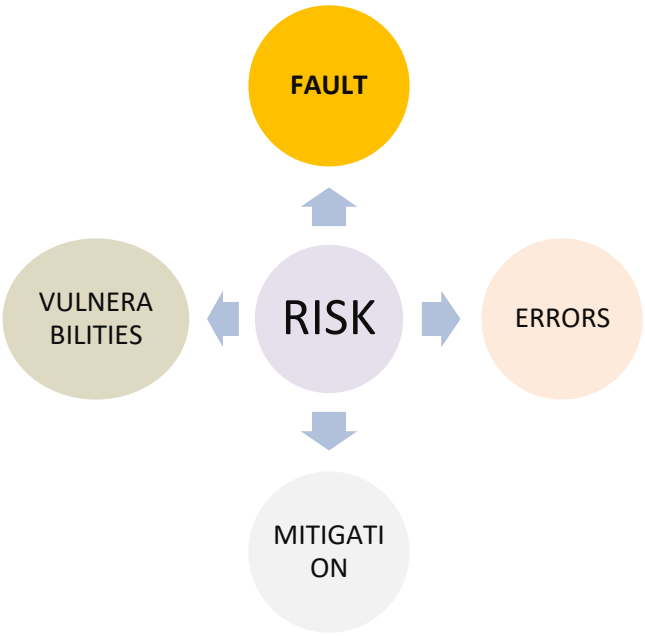

Fig 1 Software risk contributions

2. RISK EVALUATION WITH VULNERABILITY

Vulnerability is an essential factor for the quality software development and at the same time project may be unsuccessful by its impact [9]. The impact of vulnerability on software project may result in produce the risk. Risk defines uncertainty and loss of software project. Vulnerability index calculated by Relationship between Vulnerability and Security: A Design Stage and also given the impacts on software. Table 1 shows the rule with risk impacts. Risk assessment and risk mitigation reduce the reason responsible for the unsecure software project. Risk assessment determines the existence of risk whereas risk mitigation minimizes vulnerability.

Table 1 Vulnerabilities Impact View

		Rule with risk: vulnerabilities index		
Project	**Calculated Index**	**High Risk (0.8-0.99)**	**Moderate Risk (0.6-0.79)**	**Low Risk (0.5-0.59)**
P_1	**.704**			
P_2	**.701**		+	
P_3	**.584**			+
P_4	**.777**		+	
P_5	**.580**			+
P_6	**.777**		+	
P_7	**.965**	+		
P_8	**.684**		+	
P_9	**.543**			+
P_{10}	**.572**			+
P_{11}	**.566**			+
P_{12}	**.600**		+	
P_{13}	**.557**			+
P_{14}	**.926**	+		
P_{15}	**.577**			+
P_{16}	**.917**	+		
P_{17}	**.938**	+		

2.1 Risk Associated

Risk factors are the indeterminate conditions and impacts that influence the cost, span, furthermore, nature of the project contrarily [3, 6]. On the off projects from finishing effectively and accomplishing its objectives and expected results . Many experts have turned out to be occupied with distinguishing software projects risk factors [7]. This is because of the way chance that these factors are disregarded or not alleviated well, they will show genuine dangers that obstruct the software

that software projects risk factors change additional time, and in this way normal software risk identification considers must be led occasionally so as to recognize increasingly risk factors.

Fig 2 Risk Phases

3. CONCLUSION

In this paper, a detailed vulnerability was examined deeply. It was found that the relationship between software projects vulnerability factor is a cause-effect risk. To clarify this, a vulnerability index table and a cause-effect risk category. This would reduce the maximum number of vulnerability. By this the constraint of vulnerability can be reduced and risk can improve.

REFERENCES

[1] Kitchenham, B., Pfleeger, S. L. "Software Quality: the Elusive Target,"1996. IEEE Software, vol. 13, no. , pp. 12-21.

[2] Al-Qutaish, R. E., "Quality Models in Software Engineering Literature: An Analytical and Comparative Study," Journal of American Science, 2010. Vol. 6, no. 3, pp. 166-175.

[3] Devpriya Soni, Ritu Shrivastava , M. Kumar ,"A Framework for Validation of Object-Oriented Design Metrics", (IJCSIS) International Journal of Computer Science and Information Security,Vol. 6, No.3, 2009.

[4] Istehad Chowdhury, Mohammad Zulkernine "Can Complexity, Coupling, and Cohesion Metrics be used as Early Indicators of Vulnerabilities? " SAC '10 Proceedings of the 2010 ACM Symposium on Applied Computing 2010.

[5] Chen, H., Wanger, D. (2002): MOPS: an Infrastructure for Examining Security Properties of Software. *Technical Report: CSD-02-1197, 2002,* ACM, pp. 235-236.

[6] Tevis, J.-E. J., Hamilton, Jr., J. A. (2004): Methods for the prevention, detection and removal of software security vulnerabilities. In ACMSE'04- Southeast Conference, 2-3 Apr, 2004, ACM Press, pp. 197- 202, doi:acm.org/10.1145/986537.986583.

[7] McGraw, G. (2003): From the ground up: The DIMACS software security workshop. *IEEE Security & Privacy*, *vol. 1, Mar-Apr 2003*, pp. 59 – 66, doi: 10.1109/MSECP.2003.1193213.

[8] Taylor, B., Azadegan, S.(2006): Threading Secure Coding Principles and Risk Analysis into the Undergraduate Computer Science and Information System Curriculum. In conf. on Information Security Curriculum Developemnt, 2006, ACM, pp. 24-29, doi:acm.org/10.1145/1231047.1231053.

[9] D. DaCosta, C. Dahn, S. Mancoridis, and V. Prevelakis, "Characterizing the 'security vulnerability likelihood' of softawre functions," Proc. IEEE Conf. Software Maintenance, (ICSM'03), IEEE, 22-26 Sep. 2003, pp. 266-274, doi:.ieeecomputersociety.org/10.1109/ICSM. 2003.1235429.

[10] Dr. Brijesh Kumar Bhardwaj, "Relationship between Vulnerability and Security: A Design Stage" (IJREAM) Vol-04, Issue-03, June 2018, ISSN: 24549150.