ISSN:2320-0790

COMPUSOFT
An International Journal of Advanced Computer Technology

Recent Advances in Risk Analysis and Management (RAM)

Arpita Banerjee[1], C. Banerjee[2], Dr.Ajeet Singh Poonia[3]

[1]Assistant Professor, St. Xavier's College, Jaipur, Rajasthan, India
[2]Assistant Professor, Amity Institute of Info. Technology, Amity University, Rajasthan, India
[3]Associate Professor, CSE, Govt. College of Engineering & Technology, Bikaner, Rajasthan, India

Abstract: In today's age, organizations consider software development process as an investment activity which is dependent on the comprehensive and precise working of each phase in Software Development Lifecycle. Flaws from each phase could remain undetected starting from requirement phase till maintenance phase. The flaw or defects if left unattended in the respective phase will be carried forward to next phase aggregating the issues. These undetected flaws should be identified and removed as early as possible so as to reduce additional overheads. From the data available, it is concluded that risk analysis is a major factor which is ignored during all the phases of software development process resulting in the emergence of undetected defects and flaws. Because of the failure of many projects, the importance of risk analysis during software development process is now being well recognized. A series of reversed as well as assorted researches are proceeding towards analyzing the risk 'right from the beginning' during the software development process. Through researchers have contributed significantly in the field, still more needs to be achieved. This paper presents a review of the current research being done in Risk Analysis and Management (RAM), based on the recently published work. The study is carried out with respect to analysis and management of risk in various phase of SDLC. Such a thorough review enables one to identify mature areas of research, as well as areas that need further investigation. Finally, after critical analysis of the current research findings, the future research directions are highlighted with their significance.

Keywords: Risk, Risk Analysis and Management (RAM), Recent Work in RAM, Future Directions in RAM.

I. INTRODUCTION

With the advent of technology comes threat to the system. System are first designed and then developed and the process of creating software system is called System Development Life Cycle. Due to the ever emerging issues of security of these software systems, it was thought that to secure the software system, security should be incorporated right from the beginning i.e., the requirements engineering phase [1]. To assess security of a software system we need to first assess the threats ad risk involved and the best way to achieve is by using some metrics which is dedicated specially for security purpose [2][3].

Security certification is the basic tool needed for decision making while making a software system process with a standard level of risk. The notion of "Risk" is shaped by the security needs in a problem domain, thus, contextually subjective. Analysis of risk is considered as foundation stone on which the entire software can be built. Successful analysis of risk, not more for business-level decision-support is a way of gathering the essential data to make a good judgment.

A high-level approach to repeated risk analysis should be fundamentally incorporated throughout the software development life cycle. In earlier days, the analysis of risk was not taken seriously, which caused many big software problems. These problems' nature and quality both continue to grow exponentially with the growth in software complexity and its versatility [4].The initial phases is the foremost opportunity for the product team as well as the risk management team to consider how analysis and management of risk will be inculcated into a development process, identify key security threats and vulnerabilities and minimize the software risk. The SDLC team's overall perspective should be to do analysis of risk factors to minimize threats and vulnerabilities [5].For creating an understanding among the members of software development team, security awareness programs needs to be implemented among them of security [6] [7]. Special focus needs to be put among the programmers regarding the awareness of various software awareness tools which are collaborative and distributed in nature so that the team members could value the importance of risk analysis and management during the software development process [8] [9]. Further, for a comprehensive and synchronized approach to risk analysis and management, a balanced approach covering the technical aspect as well as the management aspect needs to be adopted [10].

From the research findings, it is quite evident that the errors discovered in the later phases of software development process were actually the outcome of the initial phases. Rectifying these errors in the later stages not only takes time but the overall cost of re-work becomes too high [4].Therefore risk factor should be calculated to determine how much attention and time is to be given to implement the requirements for conducting risk analysis [6].In this paper, we present a review of the recent research directions in analyzing of risk in various phase of SDLC. The rest of the paper is organized as follows: In Section II, current research in RAM is briefly reported, whereas in Section III, we present the future research directions. Conclusion is reported in Section IV.

II. LITERATURE SURVEY OF RAM RESEARCH

Although significant contribution has been made from the research and academic community regarding the varied aspects of Risk Analysis and Management (RAM)'.But still there is scope for further research for refinement of the concept which drives the research community towards a more comprehensive and practical Risk Analysis and Management (RAM) tool. A selection from the trend setting research contributions are briefly described one by one for analysis on the advances, as follows:

Robin Gandhi and Seok-Won Lee presented a partial research on ontology guided process of building "formal metrics" for understanding risk. For this the related evidence are collected from the Certification and Accreditation (C&A) process. The outcome of this research was a methodological approach for development of metrics and understanding with the use of the structured depiction of guiding security requirements in problem domain ontology [11].

Authors named Karel de Bakker, et al. presented a meta-analysis of realistic evidences. The main outcome of this analysis was to validate the contribution of risk management to success of any IT project. Their paper also studies the validity of the assumptions used as basis of risk management [12].

Dapeng Liu, et al. suggested that software projects were still suffering from many problems due to various classes of software risks as they were often not well implemented in real-world software projects. This paper discussed Software Process Simulation Modeling (SPSM), which is emerging as a promising approach to address a variety of issues in software engineering area, including risk management [13].

Edzreena Edza Odzaly, et al. presented a survey of experienced project manager's perception on software risk management. Eighteen experienced project managers were surveyed to find the uses of risk management mechanism. Their research paper concluded that high cost and comparative low values was the main cause of less use of risk management tools [14].

Guo Chao Peng, et al. contributed to the research world by developing, establishing and managing potential risks associated with the post-adoption of Enterprise Resource Planning (ERP) systems [15].

The Katie Grantham, et al. presented identified risks during the conceptual phase of a product design in their research paper. They further developed software named RAD which was based on analysis of risk in early design. Further their paper also stated some mechanism to identify risks in the early phase of product design by relating recorded historical failure information to product functions. Further the authors also used a multi-level evaluation framework to determine how well the application meets the needs of various organizations. As part of the evaluation, a questionnaire was developed and administered to a sample industrial and academic user group [16].

Morakot Choet kiertikul, et al. work aims to reduce the process overhead of risk assessment by automatically collecting data from the project management repository to adequately and appropriately determine the approximate level of risk in off shoring projects. Their work presented an extension of their previous model of quantitative Capability Maturity Model Integration (CMMI) assessment. They also applied the best practices from the Capability Maturity Model Integration (CMMI) as a guideline for quantitative risk analysis in off shoring and using risk taxonomy from the Software Engineering Institute (SEI) Taxonomy-Based Risk Identification [17].

Saima Amber, et al. in their paper suggested a framework in which risk management is executed within Requirement Engineering (RE) process. Three models of risk management are considered which are used to identify risky functional requirements. These models are compared on the basis of risk identification methodologies. A new model is derived which is based on UML oriented approach for modeling and reasoning about risk during the requirements analysis process [18].

Mohd. Sadiq, et al. developed a tool called 'esrcTool' which estimates the risk factor in any software also determines the cost of the software. They used function points for the estimation of risk and cost. Many models are present to estimate the risk of the software like SoftRisk Model, SRAM, SRAEM, etc. But 'esrcTool'is more practical and uses SRAEM i.e. Software Risk Assessment and Estimation Model, because in this model on one side FP is used as an input variable, and on the other hand side, International Software Benchmarking Standards Group Release Report (ISBSG) is used in order to determine the cost of the software [19].

Lazaros Sarigiannidis, et al. in their research paper investigated a wide range of relevant literature. After going through various studies a conceptual framework for managing risk in software development projects is proposed. They also introduced new conceptual factors, bring out their interrelation, and suggest new prospects and managerial implications for both practitioners and academics in their research paper [20].

Norman Fenton and Martin Neil in their research paper addresses the basic limitation of both data driven statistical approaches and risk register for effective risk management and assessment with the use of Baysian Network. The authors have used Baysian Network to identify, understand and quantify how risk can be mitigated and controlled [21].

Antoine Cailliau and Axel van Lamsweerde presented a probabilistic framework for specification of goal and problem assessment. The paper focused on the uses of computed information to sort obstacles for countermeasure selection for a more complete and robust goal model. They used framework to evaluate a non-trivial carpooling support system [22].

K Venkatesh Sharma and P V Kumar in their research paper proposed a requirement engineering model based on the Tropos goal model. They have used a modified Tropos goal model in the proposed goal risk model which consists of three layers [23].

Yogini Bazaz, et al. in their research paper has discussed the comparison between different software risk assessment models corresponding to certain risk elements in their research paper. These risk factors are analyzed to draw some conclusion and further this conclusion is used to state the weaknesses and strengths of risk assessment models [24].

Surbhi Anand and Vinay Chopra through their research work stated the finding of all possible risk factors and their interdependencies with each other. On the basis of this report they have proposed a decision support system to analyze software risks. The results of the tool will help the software developers to take important future decisions [25].

S. K. Pandey and K. Mustafa did a critical review of existing risk assessment methodologies particularly COBRA, CORAS, CRAMM, OCTAVE, SOMAP, and NIST Guide, along with its strengths and weaknesses. The research paper aims at helping the senior IT personnel to provide their recommendations for using a risk assessment methodology based on the specific requirements of an organization [26].

Poonam Kaushalin the research paper illustrated a systematic approach for risk analysis in addition a methodology for software effort estimation of component based software development was also studied [27].

Mumtaz Ahmad Khan, et al. presented a systematic review of Software risk assessment and estimation models. The main emphasis is given to the risk assessment methods based on software metrics like Software Risk Assessment and Estimation Model (SRAEM) and Software Risk Assessment and Evaluation Process (SRAEP) using model based approach because these methods are the latest methods in the field of software risk assessment and estimation [28].

The CERT® Program at Carnegie Mellon University's Software Engineering Institute (SEI) has chartered the Software Security Measurement and Analysis (SSMA) Project to advance the state-of-the-practice in software security measurement and analysis. The SSMA Project is exploring how to use risk analysis to direct an organization's software security measurement and analysis efforts. The overarching goal is to develop a risk based approach for measuring and monitoring the security characteristics of interactively complex software-reliant systems across the lifecycle and supply chain. To accomplish this goal, the project team has developed the SEI Integrated Measurement and Analysis Framework (IMAF) and refined the SEI Mission Risk Diagnostic (MRD). This report is an update to the technical note, Integrated Measurement and Analysis Framework for Software Security (CMU/SEI-2010- TN-025), published in September 2010. This report presents the foundational concepts of a risk based approach for software security measurement and analysis and provides an overview of the IMAF and the MRD [29].

Haneen Hijazi, et al. studied different software development methodologies which exists, choosing the methodology that best fits a software project depends on several factors. They investigated the state of risk and risk management in the most popular software development process models (i.e. waterfall, v-model, incremental development, spiral, and agile development) in their research paper [30].

Supannika Koolmanoj wong suggested a detailed description on top 10 risk from various perspectives such as comparison of risk patterns between development-based projects and COTS-based projects, high score teams and low score teams, and a comparison between risk exposure and risk occurrence [31].

Haneen Hijazi, along with fellow researchers presents a comprehensive theoretical study of the major risk factors of SDLC phases. An exhaustive list of 100 risk factors was produced. This list reflects the most frequently occurring risk factors that are common to most software development projects [32].

The research paper authored by Sanjeev Puri states a risk assessment framework for a precise, unambiguous and efficient risk analysis with qualitative risk analysis methodologies to reduce risk levels and optimize quality instructions [33].

III. FUTURE RESEARCH DIRECTIONS IN RISK ANALYSIS AND MANAGEMENT (RAM)

Risk Analysis and Management (RAM) is a very active research area, with a wide variety of methods. At present, there is no consensus on a single and best approach to security requirements engineering. However, many organizations intuitively feel that attention to this area will pay off in supporting their business goals [22]. The researchers have proposed some risk analysis and management framework and some researchers have suggested a wide-range list of software risk factors that covers major threats through the software development process. This list can serve as a guiding tool to analyze the risk factors and help them in designing strategies to mitigate or avoid them.

Academicians and researchers have also done comparative study of various risk assessment techniques in order to identify their weakness and strengths in managing risk in software development environment. But this method was not able to handle risks in Globally Distributed Software Development (GDSD) Projects due to multi-locations, multi-cultures, multi groups, multi-technologies. Further, there was no specific model which was able to manage the risks in web distributed environment alone [24]. More detailed research is needed while anticipating the risk factor and their assessment. Some researchers have done research work on the effort estimation of Cost Based Software Development (CBSD) which pay off in supporting their business goals [22]. Moreover, future work may include process definition and improvement, education and training, good project management, use of proper tools and techniques, measurement, sufficient resources, and sheer hard work [27]. Software projects are still suffering from many problems and the reason behind that currently Software Process Simulation Modeling (SPSM) is mainly applied in risk analysis and risk management planning activities. The capabilities of SPSM in other risk activities have not been well explored. Most of the existing SPSM approaches and models have not been applied into actual risk management practices. Software engineering researchers and practitioners should cooperate more closely in the future [27]. One of the future research directions would be proposing priority based parameters in Tropos model (risk based model) which was generally used by the researchers as requirement engineering model. Another future work may involve developing framework / model for the assessment of the cost effectiveness of countermeasures and their integration in the goal model. This model could also provide guidance on when to use a particular technique or representation [22].

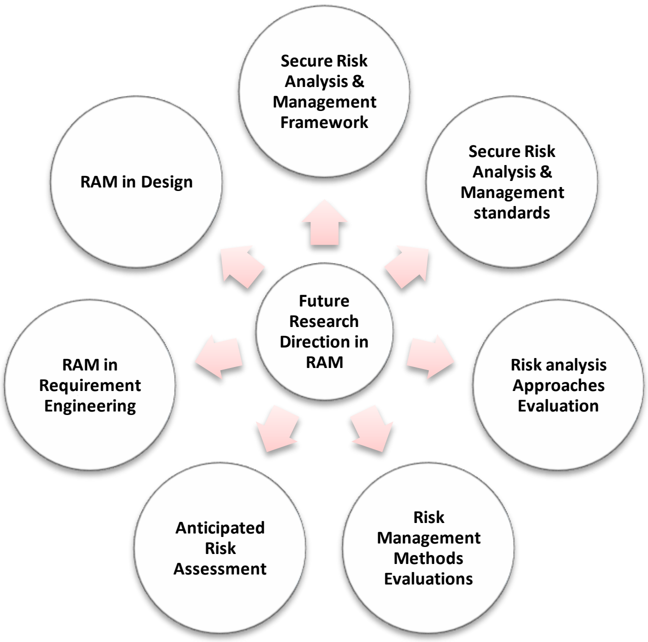
Secure Risk Analysis & Management Framework
Secure Risk Analysis & Management standards
RAM in Design
Future Research Direction in RAM
RAM in Requirement Engineering
Risk analysis Approaches Evaluation
Anticipated Risk Assessment
Risk Management Methods Evaluations

Figure 1. Future Research Directions in Risk Analysis & Management (RAM)

IV. CONCLUSION

Most of the current risk assessment methodologies can only be used to estimate risk in the later stages of the software life cycle, typically from design models or code. As a result these methodologies can identify risks but have limited capability in preventing these risks from occurring. Software requirement's risk addresses the possibility of suffering a loss of any functional or non-functional requirement of the software system. It is more feasible to make changes to the software system under development in the early stages of the software development cycle.

Risk analysis is, at best, a good general-purpose index by which security design's effectiveness can be properly judge. As an estimate, approximately 50 percent of security issues are the result of design flaws, therefore performing a risk analysis at the design level becomes important part of a good software security program. If risk-analysis methods at the design level for any application are implemented effectively the software often yields valuable, business- relevant results. It is a continuous process and is applied to many different levels, identifying system-level vulnerabilities, assigning probability and impact, and determining reasonable mitigation strategies.

By evaluating the result, business stakeholders can determine how to manage a particular risk and analyze its cost which may lower the Total Cost of Ownership (TOC) for the development of software. Keeping in view, we presented a number of research areas in which further work is required, based on the published work of the year 2009, 10, 11, 12, 13 and 14. However, it is evident that directions being reported are conclusive, effective and efficient ways to incorporate risk analyzing and management right from the beginning in the software development life cycle i.e, *the requirements engineering phase*.

V. REFERENCES

[1] Banerjee, C., &Pandey, S. K. (2009). Software Security Rules, SDLC Perspective. arXiv preprint arXiv:0911.0494.

[2] Banerjee, C., Banerjee, A., &Murarka, P. D. Measuring Software Security using MACOQR (Misuse and Abuse Case Oriented Quality Requirements) Metrics: Attacker's Perspective. (2014) IJETTCS, 3(2), 245-250.

[3] Banerjee, C., Banerjee, A., & D Murarka, P. (2014). Measuring Software Security using MACOQR (Misuse and Abuse Case Oriented Quality Requirement) Metrics: Defensive Perspective. International Journal of Computer Applications, 93(18), 47-54.

[4] S. K. Pandey et. al. (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 04, 2010, 1079-1085, Recent Advances in SRE Research [1]

[5] Denis Verdon, Gary McGraw Risk Analysis in Software Design [2]

[6] Banerjee, C., & Pandey, S. K. (2010). Research on software security awareness: problems and prospects. ACM SIGSOFT Software Engineering Notes, 35(5), 1-5.

[7] Banerjee, A. B., & Murarka, P. D. (2013). An Improvised Software Security Awareness Model. International Journal of Information, Communication and Computing Technology, 1(2), 43-48.

[8] Banerjee C., Banerjee Arpita, Pandey S. K. (2013): Software Security Awareness: Comparison of Artifacts Based Awareness Tools and Techniques. SGVU Journal of Engineering & Technology, 1(1), 33-38

[9] Banerjee Arpita, Banerjee C. (2014). Cyber Security Awareness Through Education: Problems and Prospects. IMPETUS an Interdisciplinary Research Journal.2(1).

[10] Banerjee C., Murarka P D, Banerjee Arpita (2013). IT Security Practices in an Organisation: Balancing Technology and Management Perspective. IMPETUS an Interdisciplinary Research Journal.2(1).1-6.

[11] Gandhi, R. A., & Lee, S. W. (2007, October). Visual analytics for requirements-driven risk assessment.In Requirements Engineering Visualization, 2007. REV 2007. Second International Workshop on (pp. 6-6). IEEE.

[12] De Bakker, K., Boonstra, A., &Wortmann, H. (2010). Does risk management contribute to IT project success? A meta-analysis of empirical evidence.International Journal of Project Management, 28(5), 493-503.

[13] Liu, D., Wang, Q., & Xiao, J. (2009, October). The role of software process simulation modeling in software risk management: A systematic review. InEmpirical Software Engineering and Measurement, 2009.ESEM 2009. 3rd International Symposium on (pp. 302-311). IEEE.

[14] Odzaly, E. E., Greer, D., & Sage, P. (2009, October). Software risk management barriers: An empirical study. In Empirical Software Engineering and Measurement, 2009.ESEM 2009. 3rd International Symposium on (pp. 418-421). IEEE.

[15] Peng, G. C., &Nunes, M. B. (2009). Surfacing ERP exploitation risks through a risk ontology. Industrial Management & Data Systems, 109(7), 926-942.

[16] Grantham, K., Elrod, C., Flaschbart, B., &Kehr, W. (2012). Identifying Risk at the Conceptual Product Design Phase: A Web-Based Software Solution and Its Evaluation.

[17] Choetkiertikul, M., &Sunetnanta, T. (2010, August). A risk assessment model for offshoring using CMMI quantitative approach.In Software Engineering Advances (ICSEA), 2010 Fifth International Conference on (pp. 331-336).IEEE.

[18] Amber, S., Shawoo, N., & Begum, S. (2012). Determination of Risk During Requirement Engineering Process. Journal of Emerging Trends in Computing and Information Sciences, ISSN, 2079-8407.

[19] Sadiq, M., Rahman, A., Ahmad, S., Asim, M., & Ahmad, J. (2010, May). esrcTool: a tool to estimate the software risk and cost. In Computer Research and Development, 2010 Second International Conference on (pp. 886-890). IEEE.

[20] Sarigiannidis, L., &Chatzoglou, P. D. (2011). Software development project risk management: A new conceptual framework. Journal of Software Engineering and Applications, 4(05), 293.

[21] Fenton, N., & Neil, M. (2011). The use of Bayes and causal modelling in decision making, uncertainty and risk. CEPIS Upgrade 12 (5), 10-21.

[22] Cailliau, A., & Van Lamsweerde, A. (2012, September). A probabilistic framework for goal-oriented risk analysis. In Requirements Engineering Conference (RE), 2012 20th IEEE International (pp. 201-210). IEEE.

[23] Sharma, K. V., & Kumar, P. V. (2012, December). An efficient risk analysis in requirement engineering. In Engineering (NUiCONE), 2012 Nirma University International Conference on (pp. 1-5). IEEE.

[24] Bazaz, Y., Gupta, S., PrakashRishi, O., & Sharma, L. (2012, March). Comparative study of risk assessment models corresponding to risk elements. In Advances in Engineering, Science and Management (ICAESM), 2012 International Conference on (pp. 61-66).IEEE.

[25] Anand, S., & Chopra, V. (2012). Decision Support System for Software Risk Analysis During Software Development. International Journal for Science and Emerging Technologies with Latest Trends, 2(1), 29-35.

[26] PANDEY, S. K. (2012). A Comparative Study of Risk Assessment Methodologies for Information Systems. Bulletin of Electrical Engineering and Informatics, 1(2), 111-122.

[27] Dapeng Liu (2012). Software effort estimation and risk analysis –A Survey, International Journal of Engineering and Innovative Technology (IJEIT) 1(1).

[28] Khan, M. A., Khan, S., &Sadiq, M. (2012). Systematic review of software risk assessment and estimation models. International Journal of Engineering and Advanced Technology, 1, 298.

[29] Alberts, C. J., Allen, J. H., & Stoddard, R. W. (2012). Risk-based measurement and analysis: application to software security.

[30] Hijazi, H., Khdour, T., &Alarabeyyat, A. (2012). A Review of Risk Management in Different Software Development Methodologies. International Journal of Computer Applications, 45.

[31] SupannikaKoolmanojwong. (2014). Top-10 Risks in Real-Client Software Engineering Class Projects, IEEE

[32] Hijazi, H., Alqrainy, S., Muaidi, H., &Khdour, T. (2014). Risk Factors in Software Development Phases. European Scientific Journal, 10(3).

[33] Puri, S. (2010). A Risk Assessment Framework to Reduce Risk Level and Optimize Software Quality. SAMRIDDHI.