**INTEGRATING SECURITY IN APPLICATION DEVELOPMENT LIFECYCLE USING PROTECTION POKER TECHNIQUE**

**Abstract :** This research work presents a methodology for effectively addressing critical software security risks (CSSRs) through effective risk management techniques, particularly focusing on the application of protection poker. Beginning with a meticulous identification and assessment of the specified risks, the methodology utilizes protection poker to collaboratively evaluate the likelihood and impact of each risk, fostering informed decision-making and prioritization. Quantifying risks using a numeric scale enables a comprehensive understanding of their severity, facilitating focused resource allocation and mitigation efforts. Through a comprehensive understanding of potential vulnerabilities and proactive mitigation efforts facilitated by protection poker, organizations can prioritize resources effectively and ensure the successful outcome of projects and initiatives in today's dynamic threat landscape.

1. **Introduction**. In the current era of 5G revolution where mobile phones are easily accessible to all masses, cyber security has gained paramount importance. Security attacks in the cyber domain have increased significantly and hence calls for increased protective measures.

Security in Application Development has gained substantial importance over the recent times owing to hacking and other attacks on computer systems. As a result, security has to be made intrinsic component of all the stages of mobile application development. If sensitive data is hacked or exposed, it may cause un-repairable loss to software companies repute with partners, customers and investors, therefore system managers and users are paying more and more attention to this important aspect of security. In order to have complete secure applications/ softwares, security aspects needs to be considered into every step of Software Development Life Cycle (SDLC) or Application Development Life Cycle (ADLC). To blend security into the software engineering model, it should be considered from the beginning of the SDLC/ ADLC [1]. Most organizations normally see security as a post-development process. Security concerns have a profound impact on the overall quality of software, as organizations grappling with insecure software are compelled to address and rectify unreliable applications, while also impeding the progress of other concurrent projects. Emerging cyber vulnerabilities, both internal and external to the organization, continue to surface persistently, posing formidable financial risks and jeopardizing the integrity of critical company data. The repercussions of such security lapses extend beyond monetary losses, encompassing substantial credibility deficits that can be detrimental to the reputation and standing of the organization [2].

The main purpose of this research is to evaluate the Protection Poker technique's efficacy because of risk assessment tool for CSSRs revealed in Security Assurance Model (SAM) [3]. This type of assessment can help to improve the method used to indicate the areas where under development program needs special consideration/ deliberation. Software companies can now estimate their security parameters and their abilities to create more secure applications.

2. **LITERATURE REVIEW**.

**Definition and Importance of Integrating Security.**  "Integrating security" mentions the systematic addition of security measures and concerns at each phase of the application development life cycle (ADLC). This scheme is essential for making applications that are resilient to cyber attacks, protecting users and software producers from any losses and breaches. Integration of security to ADLC is not only a technical requirement but it can also be termed as a key business decision as primarily it serves to reduce weaknesses and post-deployment security fix costs. As a normal practice, historically security was often neglected/ ignored throughout the creation of applications. This reactive scheme has inherent probability of more susceptibilities and exploitation threats. With security-centric methodologies like DevSecOps gaining reputation for integrating security into every stage of the software development process, secure practices in ADLC have evolved over a short span of time to become more proactive.

1. **Strategies and Practices**. Coding while keeping security risks in mind helps to end vulnerabilities at the start, this is known as secure coding techniques. Coding standards must be strictly followed in order to prevent security related issues, Moreover, code analysis procedures must also be put to use in order to address security problems [4].
2. **Security Requirements Engineering**. Security requirements must be properly logged, deliberated upon and documented in the initial development stage. It ensures that program is built while keeping security consideration alive during all phases of development [5].

Risk assessment in the context of Application Development Life Cycle (ADLC) is surrounded with identification/ assessment of possible risks associated with security threats that can arise during the development of Application[6].

a. **Security at Various Phases of ADLC**

**Initial Planning and Design**: Threat modeling and secure design are the two main cardinals of this phase. Security threats are resolved through threat modeling by devising counter measures for them, while secure design principles calls for such application design which ensures enhanced security..

**Implementation**: Secure coding strategies/ code reviews are the two main activities performed in this phase to ensure that code is functionally correct and resistant to known vulnerabilities.

**Testing**: Known secure testing methods, such as dynamic and static analysis, are used to reveal and identify susceptibilities that may have gone unnoticed in early phases. It also includes code reviews and penetration testing.

**Deployment:** In order to keep application’s security operational. Secure deployment practices and vulnerability management are key mechanisms that ensures the operation-ability of application with security mechanism intact.

**Maintenance**: Constant monitoring and incident response are vital for maintaining the application's security post-deployment..

**Potential Research Gap**. A potential gap in quantitatively assessing the severity of 46 x CSSRs identified in SAM during ADLC is discovered by the literature review. While numerous security practices and risk assessment procedures exist, a collaborative method to estimate the impact of the identified risks on application security seems to be missing. 46 x security risk identified in Security Assurance Model are given below :-

| **Ser** | **Issue** |
| --- | --- |
| a. | Injection |
| b. | Broken authentication and session management |
| c. | Cross-site scripting (XSS) |
| d. | Insecure direct object references |
| e. | Security mis-configuration |
| f. | Sensitive data exposure |
| g. | Missed function-level access control |
| h. | Cross-site request forgery (CSRF) |
| i. | Using components with known vulnerabilities |
| j. | Un-validated redirects and forwards |
| k. | Insufficient logging and monitoring |
| l. | Broken access control |
| m. | Improper input user data |
| n. | Buffer overflow |
| o. | Improper error handling |
| p. | Race conditions |
| q. | Failure to restrict URL access |
| r. | Insufficient transport layer protection |
| s. | Server-side request forgery (SSRF) |
| t. | Insecure cryptographic storage |
| u. | Session fixation |
| v. | Poor password policy and management |
| w. | Improper authorization |
| x. | Click-jacking |
| y. | Excessive data exposure |
| z. | Broken anti-automation defenses |
| aa. | Improper file and resource protection |
| bb. | Not enough security configuration |
| cc. | Broken business logic |
| dd. | Improper certificate validation |
| ee. | Time and state-related attacks |
| ff. | Fail to restrict upload of dangerous file types |
| gg. | Insufficient session expiration |
| hh. | Cryptographic issues |
| ii. | Insecure communications |
| jj. | Inadequate encryption strength |
| kk. | Authentication bypass |
| ll. | Improper use of a security feature |
| mm. | XML external entities (XXE) |
| nn. | Insufficient security controls in a third-party service |
| oo. | Missing security headers |
| pp. | Broken cryptography usage |
| qq. | Improper asset management |
| rr. | Out-of-band channel exploitation |
| ss. | Security relevant mis-configuration |
| tt. | Elevation of privilege |

**Proposed Solution**. The proposed solution necessitates engaging with software companies to assess the effectiveness of the protection poker technique in reducing identified critical software security risks. Feedback from these companies is gathered through a questionnaire. In Protection Poker, participants gauge the impact (probability and consequence) of security risks using playing cards. This game-like method encourages conversation and promotes a collective comprehension of security risks among the development team. By integrating Protection Poker into the ADLC, development teams could jointly evaluate the seriousness of CSSRs highlighted by a SAM, allowing them to prioritize security efforts and resources on the most significant risks. The integration of Protection Poker into the ADLC involves the following steps:

1. **Identify CSSRs**: A SAM would be utilized during the security assurance phase to pinpoint potential CSSRs in the application being developed.
2. **Plan a Protection Poker Session**: Organize a session for Protection Poker involving developers, security experts, and other relevant stakeholders.
3. **Assess Risk Impact**: Each CSR would be deliberated upon, and participants would utilize playing cards to determine the likelihood and consequence of the risk. The aggregated card values would be used to produce a risk score.
4. **Prioritize Security Focus**: With the risk scores allocated through Protection Poker, development teams can prioritize their security endeavors towards the most critical CSSRs.

**Research Questionnaire:**

1. **Identifying Risks:** How does your software house identify the 46 security risks outlined in the security assurance model during the application development lifecycle?
2. **Risk Management:** What strategies or methods do you use to manage and mitigate the identified risks?
3. **Risk Assessment:** How do you integrate risk assessment methods like the protection poker technique into your application development process?
4. **Risk Prioritization:** Based on the risk estimation provided, how do you prioritize risks and allocate resources for their mitigation?
5. **Protection Techniques:** Can you share your experiences with using protection poker or similar techniques to assess and address these risks during application development?
6. **Challenges and Solutions:** What specific challenges have you faced in managing the aforementioned risks, and what strategies have you found most effective in addressing them?

**Proposed Methodology** :

The methodology employed in this research involves the assessment and analysis of 46 security risks that have been identified through the security assurance model. To estimate the levels of risk associated with these vulnerabilities, the protection poker technique is utilized. It involves an organized and efficient strategy to assess the likelihood and consequences of each risk, leading to an in-depth understanding of potential threats. It generally involves quantifying a risk level associated with every identified security concern. Following stages are involved.

**Risk Identification and Assessment.**

1. This risk assessment process begins by taking 46 x risks mentioned in the security assurance model into evaluation process. These security risks addresses a wide range of probable vulnerabilities that could eventually threaten the security and integrity of the application. By going through the characteristics of each risk, deeper understanding of their nature and potential consequences is developed which helps in improving application's overall security stance.
2. Furthermore, it is also important to examine the underlying root causes of each risk as this knowledge will help in creating effective mitigation strategies.
3. Protection poker technique is adapted to evaluate each risk which helps in harnessing the expertise of discussing members who are well-versed in security and development domains. This method revolves around a structured and candid discussion where team members assess the probability and impact of each risk through this collective insight and eventually this discussion ends when reached to some consensus.
4. By assessing probability of occurrence of each risk and its relevant probable impact, team members can prioritize their mitigation efforts and address the critical security issues in an efficient manner.
   * 1. **Risk Estimation.**
   1. In order to effectively evaluate the risks linked to a specific circumstance, it is imperative to measure each risk using a numerical scale that varies from 1 to 10. This scale terms 1 as the minimum risk level whereas 10 as the maximum one. Linking of a numerical figure to each risk helps in the evaluation of both risk’s probability of occurrence and its probable impact.
   2. It is usually more beneficial to deconstruct risks into more significant/ practical terms that actually demonstrates their seriousness, therefore for this research risks have been demonstrated in terms of two elements i-e likelihood and impact which offers a more comprehensive insight into their potential repercussions.
   3. To get over all risk score, next is the phase of merger of likelihood and impact of each risk. Risk level associated with each potential issue can be clearly indicated with this unified risk and hence assisting decision-makers in prioritizing their responses.
      1. **Data Analysis.**
5. In order to assess risks level in a company, it is crucial to compare them against industrial/ recommended practices. This comparison enables software houses/ firms to identify areas that need additional stringent measures and also in the formulation of efficient risk mitigation plans.
6. The process involves pinpointing risks with the highest scores. These risks represent the most significant threats to the organization and demand urgent action. By prioritization of these risks, companies can optimize their resource allocation and can focus on resolving the most critical concerns.
   * 1. **Data Collection**

The survey is conducted among three software firms (AppInSnap, Omnisoftex & Centangle Interactive) to have their insights on how they address the identified risks during the application development process. These responses are being utilized to comprehend how industry practices manage risks and validate the risk assessments. Through distributing the survey and analyzing the replies, a comprehensive understanding of risk management practices in the industry is being acquired, ensuring the accuracy of risk assessments. This interactive procedure provides a detailed examination of how various software firms handle security risks in their application development procedures.

* + 1. **Synthesis and Analysis**.

Risk assessment process’s output and feedback from the software firms are analyzed. Data obtained is meticulously reviewed to identify potential risks and grasp the feedback from the software companies. This feedback obtained from software firms is useful in guiding decision-making. Merger of response from software firms with that of risk assessments gathered through protection poker, helps in better understanding of security requirements, hence leading to the resolution of security issues early in the development phase.

Text in Red yet to be made AI Free

**Results**

**AppInSnap**

The survey is meant to identify security risk management practices at AppInSnap, more focused on how this organization carries out identification, assessment, and mitigation of security risks during the application development lifecycle. 15 members participated in the survey conducted for 1 hour and 20 minutes. Members from various teams including development, security, and project management shared their valuable feedback, survey reaveal’s AppInSnap approach towards risk management, the tools and techniques in use. Detailed response is given below.

|  |  |
| --- | --- |
| Question | Responses |
| . Identifying Risks: How does your software house identify the 46 security risks outlined in the security assurance model during the application development lifecycle? | AppInSnap employs a proactive approach to identify security risks. At the beginning of each project, they conduct thorough risk assessments using industry-standard frameworks and guidelines, such as ISO/IEC 27005 and NIST SP 800-30. The team collaborates closely with clients to understand their specific security requirements and potential vulnerabilities. Additionally, they use automated security testing tools (e.g., OWASP ZAP, Burp Suite) and manual code reviews to identify risks throughout the development process. The most frequently identified risks include SQL injection, cross-site scripting (XSS), and insecure authentication mechanisms. |
| 2. Risk Management: What strategies or methods do you use to manage and mitigate the identified risks? | AppInSnap's risk management strategy involves implementing robust security measures at every stage of the development lifecycle. They follow industry best practices and standards, such as OWASP guidelines and CIS Controls. The team continuously monitors and updates security protocols to address emerging threats and vulnerabilities promptly. Regular security audits and penetration testing are conducted to ensure application integrity. For instance, multi-factor authentication (MFA) and secure coding practices are enforced to mitigate identified risks effectively. |
| 3. Risk Assessment: How do you integrate risk assessment methods like the protection poker technique into your application development process? | Risk assessment methods like the protection poker technique are integrated into the application development process at AppInSnap. Protection poker sessions, lasting 2-3 hours, allow team members to collaboratively evaluate the likelihood and impact of each identified risk. This ensures a comprehensive understanding of potential threats. By involving stakeholders from various domains, including security and development, risks can be prioritized, and resources allocated more effectively. The sessions are typically held at the start of each major project phase and whenever significant changes to the system are planned. |
| 4. Risk Prioritization: Based on the risk estimation provided, how do you prioritize risks and allocate resources for their mitigation? | AppInSnap prioritizes risks according to their potential impact on the application and organization based on risk estimation from the protection poker technique. High-risk areas, such as those involving sensitive user data or critical application functions, receive immediate attention, with resources allocated accordingly for mitigation. The team regularly reviews and updates risk priorities as the project progresses. For example, risks associated with data breaches and unauthorized access are given top priority, ensuring that encryption and access controls are rigorously applied. |
| 5. Protection Techniques: Can you share your experiences with using protection poker or similar techniques to assess and address these risks during application development? | AppInSnap has found the structured approach of protection poker to be highly effective in making informed decisions and prioritizing mitigation efforts. The collaborative nature of the technique ensures that diverse perspectives are considered, leading to a more thorough evaluation of risks. Team members appreciate the interactive and inclusive format of the sessions, which helps in balancing security and development timelines. This method has been particularly useful in addressing risks such as data validation errors and configuration issues. |
| 6. Challenges and Solutions: What specific challenges have you faced in managing the aforementioned risks, and what strategies have you found most effective in addressing them? | The primary challenges faced by AppInSnap include keeping pace with evolving threats and maintaining a balance between security and development timelines. To address these challenges, they invest in continuous training and education for their team members, stay updated with the latest security trends, and implement agile security practices. For example, adopting a DevSecOps approach has allowed for seamless integration of security into the development process. Fostering a culture of security awareness and collaboration within the organization has also proven to be an effective strategy in mitigating risks. |

**OmniSoftex**

Through this research security risk management practices at OmniSoftex are analyzed. Based on a survey conducted with 9 participants from different teams, such as development, design, operations, and project management, Omnisoftex risk identification , assessment and mitigation techniques are ascertained. The survey included participants with different experiences and background from junior developers to senior managers, and hence evaluating OmniSoftex overall security posture while developing a software. Moreover, OmniSoftex also conducted protection poker sessions lasting approximately 50 - 55 minutes, where members mutually discusses, assess and prioritize security risks. Detailed response to follow :

|  |  |
| --- | --- |
| Question | Responses |
| . Identifying Risks: How does your software house identify the 46 security risks outlined in the security assurance model during the application development lifecycle? | OmniSoftex has established processes and protocols to identify the 46 security risks outlined in the security assurance model during the application development lifecycle. They conduct comprehensive threat modeling exercises, code reviews, and security assessments to identify potential vulnerabilities. The team collaborates closely with clients to understand their security requirements and prioritize risks based on their potential impact. The most frequently identified risks include cross-site scripting (XSS), SQL injection, and weak authentication mechanisms. |
| 2. Risk Management: What strategies or methods do you use to manage and mitigate the identified risks? | OmniSoftex's risk management approach focuses on implementing a multi-layered security framework to mitigate identified risks effectively. They follow industry best practices and standards, such as ISO 27001 and NIST, to ensure the security of their applications. Additionally, they utilize advanced security tools and technologies to monitor and detect security threats in real-time. For example, they use intrusion detection systems (IDS) and web application firewalls (WAF) to provide additional layers of protection. |
| 3. Risk Assessment: How do you integrate risk assessment methods like the protection poker technique into your application development process? | At OmniSoftex, risk assessment methods like the protection poker technique are integrated into the application development process. Protection poker sessions, lasting 3-4 hours, allow team members to systematically evaluate and prioritize risks based on their likelihood and impact. By involving stakeholders from different departments, including security, development, and operations, they ensure a holistic approach to risk assessment. These sessions are typically held quarterly and whenever significant changes to the system are planned. |
| 4. Risk Prioritization: Based on the risk estimation provided, how do you prioritize risks and allocate resources for their mitigation? | Based on the risk estimation provided by the protection poker technique, OmniSoftex prioritizes risks by considering their potential impact on the confidentiality, integrity, and availability of the application. High-risk areas, such as those involving sensitive user data or critical application functions, are addressed with immediate attention, and resources are allocated accordingly for mitigation. The team continuously monitors and updates risk priorities to adapt to evolving threats. For instance, risks associated with data breaches and unauthorized access are given top priority, ensuring robust encryption and access controls are in place. |
| 5. Protection Techniques: Can you share your experiences with using protection poker or similar techniques to assess and address these risks during application development? | OmniSoftex has found the protection poker technique to be highly effective in fostering collaboration and consensus-building among team members. The structured approach of protection poker enables them to prioritize risks effectively and implement targeted mitigation strategies. Team members appreciate the interactive and inclusive format of the sessions, which helps in balancing security and development timelines. This method has been particularly useful in addressing risks such as configuration issues and data validation errors. |
| 6. Challenges and Solutions: What specific challenges have you faced in managing the aforementioned risks, and what strategies have you found most effective in addressing them? | In managing security risks, OmniSoftex faces challenges such as the complexity of integrating security into the development lifecycle and balancing security requirements with project timelines. To address these challenges, they have invested in automation tools, threat intelligence platforms, and continuous training for their team members. For example, using automated security testing tools and integrating security checks into the CI/CD pipeline have proven to be effective strategies. Additionally, fostering a culture of security awareness and accountability within their organization has been instrumental in mitigating risks effectively. |

**Centangle Interactive**

This study analyzes the security practices at Centangle Interactive, mainly how it identifies, evaluates, and mitigates security risks throughout the software development lifecycle. Survey lasted for approximately an hour with 17 participants from various teams, experience and background. The survey is more focused on what are the Centangle Interactive’s risk management approaches, the procedures employed, and security related challenges experienced by team mebers. Feedback from members participated is tabulated below :

|  |  |
| --- | --- |
| Question | Responses |
| 1. Identifying Risks: How does your software house identify the 46 security risks outlined in the security assurance model during the application development lifecycle? | Centangle Interactive employs a proactive approach to identify security risks. They conduct thorough risk assessments and security audits using industry-standard frameworks and guidelines, such as ISO/IEC 27001 and NIST SP 800-30. The team collaborates closely with clients to understand their specific security requirements and potential vulnerabilities. Additionally, they use automated security testing tools (e.g., Nessus, Qualys) and manual code reviews to identify risks throughout the development process. The most frequently identified risks include SQL injection, cross-site scripting (XSS), and insecure authentication mechanisms. |
| 2. Risk Management: What strategies or methods do you use to manage and mitigate the identified risks? | Centangle Interactive's risk management strategy involves implementing comprehensive security controls and protocols to mitigate identified risks effectively. They follow a defense-in-depth approach, combining technical controls (e.g., firewalls, intrusion detection systems), security policies, and user awareness training to safeguard their applications. Regular security testing and vulnerability assessments are conducted to identify and address potential vulnerabilities. Specific measures include encryption of sensitive data, regular patch management, and secure coding practices. |
| 3. Risk Assessment: How do you integrate risk assessment methods like the protection poker technique into your application development process? | Risk assessment methods like the protection poker technique are integrated into the application development process at Centangle Interactive. Protection poker sessions, lasting 1.5-2 hours, allow team members to collaboratively evaluate the likelihood and impact of each identified risk. This ensures a comprehensive understanding of potential threats. By involving stakeholders from various domains, including security, development, and business, risks can be prioritized, and resources allocated more effectively. The sessions are typically held at the start of each major project phase and whenever significant changes to the system are planned. |
| 4. Risk Prioritization: Based on the risk estimation provided, how do you prioritize risks and allocate resources for their mitigation? | Centangle Interactive prioritizes risks according to their potential impact on the confidentiality, integrity, and availability of the application, based on risk estimation from the protection poker technique. High-risk areas, such as those involving sensitive user data or critical application functions, receive immediate attention, with resources allocated accordingly for mitigation. The team regularly reviews and updates risk priorities to adapt to the changing threat landscape. For example, risks associated with data breaches and unauthorized access are given top priority, ensuring that encryption and access controls are rigorously applied. |
| 5. Protection Techniques: Can you share your experiences with using protection poker or similar techniques to assess and address these risks during application development? | Centangle Interactive has found the structured approach of protection poker to be highly effective in making informed decisions and prioritizing mitigation efforts. The collaborative nature of the technique ensures that diverse perspectives are considered, leading to a more thorough evaluation of risks. Team members appreciate the interactive and inclusive format of the sessions, which helps in balancing security and development timelines. This method has been particularly useful in addressing risks such as data validation errors and configuration issues. |
| 6. Challenges and Solutions: What specific challenges have you faced in managing the aforementioned risks, and what strategies have you found most effective in addressing them? | The primary challenges faced by Centangle Interactive include resource constraints, evolving threat landscapes, and compliance requirements. To address these challenges, they invest in automation tools (e.g., automated code scanning tools), threat intelligence platforms, and continuous training for their team members. Adopting a DevSecOps approach has allowed for seamless integration of security into the development process. Fostering a culture of security awareness and accountability within the organization has also proven to be an effective strategy in mitigating risks. |

**DISCUSSION AND ANALYSIS**

Study of the feedback from three software houses (AppInSnap, OmniSoftex, and Centangle Interactive) shows what is their risk management approach, how they use protection poker, and how do they prioritize risks based on likelihood and impact. Risk assessment feedback reveals variations in the likelihood, impact, and overall risk scores assigned to each security risks. In order to convincingly understand and state the working ethics better out of the three software houses, the risk assessment methodologies employed are examined and explained below :-

**OmniSoftex:**

* Total Risk Score: 1911
* Number of Risks: 46
* Average Risk Score: 41.54

**AppInSnap:**

* Total Risk Score: 2309
* Number of Risks: 46
* Average Risk Score: 50.19

**Centangle Interactive:**

* Total Risk Score: 2348
* Number of Risks: 46
* Average Risk Score: 51.04

**AppInSnap:** AppInSnap's risk assessment methodology appears to focus on a balanced evaluation of likelihood and impact, resulting in moderate overall risk scores across most risk types. They prioritize risks such as Injection, Sensitive Data Exposure, and Buffer Overflow, which have high likelihoods and significant impacts, leading to higher overall risk scores. This approach suggests a methodical assessment process, where risks are carefully evaluated based on their potential to compromise application security and integrity.

**OmniSoftex:** OmniSoftex's risk assessment methodology seems to place more emphasis on the impact of risks rather than their likelihood. While they assign relatively high impact scores to risks like Sensitive Data Exposure and Injection, the likelihood scores for these risks are lower compared to other software houses. This approach indicates a focus on mitigating risks with significant potential impact, aligning with a strategy of prioritizing resources based on potential consequences rather than the probability of occurrence.

**Centangle Interactive:** Centangle Interactive's risk assessment methodology stands out for its comprehensive evaluation of both likelihood and impact, resulting in higher overall risk scores across several risk types. They prioritize risks such as Injection, Buffer Overflow, and Elevation of Privilege, which have high likelihoods and significant impacts, leading to elevated overall risk scores. This approach suggests a thorough risk assessment process, where risks are rigorously evaluated based on their potential to exploit vulnerabilities and compromise application security.

**Conclusion:**

Centangle Interactive's risk assessment methodology evident from its higher average risk score surfaces as well prepared and well apprehended in comparison to the other two software houses. Methodical evaluation of both likelihood and impact of each risk reveals Centangle Interactive is more sensitized and hence demonstrates a proactive approach to risk management, thereby guarantying detailed examination of likely threats to application security. Though certain limitations of practical implementation of Protection Poker Technique have also been revealed like better integration of protection poker technique into project management activities but serves an opportunity as a new avenue for future work Top of Form

**References**

[1]. H. Villamizar, A. Anderlin Neto, M. Kalinowski, A. Garcia and D. Méndez, "**An Approach for Reviewing Security-Related Aspects in Agile Requirements Specifications of Web Applications**,"  *IEEE 27th International Requirements Engineering Conference (RE)*, Jeju, Korea (South), pp. 86-97, 2019, doi: 10.1109/RE.2019.00020.

[2]. Martin Otieno, David Odera and Jairus Ekume Ounza **“Theory and practice in secure software development lifecycle: A comprehensive Survey”**  , World Journal of Advanced Research and Reviews, 18(03), pp 53–78, 2023**.**

[3]. Wisdom Umeugo , Kimberly Lowrey and Shardul Y Pandya, “**Factors Affecting The Adoption Of Secure Software Practices In Small And Medium Enterprises That Build Software In-house**,” International Journal of Advanced Research in Computer Science, Volume 14(2), 2023.

[4]. R. A. Khan, S. U. Khan, M. Alzahrani and M. Ilyas, "**Security Assurance Model of Software Development for Global Software Development Vendors**," in IEEE Access, vol. 10, pp. 58458-58487, 2022, doi: 10.1109/ACCESS.2022.3178301.

[5] Mamdouh Alenezi and Sadiq Almuairfi “**Security Risks in the Software Development Lifecycle**” International Journal of Recent Technology and Engineering, Vol 8 (3), 2019, DOI:10.35940/ijrte.C5374.098319 .

[6]. W. Wang, Q. Zeng and A. P. Mathur, "**A Security Assurance Framework Combining Formal Verification and Security Functional Testing**,"  *12th International Conference on Quality Software*, Xi'an, China, pp. 136-139, doi: 10.1109/QSIC.2012.

[7] Tøndel, I.A., Jaatun, M.G., Cruzes, D.S. and Williams, L. (2019), "**Collaborative security risk estimation in agile software development**", *Information and Computer Security*, Vol. 27 No. 4, pp. 508-535.

[8] K. Qian, R. M. Parizi and D. Lo, "**OWASP Risk Analysis Driven Security Requirements Specification for Secure Android Mobile Software Development**," *IEEE Conference on Dependable and Secure Computing (DSC)*, Kaohsiung, Taiwan, pp. 1-2, 2018 doi: 10.1109/DESEC.2018.8625114.

[9]. C. Onwubiko, "**Security operations centre: Situation awareness, threat intelligence and cybercrime**,"  *International Conference on Cyber Security And Protection Of Digital Services (Cyber Security)*, London, UK, pp. 1-6, 2017 doi: 10.1109/CyberSecPODS.2017.8074844.

[10] S. -J. Chen, Y. -C. Pan, Y. -W. Ma and C. -M. Chiang, "**The Impact of the Practical Security Test during the Software Development Lifecycle**," 2022 24th International Conference on Advanced Communication Technology (ICACT), PyeongChang Kwangwoon\_Do, Korea, Republic of, 2022, pp. 313-316, doi: 10.23919/ICACT53585.2022.9728868.

[11] Y. Zeng, Y. Cheng, G. Xie and R. Wang, "**Design of Mobile Application Lifecycle Security Management Platform**," 2021 International Conference on Computer Network, Electronic and Automation (ICCNEA), Xi'an, China, 2021, pp. 26-30, doi: 10.1109/ICCNEA53019.2021.00017

[12] Al-Darwiash, A.I. and Choe, P. (2019), “**A framework of information security integrated with human factors”, International Conference on Human-Computer Interaction (HCII)**, Springer, pp. 217-229. Arteaga, J.M., Gonzalez, R

[13] E. Khanna, R. Popli and N. Chauhan, "**Identification and Classification of Risk Factors in Distributed Agile Software Development**," in *Journal of Web Engineering*, vol. 21, no. 6, pp. 1831-1851, September 2022, doi: 10.13052/jwe1540-9589.2164.

[14] O. Kovalenko, O. Smirnov, A. Kovalenko and S. Kavun, "**Quantitative Risk Assessment Method Development in the Context of the SDLC-model**," *2021 IEEE 8th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T)*, Kharkiv, Ukraine, 2021, pp. 203-208, doi: 10.1109/PICST54195.2021.9772143.

[15] Samar Al-Saqqa, Samer Sawalha and Hiba AbdelNabi “**Agile Software Development: Methodologies and Trends**”, International Journal of Interactive Mobile Technologies, Vol. 14, No. 11, pp 246-270, 2020.