ON SOFTWARE SECURITY REQUIREMENTS ELICITATION AND ANALYSIS METHODS

Javed Ahmad[1], Chaudhary Wali Mohammad[1] and Mohd. Sadiq[2]

Abstract **-** Security requirements play an important role to protect valuable data and information from unauthorized users. The elicitation of the security requirements in the early phase of the software development process can help the software engineers to successfully develop the secure information system. Different methods have been developed for the elicitation of the security requirements like multilateral security requirements analysis, software quality requirements engineering, knowledge acquisition for automated specifications, etc. The objective of this paper is to evaluate the different software Security Requirements Elicitation and Analysis (SecREA) methods based on the following parameters: (a) stakeholders' participation, (b) identification of the functional requirements which needs more security, (c) selection and prioritization of the security requirements under crisp and fuzzy environment, (d) common activities in SecREA methods. After evaluating the SecREA methods, we discuss the scope for future work.

Keywords **-** Security engineering methods, Security requirements, Security requirements elicitation

I. INTRODUCTION

TO deal with daily virus alerts, malicious crackers, and the threats of cyber terrorism, the security requirements in different types of the systems like information systems, web-based systems, cloud computing, electronic information systems, etc., have received much attention by the requirements engineers at the time of requirements elicitation process [1][2]. The security in these systems is essential because of the increasing usage of their services. Since it is targeted by adversaries, so in these systems, login is a critical step for the user authentication, which is an important security requirement. The password which is used to login the system can be steal by the hackers so there should be a secure transmission of passwords from the client system to the server to protect the passwords of users [3]. It has been observed that requirements engineers are not trained to elicit the security requirements and those who are trained they only have knowledge about the security-specific architectural constraints like passwords and encryption [4]. As pointed out by Firesmith [4], the security requirements "ensures that users and client applications can only access data and services for which they have been properly authorized", "confidential communications and data are kept private", etc. For authentication different methods have been developed like two-factor authentication (2FA), 2FA communication channel based on steganography in the QR code, etc. [5].

There are different types of security requirements, i.e., identification requirements, authentication requirements, authorization requirements, immunity requirements, integrity requirements, intrusion detection requirements, non-repudiation requirements, privacy requirements, security auditing requirements, survivability requirements, physical protection requirements, and system maintenance security requirements [4]. These requirements are the need of the stakeholders which are identified with the help of different software requirements elicitation techniques (SRETs) like traditional techniques, group elicitation technique, and cognitive technique, goal-oriented techniques, etc. [6]. The elicited requirements are implemented during software development process so that the need of the stakeholders can be fulfilled and a secure system can be developed. Stakeholders' identification is one of the important activities of SRETs. Despite its importance in SRETs, less attention is given in the literature for the identification of the stakeholders in security requirements elicitation process. Practically, it is not possible to elicit the SRs if the stakeholders have not been identified before the elicitation of the software requirements. So, stakeholders should be identified before the starting of the software requirements elicitation process [7].

Software requirements are classified into functional requirements (FRs) and non-functional requirements (NFRs). FRs describes the functionality of the software. For example, in an Institute Examination System (IES), "to download the admit card of a student" is an FR [7]. NFRs describe the non-behavioural aspects of the system or "how the system is supposed to be". For example, the "IES should be secure" is an example of NFR [7]. Mairiza

1 Department of Applied Sciences and Humanities, Faculty of Engineering and Technology, JamiaMilliaIslamia (A Central University), New Delhi,110025, India (e-mail: javed.jmi08@gmail.com, cmohammad@jmi.ac.in)

2 Department of Computer Science and Automation, Indian Institute of Science Bangalore, Karnataka-560012, India (e-mail: msadiq@jmi.ac.in, sadiq.jmi@gmail.com)

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

and Zowghi [8] investigated the notions of the NFRs; and as a result, they found 114 types of NFRs like privacy, security, reliability, etc. Among various NFRs, they have identified only 23 NFRs which have definition and attributes, i.e., "accessibility, adaptability, availability, efficiency, fault tolerance, functionality, integrability, integrity, maintainability, modifiability, performance, portability, privacy, readability, reliability, reusability, robustness, safety, scalability, security, testability, understandability, and usability". As mentioned in California breach Report, published in 2016, that "nearly 50 million records of Californians have been breached and the majority of these breaches resulted from security failures" [9]. One of the reasons of such type of the failure is the software security requirements. In literature, researchers have focused more on NFRs because neglecting NFRs may lead to the failure of the software [8]. For example, the failure of the London Ambulance System [10][11], New Jersey Department of Motor Vehicles Licensing System [12], was due to lack of NFRs. In addition to this, researchers and academicians have also introduced some new kind of NFRs in order to fulfill the need of the stakeholders and system. For example, [13] introduced two new kinds of NFRs in the context of Ubiquitous Computing (UC) and Internet of Things (IoT), i.e., Context-awareness and mobility. In their work, the author has pointed out that these two NFRs can impact traditional NFRs like security and usability. Based on our review, we found that among various NFRs, security plays an important role to protect the important information of the stakeholders of any kind of the domain, i.e., IES, UC and IoT. Therefore, in this paper, we mainly focus on the "security requirements elicitation" and "security requirements analysis". Hereafter, these two sub-processes of security requirements engineering (RE) are referred to as "Security Requirements Elicitation and Analysis" (SecREA).

In literature, different methods have been developed to identify the security requirements [14] like "multilateral security requirements analysis" (MSRA), "software quality requirements engineering" (SQUARE), "knowledge acquisition for automated specifications" (KAOS), etc. A successful software security requirements elicitation technique is one that supports the identification of the stakeholders because they are the main source of the software requirements [7]. Stakeholders are those people in an organization who are directly or indirectly related to a project. Stakeholders are the main sources of the requirements. It is indispensable to identify the stakeholders before the starting of the requirements elicitation process. Lack of stakeholder's participation during the elicitation process is one of the reasons of software failure [15]. One of the important activities of SecREA methods is the stakeholders' identification. In our work, it is considered as one of the criterion for the evaluation of the SecREA methods. After requirements elicitation process, a system may have hundreds or thousands of requirements and it is not possible to secure each and every requirement because of the budget and other constraints of an organization [16][17]. There should be a systematic way to select and prioritize the requirements which needs more security attention. So in this study, "identification of the FRs which needs more security attention" is considered as the criterion for the evaluation of the SecREA methods. Different types of the data are used during the evaluation of the FRs like crisp, rough, and fuzzy data. In this study, we have used fuzzy-based approaches for the evaluation of the FRs because in real-life applications linguistic variables are employed to specify the preferences of FRs and NFRs during the decision making process. Fuzzy logic is used to model the linguistic variables by using the different types of the fuzzy numbers like triangular fuzzy numbers, trapezoidal fuzzy numbers, etc. Therefore, in this study, "selection and prioritization of SRs under fuzzy environment" is considered as a criterion for the evaluation of the SecREA methods [18]. Based on our review, we found that different activities are involved in the SecREA methods; and we have considered these activities of SecREA methods as a criterion so that the common activities of the SecREA methods can be identified. Different studies have performed the evaluation and comparative analysis of different SecREA methods based on different criteria, i.e., risk, analysis, risk identification, support of tools, etc. But little attention is given to evaluate the SecREA methods based on the following criteria (C):

C1: stakeholders' identification, C2: identification of the FRs which needs more security attention, C3: selection and prioritization of SRs under fuzzy environment, and C4: common activities in SecREA methods. The contributions of the paper are as follows:

1. Classification of the SecREA methods
2. Evaluation of the SecREA methods based on the criteria C1, C2, C3, and C4.

This paper is organized as follows: Section II presents the related work. Classification and evaluation of SecREA methods are presented in section III and section IV, respectively. Discussion related to the evaluation of the SecREA methods is given in section V. Finally, section VI concludes the paper.

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

II. RELATED WORK

A security requirement is an NFR whose objective is to protect valuable information from unauthorized users. As pointed out by Fabian et al. [14] that security requirements cannot be elicited until we know that what to secure, against whom, and to what extent on the basis of the following credo of the requirements engineering: "If you don't know what you want, it is hard to do it right". In literature, different methods have been proposed for SecREA [14][19] like SQUARE, Misuse cases, KAOS, "multilateral security requirements analysis" (MSRA), "secure Unified Modeling Language" (UML), "goal-based requirements analysis methods" (GBRAM), "security engineering process using patterns" (SEPP), "security requirements engineering framework (SREF), CORAS, "model-based information system security risk management" (ISSRM), and "security requirements engineering process" (SREP).

To investigate the strength and limitations of the existing SecREA, many reviews have been conducted [20]. In addition to the review, there are also few studies that focus on the evaluation and analysis of the SecREA methods. For example, Fabian et al. [14] compared and classified the different security requirements engineering methods. In their work, they have collected the papers for the comparison that have been published from 1999 to 2008. Ramesh and Reddy [21] presented a survey of 15 security requirements elicitation methods by considering the classification, merits, and demerits. Jaiswal and Gupta [22] presented an in-depth analysis of the security engineering methods. In their work, they have classified security engineering methods on the basis of the following: (a) use-case based approach (b) goal-oriented approach, and (c) process-oriented approach. At a glance these studies have presented the analysis of the existing SecREA methods. Nonetheless, a close look reveals two main limitations of the existing studies. Firstly, these reviews have not focused on (a) stakeholder's participation, (b) identification of the functional requirements which needs more security, (c) selection and prioritization of the security requirements under crisp and fuzzy environment (d) common activities of SecREA methods. Secondly, new SecREA methods have been introduced in the literature so an up-to-date analysis of the existing work is needed. Such evaluation and analysis are helpful for researchers in improving the current state-of the art in the area of SecREA methods.

III. CLASSIFICATION OF SECREA METHODS

Software requirements are classified into FRs and NFRs. We have further classified the NFRs into five types, i.e., "performance", "reliability", "usability", "security", and "maintainability" because these requirements are used as criteria during decision-making process [8][23][24], as shown in Fig. 1.

A. Performance Requirement

Performance requirement is an NFR which focuses on low response time and high throughput. For example, the FRs for an IES is that it should authorize only bonafide students to download the admit card for the current examination and the performance NFR is that the authorization should be done quickly. So, an implemented system must have good performance [23].

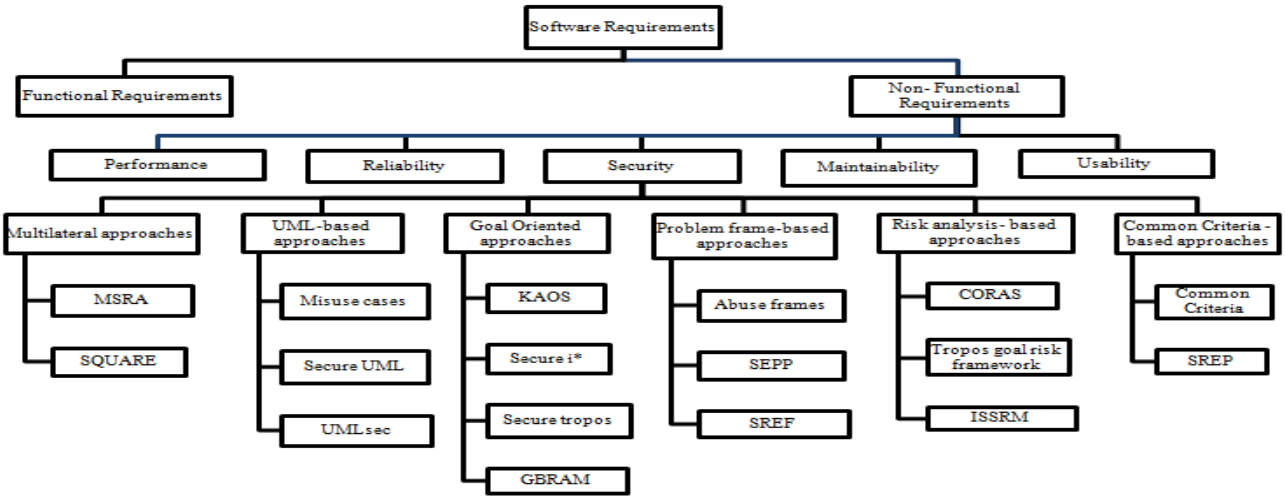

Fig.1 Classification of SecREA methods

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

B. Reliability Requirement

Reliability requirement is the probability of failure-free operation of the requirements of a system defined over a period of time, i.e., (t,t+x]. It is given that the most recent failure occurred at time t ($x \geq 0$) [24]. Different software metrics have been developed for different types of the systems. For example, the probability of failure on demand (POFOD) is defined for safety-critical systems. It can be defined as "the likelihood that the system will fail when the request is made". In practical applications, the POFOD = 0.002 means that 2 in 1000 requests may result in failure. The rate of occurrence of failure (ROCOF) deals with the frequency of failures. If ROCOF = 0.01 then it means that 1 failure are likely in each 100 time units. Such kind of system is used in transaction processing time. Another useful reliability metric is availability, which means that how likely a system is available for use including the repair and other down-time. This is used for continuously running systems. For example, telephone switching system, submission of the applications in an IES, etc. The availability = 0.995 means that the system is available 995 out of 1000-time units. Mean time to failure (MTTF) is the measure of the time between failures. The MTTF =500 means that average of 500-time units passes between failures. It is used for system having long transactions.

C. Security Requirements

The need of the security requirements arises when stakeholders that some resources in the system, i.e., information or reputation, cash, etc., is valuable to their organization. Such resources are called assets and stakeholders aim is to protect them from any misuse involving these assets. Security requirements express this inclination by describing the involved assets and the misuse to be averted. Practically, it is not possible to implement 100 % secure system because it can increase the cost and schedule of the system. It has been identified that programming languages also play an important role to produce security vulnerabilities. For example, C language can result in an array of boundary defects that can be exploited to run malicious scripts [25]. In security-based applications, one-time passwords (OTP) are the most preferred solution for single session [26]; and prediction in an OTP can create serious threats to information security. Sometimes, it is also important to protect the assets from bots. Therefore, in web applications, CAPTCHA is employed to differentiate between a human being and a bot [27]. In Fig. 1 methods to elicit and analyze the security requirements have been classified into six approaches and a brief discussion about these approaches is given below:

1. Multilateral Approaches

In multilateral approach the security interests of the stakeholders are expressed in terms of the security goals [28]. The main aim of MSRA method is to analyse the privacy and security needs of different stakeholders having different views [29]. The different views from different stakeholders are collected, conflicts are identified and views are integrated. MSRA is a seven-step method to elicit security goals and later provide suggestions to compose final security requirements. On the other hand, SQUARE methodology is a nine step method developed by Nancy Mead et al. [30] in 2005 at Software Engineering Institute's Networked System Survivability program of Carnegie Mellon University to facilitate organizations to incorporate security in the early phase of the software development lifecycle. SQUARE methodology is used to elicit and prioritize security requirements [30]. The aim of this methodology is to integrate the security requirements engineering in software development processes.

2. UML-Based Approaches

UML is considered as de-facto standards for object-oriented modeling. Misuse case diagram is an extension of the Use case diagram with negative cases that specify the unwanted behavior in the proposed system to elicit security requirements [31]. Secure UML is an UML-based modeling language to model the requirements for secure and distributed systems. To deal with the access control information Secure UML is the best suited modeling language that defines a vocabulary for annotating UML based models [32]. UMLsec is an extension to UML for integrating information related to security[33]. The information covers security properties like secure information flow, access control, and confidentiality.

3. Goal-Oriented Approaches

The goal-oriented model uses AND/OR refinement tree to model the goal of the stakeholders by asking questions why (for up elaboration) and how (for down elaboration) to the leaves of the refined tree [34]. KAOS follows goal oriented approach for elaborating and formalizing security requirements using anti-models, linear temporal logic, and conflict analysis. KAOS is mainly concerned with the identification of security objectives, conceptualizing objectives into requirements, and assigning responsibilities to the actors [35]. Secure Tropos is used for the development of multi- agent systems [36]. It integrates the concept of trust and security in the Tropos model. Tropos uses the following concepts: actor, goals, resource, and dependency. GBRAM uses goal and

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

scenario-driven requirements engineering to formulate privacy and security policies [37]. GBRAM is specifically used to elicit the security requirements which are already integrated into policies for analysis and elaboration of organisational goals.

4. Problem Frame-Based Approach

Problem frames are used in requirements engineering process to specify the software development problems. It is used to model the software problems with the reuse of the previous knowledge [38]. Following methods adopt the problem frame-based approach: abuse frames, SEPP, and SREF. Lin et al [39] proposed abuse frames using Jackson's problem frames [38] to identify threats from the viewpoint of the malign user. Abuse frame uses anti-requirements of malign user to represent threat. Anti-requirement is a notion to represent negative requirements. SEPP uses the concept of security problem frames (SPF) and concretized security problem frames (CSPF) [40][41]. SPFs are problem frames which deal with security requirements and consider only problems relating to security without anticipating solutions. Whereas, CSPF uses generic security mechanisms to solve a security problem to convert security requirements into concretized security requirements. SREF is a four step iterative method based on constructing a context for the application or system using problem frame-based approach and constraints are used as security requirements to evaluate and develop satisfaction arguments for security requirements [25].

5. Risk Analysis Based Approach

The activities involved in risk analysis constitute of analyzing threat, vulnerability, and the adverse impact on each component of the system [42]. CORAS, Tropos goal risk framework, and ISSRM follow risk analysis approach. CORAS is a seven step model for risk analysis. It provides a detailed guideline to explain the working procedure of the language to model threat and risk analysis [43]. Tropos goal risk framework considers the risk based on trust relations among actors. ISSRM is a risk analysis model consisting of four activities and it uses i* modeling for security requirements analysis [42, 45].

6. Common Criteria Based Approaches

Common Criteria and SREP method follow the common criteria based approaches. The international standard (ISO/ IEC 15408) for computer security is referred to as Common Criteria [46]. It covers user, developer, and evaluators to specify their requirements. Users specify security requirements, developers to specify security properties of product, and evaluators to compare the product to the claim product. SREP is a systematic process developed by Mellado et al. [47] to handle security requirements at every phase of development which includes the following phases such as analysis, design, implementation, and testing.

D. Maintainability Requirements

Maintainability requirement can be defined as the probability that a system can be repaired in a defined environment within a specified period of time. A system will take a short time to repair if it is maintained for long time. The concepts used in maintainability requirement are as follows: "failure mode, effect, and critically analysis", "failure mode and effect analysis", "mean time to repair and system downtime", etc. [48].

E. Usability Requirements

Usability requirement describes how easy a system can be understood for its use and how efficiently it can be used to perform its tasks [49] [50]. According to ISO 9241-11, the usability can be defined as ''the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use'' [51].

IV. EVALUATION OF SECREA METHODS

In this section, we evaluate the SecREA methods to answer the following research questions: RQ-1: How SecREA methods deal with the stakeholder's identification activity? RQ-2: Is there any support in the existing SecREA methods to identify those functional requirements, which needs security? RQ-3: Is there any support in the existing SecREA methods to deal with imprecise and vague data during the prioritization process? RQ-4: Which activity is common in SecREA methods?

A.RQ-1: How SecREA methods deal with the stakeholder's identification activity?

There are different steps or activities which are involved in different SecREA methods. Stakeholder identification is an important activity that should be identified before the starting of the requirements elicitation process. In MSRA method, different steps are used for the analysis of the privacy and security needs of different stakeholders [52]. In real-life applications, each stakeholder has different views for the software requirements. Therefore, these views are collected, conflicts are identified and views are integrated. The steps of the MSRA method include the following: (1) Identify Stakeholders:

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

stakeholders are the entities which are directly or indirectly involved with the system; (2) Identify episodes: episodes partition the security goals and are later used in identifying conflicts between multiple security goals; (3) Elaborate security goals: identify and elaborate the security goals of the different stakeholders for each of the episodes (4) Identify facts and assumptions: facts and assumptions are identified; (5) Refine stakeholder views on episodes: the stakeholder views taking facts, assumptions and the relationship between episodes are refined (6) Reconcile security goals: conflicts between security goals are identified, compromises between conflicting goals are found, and consistent set of security goals are established; (7) finally, security and functional requirements are reconciled. There is a variation of MSRA which is known as confidentiality requirements elicitation and engineering method. This method focuses only on confidentiality requirements. In the series of the multilateral approaches, SQUARE methodology is a complete security RE methodology in which security requirements are elicited and prioritized by the stakeholders and requirements engineer [30]. This method has nine steps, i.e., (1) agree on definitions, (2) identify security goals, (3) develop artefacts, (4) perform risk assessment, (5) select elicitation techniques, (6) elicit security requirements, (7) categorize requirements, (8) prioritize requirements, and finally, (9) requirements inspection. Based on the critical analysis of these two multilateral approaches, i.e., MSRA and SQUARE method, we found that some attention is given to stakeholder identification and also to deal with the different views of the stakeholders in MSRA method. In SQUARE methodology, little attention is given to the stakeholder identification and analysis.

In UML-based approaches, misuse cases focus on the negative aspects of the system. It is used to identify the misuser and misuse cases which help to identify the security requirements. This methodology focuses on the graphical and textual representation of security threats. An extended version of the UML was developed by Lodderstedt et al. [32], which specifies the role-based access control to represent the requirements. It deals with the role-based access control policies, confidentiality, and integrity goals. In the series of the UML based approaches, UMLSec was developed for security critical system [33]. This method covers all the three CIA goals [14]. In these methods, the main attention is on the representation of the security goals and there is no support of the stakeholder identification methods in these approaches.

Goal-oriented requirements engineering focuses on why and how of a system rather that what of a system [18]. Among various GORE techniques, KAOS is used to identify the security requirements. This methodology takes into account the multiple stakeholders and their views for the intended system. It focuses mainly on the completeness, consistency, and feasibility of requirements. Tropos is an agent-oriented methodology for software development. Whereas, Secure Tropos is an extension of the Tropos.

Abuse frames is a problem frame-based approach which is used to deal with the viewpoints of malicious users for the identification of the threats. The security requirements are identified from the threats by considering the following steps: (1) identify problem and sub-problems using common problem frames, (2) identify the threats and construct abuse frame diagrams, (3) identify security vulnerabilities, and (4) resolve security vulnerabilities. The security problem frames were used to develop the SEPP. In this method, the main focus was on confidentiality and integrity and not on availability [14]. The SREF was developed to elicit and analyse the security requirements. This method includes the following steps: (1) identify FRs, (2) identify security goals, (3) identify security requirements, and (4) construct satisfaction arguments.

CORAS, Tropos goal risk framework, and ISSRM are some of the security requirements elicitation methods that are based on risk analysis. The CORAS method is used for threat and risk analysis. There are seven steps in the CORAS [43], i.e., (1) discussion on goal between analyst and client; (2) identification of the threats, vulnerabilities, and threat scenarios; (3) document is approved after some refinement; (4) identification of the threats and incidents; (5) discussion on the consequences and likelihood in a workshop; (6) risk analysis; and (7) cost and benefit analysis. In risk-based methods, the Tropos goal risk framework is used to access the risk among actors based on trust relations [44]. In this method, GR-Tool is used for the identification of the security requirements. The ISSRM is a security risk management which includes the following steps: (1) context analysis and asset identification; (2) security goal identification; (3) identification of the security requirements from security goals. Based on our review, we found that similar to problem frame-based method, the risk-based methods also do not support the stakeholder identification activity [53]. In common criteria-based methods; there is no support for stakeholder's identification. The common criteria method of security requirements elicitation is adopted from the common criteria project which was established in 1995. It includes the following, i.e., trusted computer system evaluation criteria, Canadian trusted computer product evaluation criteria, and information technology security

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

evaluation criteria. The SREP integrates the security standards common criteria to handle security requirements [47]. The SREP includes the following steps: (1) agree on definitions, (2) identify vulnerable and/or critical assets, (3) identify security objectives and dependencies, (4) identify threats and develop artefacts, (5) risk assessment (6) elicit security requirements (7) categorize and prioritize requirements, (8) requirements inspection, and (9) repository improvement.

B. RQ-2: Is there any support in the existing SecREA methods to identify those functional requirements which need security?

It has been observed that a system cannot be 100 % secure. Therefore, it is an important issue that which set of functional requirements should be secured according to the consensus of the stakeholders. Based on our review, we found that only few methods of SecREA support the prioritization of requirements. For example, SQUARE methodology. The remaining methods have been developed to deal with the other issues of the security like risk analysis, threat analysis, etc. Prioritization of the requirements is supported by only MSRA, SQUARE, and SREP methodologies. Based on our review [7][18][54], we found that security has been considered as a criterion for the evaluation of the FRs of software. The objective of this evaluation is to compute the ranking values of the software requirements so that the selected set of the software requirements can be implemented in different releases of software. There is no method in the literature which elicits the security requirements for those FRs which are more important according to the stakeholders and their budget. Practically, it is not possible to provide the security to each and every requirement. Therefore, there should be some systematic methodology which identifies those requirements which needs more security for the development of the secure system.

C. RQ-3: Is there any support in the existing SecREA methods to deal with imprecise and vague data during the prioritization process?

Based on the evaluation of the SecREA methods, we found that only few methods support the prioritization requirements as an important step. For example, in SQUARE methodology crisp values are used during the prioritization process. In real-life applications, different stakeholders are involved during the prioritization process of the requirements; and these stakeholders specify their preferences on different software requirements using linguistic variables. For example, the system should be more secure, there should be medium security for some FRs. Here, more and medium are the linguistic variables. Fuzzy logic is an appropriate tool to deal with these linguistic variables. In literature different methods have been developed for the selection and prioritization of the software requirements. For example, Sadiq and Jain [18] developed a fuzzy based method for the prioritization of the software requirements. In SecREA method, there is no support to deal with imprecise and vague data during the prioritization of the requirements. Therefore, it is an open research issue that how to apply fuzzy based methods in SecREA methods.

The result of the evaluation of the SecREA methods based on RQ1, RQ2, RQ3, is exhibited in Table I.

Table I. Evaluation Of Secrea Methods Based On Rq1, Rq2, And Rq3

S. No.	SecREA methods	RQ1	RQ2	RQ3
1	MSRA	✓	✓	×
2	SQUARE	×	✓	×
3	KAOS	✓	×	×
4	Secure Tropos	✓	×	×
5	CORAS	✓	×	×
6	SREP	×	✓	×
7	Common Criteria	✓	×	×

From the results of Table 1, it is clear that only few methods support stakeholder's identification or participation of the stakeholders during the security requirements elicitation process like MSRA, KAOS, Secure Tropos, CORAS, and Common Criteria. Prioritization of requirements is considered as a key step in MSRA, SQUARE, and SREP methods. In these methods, only crisp values are used during the prioritization process, and there in no support to deal with imprecise and vague data during the decision-making process.

D. RQ-4: Which steps are common in SecREA methods?

After evaluating the SecREA methods based on the activities/steps involved in these methods, we found that there are some activities,which are common like identification of security goals, identify threats/security vulnerabilities, and risk assessment. For example, identification of security goal is an important step of the following SecREA methods like MSRA, SQUARE, SREP, and ISSRM because without eliciting the security goal, security requirements cannot be identified. Identify threats/security vulnerabilities is a key step of the SecREA methods like MSRA, Secure i*, Abuse frames, CC, Misuse case, and UMLSec. Similar to these steps, the risk identification is also a key step of the SecREA

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

methods like SQUARE, Abuse frames, SEPP, CORAS, Secure Tropos, Goal Risk Framework, ISSRM, SREP, CC, Misuse Case. The result of the evaluation based on the steps used in SecREA methods is exhibited in Table II.

V. DISCUSSIONS

Research in the area of security requirements engineering has been divided into the following sub-areas, i.e., (a) identification of threats [55], (b) identification of security requirements from threats [56][57][58][59][60], (c) integrated methods [61], and (d) comparison and evaluation of different SecREA methods [62][63][64]. For the identification and classification of the threats, Microsoft developed a STRIDE model, i.e., Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of privilege. The security property for these threats is exhibited in Table III.

Table III. Threat And Security Properties

Threat	Security property
Spoofing	Authentication
Tampering	Integrity
Repudiation	Non-Repudiation
Information Disclosure	Confidentiality
Denial of Service	Availability
Elevation of Privilege	Authorization

For the successful development of a secure system it is indispensable to identify the threats from the beginning of the requirements engineering process. The security requirements elicitation is the first sub-process of security requirements engineering and error occurring at this stage will affect the entire system. Therefore, it is important to identify the vulnerabilities in each stage of the software development process, i.e., requirements engineering, software design, analysis, and implementation. Vulnerabilities are the weaknesses in the different phases of the software development process which is misused by the misusers to compromise the system. To deal with the vulnerabilities, different methods have been developed for the elicitation of the security requirements [65]. There are different ways to identify the threat[66]. Threat is an important step of different SecREA methods and different methods have been proposed to elicit the security requirements from the identified threats using UML models [62], problem-based methodology [55][67], weaving and common criteria [57]. Few studies have focused on role-based access control during the elicitation of the security requirements to handle the role and responsibilities of different types of the stakeholders [59]. As system architecture also plays a key role to understand the security requirements. Some methods have been developed to avoid the assumption of the security architecture by considering the functional dependencies over system components [58]. Saeki et al. [56] developed a method for security requirements analysis using goal-oriented concepts. In this method, common criteria method was used for the elicitation of the security requirements. The security-related concepts such as assets, threats were extracted from security targets by considering the confidentiality, integrity, and availability. Mouratidis and Jurjens [68] developed an approach by integrating the "goal-oriented requirements engineering" and model-driven security engineering. Templates have also been used for the elicitation of the security requirements from the FRs [69][70].

Different methods have been developed by integrating the different SecREA methods. For example, Houmb et al. [61] proposed a method by combining CC, the heuristic requirements editor HeRA, and UMLsec. Comparative studies of different SecREA methods deal with the evaluation and analysis of two or more methods based on different criteria so that some new research direction can be identified for further work. Raspotnig and Opdahl [62] compared the risk identification techniques for safety and security requirements based on the following criteria: Time of use, stakeholders, type of systems, application area, layered view input, process, output, interoperability, scalability, creativity, and communication. Information security risk assessment is used to translate the security goals into security requirements. Suleiman and Svetinovic [64] evaluated the SQUARE methodology based on the following: (i) identification of the set of artefacts, threats, vulnerabilities, (ii) to perform the impact analysis, risk level determination, (iii) to elicit, categorize and prioritize the security requirements. A qualitative framework was used for the evaluation of the SQUARE methodology. The comparison of misuse cases and issue-based information systems was discussed by Ikram et al. [71]. The experimental comparison between misuse cases and threat identification was carried out by Opdahl and Sindre [72].

VI. CONCLUSION

Security requirements are non-functional requirements whose objective is to protect the assets and valuable information of an organization from unauthorized users or misusers. Different methods have been developed to elicit the security requirements of an information system like KAOS,

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

Table III. Evaluation of Secrea Methods Based on Rq

Steps/Methods	SQUARE	MSRA	KAOS	Secure Tropos	GBRAM	Secure i*	Abuse Frame	SEPP	SREF	CORAS	Tropos Goal Risk Framework	ISSRM	SREP	CC	Misuse Case	Secure UML	UML Sec
Identify Stakeholders	✗	✓	✓	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✓	✗	✗	✗
Identify Episodes	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Elaborate Security goals	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Identify facts and assumptions	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Refine stakeholder views on episodes	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Reconcile security & functional requirements	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Agree on definitions	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
Identify security goals	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Develop artifacts	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
Perform risk assessment	✓	✗	✗	✓	✓	✗	✓	✓	✗	✓	✗	✗	✓	✗	✗	✗	✗
Select elicitation techniques	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Elicit security requirements	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Categorize requirements	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
Prioritize requirements	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
Requirements inspection	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
Elaboration of security	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗
Security constraint modeling	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Security entities modeling	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Security capabilities modeling	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Identification heuristics	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Classification heuristics	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Elaboration heuristics	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Refinement heuristics	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Identify problem using common problem frames	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Identify threats/ Security Vulnerabilities	✗	✓	✗	✗	✗	✓	✓	✗	✗	✗	✗	✗	✗	✓	✓	✗	✓
Address security vulnerabilities	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Identify functional requirements	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗
Construct satisfaction arguments	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗
Introductory meeting	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
High level analysis	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
Approval meeting	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
Risk identification	✓	✗	✗	✗	✗	✗	✓	✓		✓	✓	✓	✓	✓	✓	✗	✗
Summary of risk consequences & likelihood	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
Treatment identification	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
Identify vulnerable and/or critical assets	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗
Identify security objectives and dependencies	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
Repository improvement	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
Context analysis	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

SQUARE, Secure Tropos, etc. In this paper, we have evaluated the different SecREA methods to identify the research gaps in the literature. Based on the evaluation, we found that stakeholder's identification is least considered activity of SecREA methods. It has been discussed that stakeholders are the key sources of the different types of the software requirements. Therefore, stakeholder identification activity must be performed before the starting of the security requirements elicitation process so that a secure system can be developed. Prioritization of the requirements is also an important activity of SecREA but it has also received less attention in security requirements engineering. One of the important factors of the successful development of the secure software is to find out the set of requirements that would be implemented during different releases of software. In real-life applications, a system may have several requirements so we cannot secure each and every requirement. Therefore, there must be a systematic way to prioritize the software requirements so that we can identify the security requirements for the selected set of the FRs to save the time, budget and schedule of a project. Existing SecREA methods do not support that how to deal with imprecise and vague data during the software prioritization process. Based on our review, we have also identified some activities which are common in the SecREA methods, i.e., identification of security goals, identify threats/security vulnerabilities, and risk assessment. In future, we shall try to develop a security requirements elicitation method,which would mainly focus on elicitation of security requirements from the selected set of FRs by considering the preferences of different stakeholders under fuzzy environment. We shall also try to compare the existing methods by considering real-life example like institute examination system, library management system, and ATM system, etc.

REFERENCES

[1] Beckers K., Côté I., Goeke L., SelimGüler S., and Heisel M., "A structured method for security requirements elicitation concerning the cloud computing domain".InternationalJournal of Secure Software Engineering, pp. 1-24, 2014.

[2] Cherdantseva Y., Burnap P., Blyth A., Eden P., Jones K., Soulsby H., Stoddart K., "A review of cyber security risk assessment methods for SCADA systems".Computers and Security, Vol. S6, pp. 1-27, 2016.

[3] Venkateswarlu, I. B. and Kakarla, J., "Password security by encryption using an extended ADFGVX cipher",International Journal of Information and Computer Security. Vol. 11, No.4/5, pp. 510-523, 2019.

[4] Firesmith D. G., "Engineering security requirements".Journal of Object Technology, Vol. 2, No. 1, pp. 53-68, 2003.

[5] Kouraogo Y., Orhanou G, and Elhajji S., "Advanced security of two-factor authentication system using stego QR code", International Journal of Information and Computer Security. Vol. 12, No.4, pp. 436-449, 2020.

[6] Sadiq M. and Jain S. K., "An insight into requirements engineering processes".3rd International Conference on Advances in Communication, Network, and Computing, LNICST, pp. 313-318, 2012.

[7] Sadiq M., "A fuzzy-set based approach for the prioritization of stakeholders on the basis of the importance of software requirements".IETE Journal of Research, Vol. 63, Issue 5, pp. 616-629, 2017.

[8] Mairiza D. and Zowghi D., "Constructing a catalogue of conflicts among non-functional requirements". In: Maciaszek, L.A., Loucopoulos, P. (eds.) ENASE 2010, CCIS 230, pp. 31– 44, 2011.

[9] Harris K. D., California Data Breach Report: https://oag.ca.gov/sites/all/files/agweb/pdfs/dbr/2016-data-breach-report.pdf. Accessed on January5, 2021.

[10] Breitman, K.K., Leite J. C. P., and Finkelstein A., "The World's Stage: A survey on Requirements Engineering using a real-life case Study".Journal of Brazilian Computer Society, Vol. 6, pp. 1-57, 1999.

[11] Finkelestein A. and Dowel J., "A Comedy of Errors: the London Ambulance Service case study".8th International Workshop on Software Specification and Design, pp. 2-5, 1996.

[12] Boehm B.W. and In H., "Identifying quality-requirements conflicts". IEEE Software, pp. 25-35, 1996.

[13] Carvalho R.M., "Dealing with conflicts between non-functional requirements of Ubicomp and IOT applications".IEEE 25th International Requirements Engineering Conference, pp. 544-549, 2017.

[14] Fabian B., Gurses S., Heisel M., Santen T., Schmidt H., "A Comparison of Security Requirements Engineering Methods".Requirements Engineering, Vol. 15,pp. 7-40, 2010.

[15] Pacheco C. and Garcia I., "A systematic literature review of stakeholder identification methods in requirements elicitation".Journal of Systems and Software, Vol. 85, Issue 9, pp. 2171-2181, 2012.

[16] Hujainah F., Bakar R.B.U., Abdulgabber M.A.A., Zamli K., "Software requirements prioritization: a systematic literature review on significance, stakeholders, techniques and challenges".IEEE Access, vol. 6, pp. 71497-71523, 2018.

[17] Misaghian N. and Motameni H., "An approach for requirements prioritization based on tensor decomposition".Requirements Engineering, Vol. 23, pp. 169-188, 2018.

[18] Sadiq M. and Jain S. K., "Applying fuzzy preference relation for requirements prioritization in goal-oriented requirements elicitation process".International Journal of Systems Assurance Engineering and Management, Vol. 5, Issue 4, pp. 711-723, 2014.

[19] Mellado D., Blanco C., Sánchez L. E., and Fernández-Medina E., "A systematic review of security requirements engineering".Computer Standards and Interfaces, Vol. 32, pp. 153-165, 2010.

[20] Salini P. and Kanmani S., "Survey and analysis on

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

Security Requirements Engineering".Computers and Electrical Engineering, Vol. 38, pp. 1785-1797, 2012.

[21] Ramesh M. R. R. and Reddy C. S., "A survey on security requirements elicitation methods: classification, merits, and demerits".International Journal of Applied Engineering Research, Vol. 11, Issue 1,pp. 64-70, 2016.

[22] Gupta D. and Jaiswal S., "Security engineering methods-in- depth analysis". International Journal of Information and Computer Security, Vol. 9, Issue 3, pp. 180-211, 2017.

[23] Nixon B. A., "Dealing with performance requirements during the development of information systems".IEEE International Symposium on Requirements Engineering, pp. 42-49, 1993.

[24] Yamada S. and Osaki S., "Cost-reliability optimal release policies for software systems". IEEE Transactions on Reliability, Vol. R-34, No. 5, pp. 422-424, 1985.

[25] Haley C. B., Laney R., Moffett J. D., and Nuseibeh B., "Security requirements engineering: a framework for representation and analysis".IEEE Transactions on Software Engineering, Vol. 34, No. 1, 2008.

[26] Khan B. I., Olanrewaju R. F., Anwar F., Mir R. N., Yaacob M., "Scrutinising internet banking security solutions".International Journal of Information and Computer Security, Vol.12 No.2/3, pp. 269-302, 2020.

[27] Thakkar A., Patel K., "VIKAS: a new virtual keyboard-based simple and efficient text CAPTCHA verification scheme".International Journal of Information and Computer Security, Vol.12 No.1, pp. 90-105, 2020.

[28] Gürses S., BerendtB. andSantenT.,"Multilateral security requirements analysis for preserving privacy in ubiquitous environments". 2006.

[29] Gürses S. and Santen. T., "Contextualizing Security Goals: A Method for Multilateral Security Requirements Elicitation". Sicherheit ,2006.

[30] Mead N., Hough E., Stehny T., "Security quality requirements engineering (SQUARE) methodology". Carnegie Mellon Software Engineering Institute, Technical report CMU/SEI- 2005- TR-009, 2005.

[31] Sindre G., Opdahl A.L., "Eliciting security requirements with misuse cases".Requirements Engineering, Vol. 10, pp. 34–44, 2005

[32] Lodderstedt T., Basin D., and J¨urgen D., "SecureUML: A UML-based modeling language for model-driven security".International Conference on the Unified Modeling Language, pp. 426–441, 2002.

[33] Jan J¨urjens,. "Towards Development of Secure Systems Using UMLsec". LNCS 2029, pp 187–200, 2001.

[34] LamsweerdeA. V., "Goal-oriented requirements engineering: a guided tour". Proceedings Fifth IEEE International Symposium on Requirements Engineering, Toronto, pp. 249-262, 2001.

[35] Lamsweerde, A.V.," Engineering requirements for system reliability and security". NATO Secur. Through Sci. Ser. D-Inf. Commun. Secur. Vol9, 2007.

[36] Mouratidis H., Giorgini P., "Secure tropos: a security-oriented extension of the tropos methodology". International Journal Software Engineering Knowledge Engineering, Vol.17, Issue 2, pp. 285–309, 2007.

[37] Anto`n A. I., Earp J. B., "Strategies for developing policies and requirements for secure electronic commerce systems". Department of Computer Science, North Carolina State University. Technical report TR-2000-09, 2000.

[38] Jackson M., "Analyzing and structuring software development problems". Addison Wesley, 2001.

[39] LinL.,NuseibehB.,InceD.,JacksonM., "Usingabuseframes to bound the scope of security problems". In: Proceedings of 11th IEEE international requirements engineering conference(RE'04), pp.354–355,.2004

[40] Hatebur D., Heisel M., Schmidt H., "Security engineering usingproblemframes".In:Mu¨llerG(ed)Proceedingsofthe international conference on emerging trends in information and communication security (ETRICS'06), ser. LNCS 3995. Springer, pp-238–253., 2006.

[41] Hatebur D., Heisel M., Schmidt H., "A pattern system for securityrequirementsengineering".In:Proceedingsoftheinte rnational conference on availability, reliability and security (AReS). IEEE Computer Society, pp 356–365, 2007.

[42] Mayer N., Rifaut A., Dubois E., "Towards a Risk-Based Security Requirements Engineering Framework". 2005.

[43] Braber F, Hogganvik I, Lund MS, Stølen K, and Vraalsen F, "Model-based security analysis in seven steps—a guided tourtotheCORASmethod".BTTechnolJ, pp. 101–117, 2007

[44] Asnar Y., Giorgini P., Massacci F., Zannone N."Fromtrustto dependability through risk analysis". In: Proceedings of the international conference on availability, reliability and security (AReS). IEEE Computer Society, 19–26, 2007.

[45] YuE.S.K.,LiuL., modeling trust for system designusing the i* strategic actors framework". In: Proceedings of the workshop on deception, fraud, and trust in agent societies held during the autonomous agents conference. Springer, London, pp 175–194, 2001.

[46] ISO/IEC_JTC1/SC27, Information technology Security techniques Evaluation criteria for IT security, ISO/IEC 15408:2005 (Common Criteria v3.0). 2005.

[47] Mellado D., Fernandez-Medina E., Piattini M., "Applying a security requirements engineering process". In:ESORICS'06,pp192-206 2006

[48] Al-Sarayreh K. T., Abran A., and Cuadrado-Gallego J. J., "A standards-based model of system maintainability requirements".Journal of Software: Evolution and Process, pp. 1-47, 2012.

[49] Lauesen S. and Younessi H., "Six styles for usability requirements". Proceedings of REFSQ'98, pp. 1-12, 1998.

[50] Jokela T., Koivumaa J., Pirkola J., Salminen P., and Kantola N., "Methods for quantitative usability requirements: a case study on the development of the user interface of a mobile phone". Personal and Ubiquitous Computing, Vol. 10, pp. 345-355, 2006.

[51] ISO/IEC, 9241-11 Ergonomic requirements for office work with visual display terminals(VDT), Part 11 Guidance on usability. ISO/IEC 9241-11 :1998 (E)

[52] Gu¨rses S. and Santen T., "Contextualizing security goals—a method for multilateral security requirements elicitation". In: Dittmann J (ed) Proceedings of SicherheitSchutz und Zuverla¨ssigkeit, ser. Lecture notes

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

in Informatics. Gesellschaftfu¨rInformatik, pp 42–53, 2006.

[53] Mayer N., Rifaut A., Dubois E., “Towards a risk-based security requirement engineering framework”International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ’05), in conjunction with the 17th conference on advanced information systems engineering, pp. 1-15, 2005.

[54] Sadiq M., “Selection of goal with incomplete preference relations,International Journal of Business Information System”. pp. 1-18, 2020.

[55] Faßbender S., Heisel M., and Meis R., “Problem-based security requirements elicitation and refinement with PresSuRE”.International Conference on Software Technologies, pp. 311-330, 2015.

[56] Saeki M., Hayashi S and Kaiya H. (2013), “Enhancing goal-oriented security requirements analysis using common criteria-based knowledge”.International Journal of Software Engineering and Knowledge Engineering, Vol. 23, No. 5 (2013) 695-720, 2013.

[57] Saeki M. and Kaiya H., “Security requirements elicitation using method weaving and common criteria”.MODELS Workshop, pp. 185-196, 2009.

[58] Fuchs A., Rieke R., “Identification of Security Requirements in Systems of Systems by Functional Security Analysis”.Architecting Dependable Systems VII, 2010.

[59] Ahmed N. and Matulevičius R., “Presentation and Validation of Method for Security Requirements Elicitation from Business Processes”. International Conference on Advanced Information Systems Engineering, pp. 20-35, 2014.

[60] Mead N. R., Miyazaki S., Zhan J., “Integrating privacy requirements considerations into a security requirement engineering method and tool”. International Journal of Information, Privacy, and Security, Vol. 1, Issue 1, 2011.

[61] Houmb S. H., Islam S., Knauss E. Jan Ju¨rjens J., Schneider K., “Eliciting security requirements and tracing them to design: an integration of Common Criteria, heuristics, and UMLsec”. Requirements Engineering, Vol. 53, pp. 63-93, 2010.

[62] Raspotnig C. and Opdahl A. , “Comparing risk identification techniques for safety and security requirements”. The Journal of Systems and Software, The Journal of Systems and Software, Vol. 86, pp. 1124–1151, 2013.

[63] Ionita D., Bullee J. W., and Wieringa R. J., “Argumentation-based security requirements elicitation: the next round”. ESPRE, pp. 7-12, 2014.

[64] Suleiman H. and Svetinovic D., “Evaluating the effectiveness of the security quality requirements engineering (SQUARE) method: a case study using smart grid advanced metering infrastructure”. Requirements Engineering, Vol. 18, pp. 251-279, 2013.

[65] Elahi G., Yu E., and Zannone N., “A vulnerability centric requirements engineering framework: analyzing security attacks, countermeasures, and requirements based on vulnerabilities”.Requirements Engineering, Vol. 15, pp. 41-62, 2010.

[66] Ansari M. T. A., Pandey D., and Alenezi M, “STORE: Security Threat Oriented Requirements Engineering Methodology”. Journal of King Saud University – Computer and Information Sciences, pp. 1-13, 2018.

[67] El-Hadary H. and El-Kassas S., “Capturing security requirements for software systems”. Cairo University Journal of Advanced Research, Vol. 5, pp. 463-472, 2014.

[68] Mouratidis H. and Jurjens J., “From goal-driven security requirements engineering to secure design”, International Journal of Intelligent Systems, 2010.

[69] Riaz M., Slankas J., King J., Williams L., “Using templates to elicit implied security requirements from functional requirements - a controlled experiment”. ESEM, pp. 1-10, 2014.

[70] Rudolph M., Feth D., Doerr J., Spilker J., “Requirements elicitation and derivation of security policy templates”. IEEE 24th International Requirements Engineering Conference, pp. 283-292, 2016.

[71] Ikram N., Siddiqui S., Khan N. F., “Security requirement elicitation techniques: the comparison of misuse cases and issue-based information systems”.EmpiRE 2014, pp. 36-43, 2014.

[72] Opdahl A. L. and Sindre G., “Experimental comparison of attack trees and misuse cases for security threat identification”.Information and Software Technology, Vol. 51, pp. 916-932, 2009.

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731

AUTHORS' BIOGRAPHY

Javed Ahmad is pursuing Ph.D. in Computer Science and Technology from the Department of Applied Sciences and Humanities, Faculty of Engineering and Technology, Jamia Millia Islamia (A Central University) New Delhi, India. He received M.Tech. degree in Computer Science in 2012 from Hamdard University, New Delhi, India; and B. Tech. in Computer Science and Engineering in 2008 from Shobhit Institute of Engineering and Technology affiliated to Uttar Pradesh Technical University, Lucknow, India. He has been working as a visiting faculty since 2008 in Computer Engineering Section, University Polytechnic, Faculty of Engineering and Technology, JamiaMilliaIslamia, New Delhi. He has qualified National Eligibility Test in December 2018, June 2019, and December 2019; and Graduate Aptitude Test in Engineering organised by Indian Institute of Technology Madras, India in 2019. He has published research papers in the following reputed conferences and journals: IEEE International Conference on Computer Research and Development, Malaysia; IEEE International Conference on Computing, Communication, and Intelligent Systems, India; International Journal of Engineering and Technology. His area of interest includes Software Engineering, Computer Networks, and Information Security.

Chaudhary Wali Mohammad received B. Sc. (Hons.), M. Sc., M. Phil., and Ph. D., all degrees in Mathematics, from the Department of Mathematics, Faculty of Natural Science, Aligarh Muslim University, Aligarh, U.P., India, in the year, 1978, 1980, 1983, and 1986, respectively. After completion of the Ph.D. degree in Mathematics, Prof. Chaudhary joined JamiaMilliaIslamia (JMI), New Delhi, on October 1, 1986, as a Lecturer (Contract) in Mathematics and become permanent faculty as a Lecturer (Mathematics) on February 9, 1989, in Faculty of Engineering and Technology, JMI, New Delhi. Prof. Chaudhary got promoted as a Reader (Mathematics) in 1999 and became full Professor on February 9, 2007. He worked as Head of the Department of Applied Sciences and Humanities, Faculty of Engineering and Technology, JMI, New Delhi since December 5, 2014 to December 4, 2017. Prof. Chaudhary has supervised three Ph.D. Thesis in the area of Multiple Hypergeometric Functions. His area of interest includes Multiple Hypergeometric Functions, Data Structure and Algorithms, Graph Theory, Multi-Criteria Decision Making (MCDM) algorithms and its application to Software Engineering. After teaching different papers of B. Tech/M. Tech courses, i.e., Numerical Analysis and Computer Programming (NACP), Numerical and Scientific Computing, Data Structure and Computer Programming (DCSP), Prof. Chaudhary gets motivated to work in the area of Computer Science and Technology. Currently, he is supervising four Ph.D. Thesis in the area of Computer Science and Technology; and one in the area of Applied Mathematics.

Mohd.Sadiq is working as a Post-Doctoral Fellow at the Department of Computer Science and Automation, Indian Institute of Science (IISc) Bangalore, India. He is on study leave from JamiaMilliaIslamia (JMI), A Central University, New Delhi, India, for his post-doctoral research at IISc Bangalore, India. In 2017, he received Ph.D. degree in Computer Engineering from National Institute of Technology, Kurukshetra, India. Prior to joining IISc Bangalore, he has worked as a Post-Doctoral Fellow at the Center for Soft Computing Research, Indian Statistical Institute (ISI), Kolkata, India; and also as a Visiting Scientist at the Systems Science and Informatics Unit, ISI, Bangalore Centre, India. He received M.Tech. degree in Computer Science and Engineering from Aligarh Muslim University, Aligarh, India, in 2005. He has delivered expert lectures on Artificial Intelligence and Compiler Design at Indian Institute of Information Technology Lucknow, India. Dr. Sadiq has published his research work in the following reputed journals: Business and Information Systems Engineering, Springer; IETE Journal of Research, Taylorand Francis; International Journal of System Assurance Engineering and Management, Springer; International Journal of Computers and Applications, Taylor and Francis; CSI Transactions on ICT, Springer; International Journal of Business Information Systems, Inderscience; and Conferences like RePrico@RE-2014, Sweden, SEKE-2013, USA, etc. His research interests lie at the intersection of software engineering and fuzzy /rough set-based multicriteria decision-making algorithms.

ISSN (Print): 2204-0595
ISSN (Online): 2203-1731