An Ontology Based Information Security Requirements Engineering Framework

Azeddine Chikh[1], Muhammad Abulaish[2,#], Syed Irfan Nabi[2,3] & Khaled Alghathbar[1,2]

[1]College of Computer and Information Sciences, King Saud University Riyadh, KSA
az_chikh@ksu.edu.sa
[2]Centre of Excellence in Information Assurance, King Saud University Riyadh, KSA
{mAbulaish, syedIrfan, kAlghathbar}@ksu.edu.sa
[3]Faculty of Computer Science, Institute of Business Administration, Karachi, Pakistan

Abstract. Software Requirement Specification (SRS) is frequently evolving to reflect requirements change during project development. Therefore, it needs enhancement to facilitate its authoring and reuse. This paper proposes a framework for building a part of SRS related to information security requirements (ISRs) using ontologies. Such a framework allows ensuring ISRs traceability and reuse. The framework uses three kinds of generic ontologies as a solution to this problem – software requirement ontology, application domain ontology, information security ontology. We propose to enhance SRS by associating the ISR with specific entities within ontologies. We aim to facilitate a semantic-based interpretation of ISRs by restricting their interpretation through the three previous ontologies. Semantic form is used to improve our ability to create, manage, and maintain ISRs. We anticipate that the proposed framework would be very helpful for requirements engineers to create and understand the ISRs.

Keywords: Information security, software requirements engineering, Software requirements specification.

1 Introduction

Due to increasing popularity and development of domain-specific ontologies there is an increasing effort of research dedicated to applying ontologies in software engineering and in its subset software requirements engineering (SRE) [1,2,5-7]. But, much of the research efforts have concentrated upon ontologies representing requirements models in general and a little effort has been made to address specific areas such as information security requirements (ISRs) engineering. Since it is such an important sub-area for many modern information systems and involves a complex set of concepts of its own, we see this as a shortcoming and the present research work, which could be considered as another labor of this trend, aims to apply ontologies to eliciting and managing ISRs. Authors in [8], believe that information security (IS) poses challenges to SRE that exceed those posed by other non-functional

[#] *To whom correspondence should be made. On leave from Jamia Millia Islamia (A Central University), New Delhi, India.*

requirements, and so they elevate it to be a research hotspot. In order to tackle these challenges we advocate the idea that ontologies can be used to annotate ISRs content, thus providing them with semantics.

In recent past, a small number of research works have been directed towards using ontologies in ISR engineering. In [5], Asheras *et al.* have proposed an ontology-based framework for representing and reusing security requirements based on risk analysis. A risk analysis ontology and a requirement ontology are developed and combined. It aims to represent reusable security requirements formally. Similarly, Lee *et al.* [18] have presented a novel technique from SRE and knowledge engineering for systematically extracting, modeling, and analyzing ISRs and related concepts from multiple enforced regulatory documents. They apply a methodology to build problem domain ontology from regulatory documents enforced by the Department of Defense Information Technology Security Certification and Accreditation Process.

In this paper, we propose the design of an ontology-based Information Security Requirements engineering framework which supports analysts in building and managing their ISRs. The proposed framework allows analysts to reuse existing IS knowledge in building new ISRs. The fundamental challenge for our framework is the management of ISR knowledge. While IS ontologies and requirements ontologies already exist, to the best of our knowledge, no methods have been proposed to map existing knowledge and best-practices guidelines on ISR to those existing ontologies.

The remainder of the paper is organized as follows. Section 2 presents a brief overview of SRE. It also presents the role of ontologies and IS in SRE. In section 3, we present the ISRs framework that integrates IS, SR, and application domain ontologies to annotate the domain knowledge resources for building new ISRs. Finally, section 4 draw conclusions and suggest future perspectives.

2 Software Requirements Engineering

SRE is a sub-category of requirements engineering that deals with the elicitation, analysis, specification, and validation of requirements for software [10] and it is critical for successful software development. SR are growing in importance as a means for the customer to know in advance what solution he/she will get.

The software requirement specification (SRS) is an official statement of what the system developers should implement. It should include both the user requirements for a system and a detailed specification of the system requirements [9]. A different understanding of the concepts involved may lead to an ambiguous, incomplete specification and major rework after system implementation [1]. Accordingly, it is important to assure that all analysts and stakeholders in the analysis phase have a shared understanding of the application domain. Even when users can express their needs, analysts find it difficult to write them accurately. The result is that the real demands and the written requirements don't match [9]. The nature of SRE involves capturing knowledge from many sources. Ontologies can be used for both, to describe requirements specification [8,9] and formally represent requirements content. Ontologies seem to be well suited for an evolutionary approach to the specification of requirements and domain knowledge [4]. In addition, ontologies can be used to support requirements management and traceability [3].

2.1 Software Requirements Engineering Ontology

The SRE ontology we have selected to be part of our framework has been proposed by [7] and named SWORE - SoftWiki Ontology for Requirements Engineering. SoftWiki (Distributed, End-user Centered Requirements Engineering for Evolutionary Software Development) is to support the collaboration of all stakeholders in software development processes in particular with respect to SR. Potentially very large and spatially distributed user groups shall be enabled to collect, semantically enrich, classify and aggregate SR. The rationale is to provide a semantic structure, which will capture requirements relevant information and enables interlinking of this information with domain and application specific vocabularies.

Figure 1 visualizes the core of the SWORE ontology, which was developed in accordance with standards of the requirements engineering community [11]. Central to this ontology are the classes – Stakeholder and Abstract Requirement along with property details. Abstract requirements have the subclasses – Goal, Scenario, and Requirement each of which are defined by stakeholders and can be detailed by other abstract requirements. This enables the specification of abstract requirements at different levels of granularity. The collaborative aspects of requirements engineering are emphasized by integrating discussions amongst the stakeholders and voting in the model. In the requirement engineering process this documentation is often relevant for future decisions. [7].

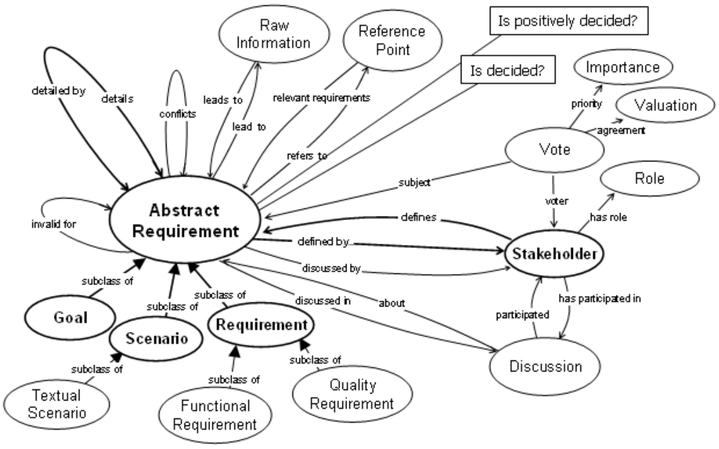

Figure 1. Visualisation of the SWORE Ontology Core [7]

2.2 Information Security in Software Requirements Engineering

Information security requirements (ISRs) include the types and levels of protection necessary for equipment, data, information, applications, and facilities to meet security policy. Specifically, ISR are identified by risk analysis – the systematic use of information to identify sources and to estimate the risk. Risk analysis is one of the three sources identified by the security standard ISO 27002 (Code of Practice for IS Management) to identify ISRs [12]. The other two sources are related to the legal, regulatory and contractual requirements of an organization and to the principles, objectives and business requirements for information processing that an organization has developed to support its operations [5].

Navigating the large number of existing dedicated standards for IS for building a new ISRs present a challenge to costumers. Some authors typically find, in reviewing requirements documents, that ISRs, when they exist, are likely to be incomplete or are in a section by themselves and have been copied from a generic list of security features. The requirements elicitation and analysis that are needed to get a better set of ISRs seldom take place. A systematic approach to security requirements engineering will help to avoid the problem of generic lists of features and to take into account the attacker perspective [13]. A number of authors highlighted the needs of an ontology for a security community [14]. Authors in [15] ask the following research question: “To what extent can the IS domain knowledge, including concepts and relations which are required by common IS risk management methodologies, be modeled formally? Which source can be used to enrich the knowledge model with concrete and widely accepted IS knowledge?”

2.3 Ontologies in Information Security

The security ontology we have selected to be part of our framework has been proposed by [15] and based on the security relationship model described in the National Institute of Standards and Technology Special Publication 800-12 [16]. Figure 2 shows the high-level concepts and corresponding relations of this ontology. A threat gives rise to follow-up threats, represents a potential danger to the organization's assets and affects specific security attributes (e.g. confidentiality, integrity, and/or availability) as soon as it exploits a vulnerability in the form of a physical, technical, or administrative weakness, and it causes damage to certain assets. Additionally, each threat is described by potential threat origins (human or natural origin) and threat sources (accidental or deliberate source). For each vulnerability a severity value and the asset on which the vulnerability could be exploited is assigned. Controls have to be implemented to mitigate an identified vulnerability and to protect the respective assets by preventive, corrective, deterrent, recovery, or detective measures (control type). Each control is implemented as asset concept, or as combinations thereof. Controls are derived from and correspond to best-practice and IS standard controls [15].

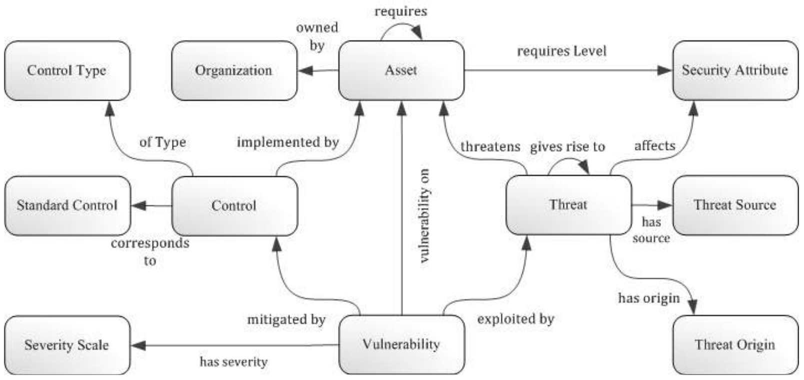

Figure 2. A sample security ontology proposed in [15]

3 Proposed Ontology Based ISRs Engineering Framework

In this section, we discuss the proposed ontology-based information security requirements engineering framework that can present to the readers (end-users, stakeholders, analysts, designers and developers) an integrated view of the knowledge and best practices related to ISRs within a given software development project. Indeed, the readers of the user requirements, such as end-users, are not habitually concerned with how the system will be implemented. But, the readers of the system requirements, such as designers and developers, need to know more precisely what the system will do because they are concerned with how it will support the domain user tasks. The proposed ontology-based ISRs engineering framework is based on the organization of knowledge in three complementary domains – application, SRE, and IS. The functional details of the major components of the framework are explained as following:

Domain Ontologies containing domain-related concepts and relationships in a structured and machine-interpretable format are used to annotate domain resources. Corresponding to the three above-mentioned domains, we have considered three different ontologies – Software Requirements Ontology (SRO), Information Security Ontology (ISO), and Application Domain Ontology (ADO). We advocate the idea that ontologies can be used to describe ISRs, thus providing them with a new dimension of content reusability. These ontologies are further discussed in the following paragraphs:

SRO encompasses the whole set of SRE concepts. It covers many possibilities – requirements on various levels from goal-level to design-level, different requirements styles from plain texts to diagrams, and from data requirements to quality requirements, many techniques and methods from elicitation and analysis to validation and verification [17]. We have selected as SRO the SWORE ontology (SoftWiki Ontology for Requirements Engineering) in [7].

ISO provides the semantic concepts based on some IS standard such as ISO/IEC_JTC1, and their relationships to other concepts, defined in a subset of the IS domain. We have selected as ISO the security ontology proposed by Fenz & Ekelhart [15] and based on the security relationship model described in the National Institute of Standards and Technology Special Publication 800-12 [16].

ADO involves understanding the application domain (library, human resources, finances, sales, etc.). In order to enable effective ISRs understanding we have to further enhance semantics of their content. Therefore, we recommend that they should be further enhanced by providing application domain ontology based annotations of their content. Many specific application domain ontologies exist on the Web that can be found using the Swoogle1 – a semantic web search engine.

Knowledge Resources correspond to the three above-mentioned domains – SR resources, IS resources, and application domain resources. These resources represent every document or reference useful in the corresponding domain. This includes theoretical as well as practical knowledge (best practices) in the domain. The framework allows indexing, using and reusing of knowledge resources in different software development projects, based on concepts from the former ontologies.

Semantic Annotator : Annotation (also called tagging) is a process that associates names, attributes, comments, or descriptions to a document or to a selected part in it [19]. It provides additional information (metadata) about an existing piece of data. Compared to tagging, which speeds up searching and helps you find relevant and precise information, Semantic Annotation enriches the unstructured or semi-structured data with a context that is further linked to the structured knowledge of a domain and it allows results that are not explicitly related to the original search. Thus, semantic annotation adds diversity and richness to the search process. Semantic Annotation helps to bridge the ambiguity of the natural language when expressing notions and their computational representation in a formal language. By telling a computer how data items are related and how these relations can be evaluated automatically, it becomes possible to process complex filter and search operations. For this purpose, the semantic annotator module exploits the concepts and relationships stored in ontologies and annotate the resources with them. Thus, this module outputs annotated knowledge resources in which relevant components are tagged with ontological concepts.

Semantic Query Processor makes the proposed framework capable to process users (requirements engineers) queries over annotated knowledge resources. The query processor has an ontology-guided query interface that helps users to formulate queries using ontological concepts at different levels of specificities. The output of the query interface sub-module is a semantic query which is passed to the query processing engine to get relevant documents from annotated knowledge resources.

Requirement Authoring takes the retrieved knowledge resources by semantic query processor as input and analyze (reuse) them to build new requirements. The new requirements are present to the users (requirements engineers) and also added in the software requirements resources for future use.

[1] http://swoogle.umbc.edu/

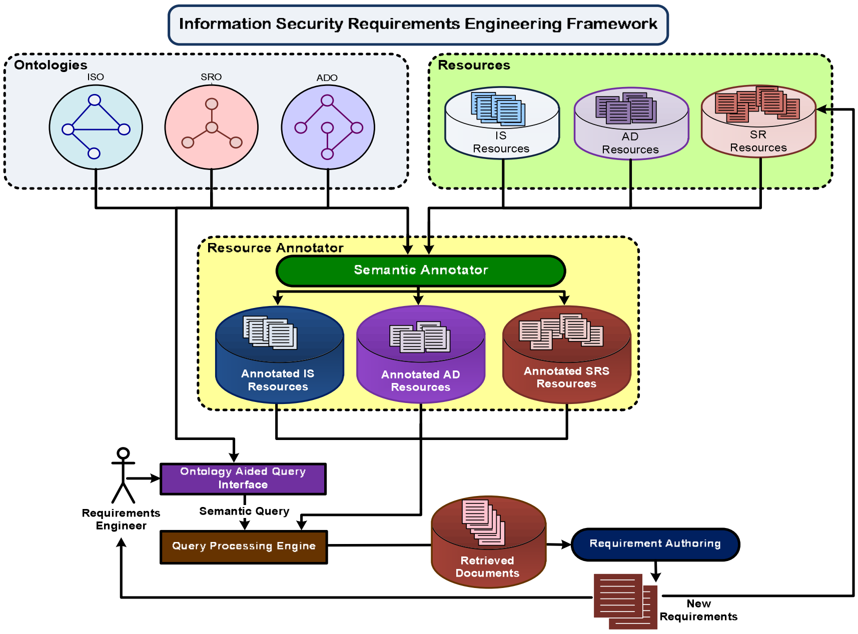

Figure 3. Ontology-based Information Security Requirements engineering framework

4 Conclusion

In this paper, we have presented an ontology-based information security requirements engineering framework which can facilitates the requirements engineers to create and understand the information security requirements after analyzing domain resources annotated with ontological concepts. The novelty of the proposed framework to unify software requirements, information security, and application domain ontologies to annotate domain knowledge resources. The annotated resources are then analyzed by the semantic query processor to identify new requirements. Presently, we are developing a prototype of the proposed framework to analyze its effectiveness for real-life applications.

References

1. Happel H. J., Seedorf S.: Applications of Ontologies in Software Engineering, in: Proceedings of the International Workshop on Semantic Web Enabled Software Engineering (SWESE), 2006.
2. Decker B., Rech J., Ras E., Klein B., Hoecht C.: Self Organized Reuse of Software Engineering Knowledge Supported by Semantic Wikis, in: Proceedings of the Workshop on Semantic Web Enabled Software Engineering (SWESE), Nov. 2005.

3. Ayank V., Kositsyna N., Austin M.: Requirements Engineering and the Semantic Web, Representation, Management, and Validation of Requirements and System-Level Architectures. Technical Report, Part II, TR 2004-14, University of Maryland, 2004.
4. Wouters B., Deridder D., Van Paesschen E.: The Use of Ontologies as a Backbone for Use Case Management, in: Proceedings of the European Conference on Object-Oriented Programming (ECOOP), Workshop : Objects and Classifications, a natural convergence, 2000.
5. Asheras J., Valencia-García R., Fernández-Breis J. T. and Toval A.: Modelling Reusable Security Requirements based on an Ontology Framework, Journal of Research and Practice in Information Technology, Vol. 41, No. 2, May 2009.
6. Kaiya H., Saeki M.: Using Domain Ontology as Domain Knowledge for Requirements Elicitation, in: Proceedings of the IEEE International Requirement Engineering Conference, 2006, pp. 186–195.
7. Yanwu Y., Xia F., Zhang W., Xiao Xian, Li Y., Li X.: Towards Semantic Requirement Engineering, Semantic Computing and Systems, IEEE International Workshop on Semantic Computing and Systems, 2008, pp. 67-71.
8. Cheng B. H. C., Atlee J. M.: Research Directions in Requirements Engineering. Future of Software Engineering (FOSE), in ICSE. Minneapolis, Minnesota, IEEE Computer Society, 2007, pp. 285-303.
9. Sommerville I., Software Engineering, Pearson Education, 2011.
10. Bourque P., Dupuis R. (eds.) Guide to the Software Engineering Body of Knowledge , IEEE Computer Society, 2004.
11. Pohl K., Requirements Engineering - Grundlagen, Prinzipien, Techniken. Dpunkt Verlag, 2007.
12. ISO27002, ISO/IEC 17799-27002 Code of Practice for Information Security Management, 2005.
13. Mead N. R.: Security Requirements Engineering, https://buildsecurityin.us-cert.gov/bsi/articles/best-practices/requirements/243-BSI.html, 2006.
14. Tsoumas B., Gritzalis D.: Towards an Ontology based Security Management, in: Proceedings of the 20th International Conference on Advanced Information Networking and Applications, IEEE Computer Society, 2006.
15. Fenz S., Ekelhart A.: Formalizing information security knowledge, in: Proceedings of the ACM Symposium on Information, Computer and Communications Security, 2009.
16. IST. An Introduction to Computer Security – The NIST Handbook. Technical report, NIST (National Institute of Standards and Technology), October 1995. Special Publication 800-12.
17. Lauesen S.: Software Requirements - Styles and Techniques, Addison-Wesley, 2002.
18. Lee S-W., Gandhi R., Muthurajan D., Yavagal D., Ahn G-J.: Building Problem Domain Ontology from Security Requirements in Regulatory Documents, in: Proceedings of the International Workshop on Software Engineering for Secure Systems, 2006.
19. Popov B., Kiryakov A., Ognyanoff D., Manov D., Kirilov, A.: KIM – A Semantic Platform for Information Extraction and Retrieval, Journal of Natural Language Engineering, Vol. 10, Issue 3-4, Cambridge University Press, 2004, pp. 375-392.