Knowledge Base for an Intelligent System in order to Identify Security Requirements for Government Agencies Software Projects

Beltrán G. Adán[1,a] , Lombana C. Cristhian[1], Calvo L. Mario[1], Ordoñez S. Sonia[1],Caviativa C. Yaneth,[1] and Garcés Jairo[1]

[1] *Grupo de Investigación en Ingenieria de Software. Vicerrectoría de Investigaciones. Universidad Manuela Beltrán. Av. Circunvalar No. 60-00, Bogotá, Colombia*

Abstract. *It has been evidenced that one of the most common causes in the failure of software security is the lack of identification and specification of requirements for information security, it is an activity with an insufficient importance in the software development or software acquisition We propose the knowledge base of CIBERREQ. CIBERREQ is an intelligent knowledge-based system used for the identification and specification of security requirements in the software development cycle or in the software acquisition. CIBERREQ receives functional software requirements written in natural language and produces non-functional security requirements through a semi-automatic process of risk management. The knowledge base built is formed by an ontology developed collaboratively by experts in information security. In this process has been identified six types of assets: electronic data, physical data, hardware, software, person and service; as well as six types of risk: competitive disadvantage, loss of credibility, economic risks, strategic risks, operational risks and legal sanctions. In addition there are defined 95 vulnerabilities, 24 threats, 230 controls, and 515 associations between concepts. Additionally, automatic expansion was used with Wikipedia for the asset types Software and Hardware, obtaining 7125 and 5894 software and hardware subtypes respectively, achieving thereby an improvement of 10% in the identification of the information assets candidates, one of the most important phases of the proposed system.*

1 Introduction

It has been shown that the most common causes of application security vulnerabilities are the incomplete identification of requirements and bad specification of requirements. In Colombia, the government entities have been subject of several information security incidents. The root cause of those incidents has been identified as a bad requirements engineering practice. A simple study of the Request For Proposals (RFP) used to contract software development and software acquisition written by government entities, shows that security requirements are underspecified. Most of the documents ask for a "secure implementation" or a "secure configuration", but they do not describe in detail the concrete aspects of such request.

In this article we describe the knowledge base of an intelligent system for the identification and specification of security requirements in software applications. The knowledge base was designed using elements of semantic web, natural language processing, knowledge management and cross sourcing. The purpose of the intelligent system is to allow government entities to identify and define, together with the software provider, the security requirements that application and systems must meet.

The article is organized as follows, section II presents the state of the art regarding knowledge bases developed for cybersecurity. Section III discusses the methodology used to build the knowledge base. In section IV we describe the ontology used to build the knowledge base for the proposed system and shows how the system interacts with the knowledge base.

[a] Corresponding author: adan.beltran@docentes.umb.edu.co

2 State of the Art: Knowledge Bases for Cybersecurity

In [1] an ontology of information security is proposed. The ontology includes the most relevant concepts of the domain as Asset, Vulnerability, Threat and Control. It also discriminates between tangible and intangible assets; and allows modeling of the physical infrastructure of the organization with information such as the place where the asset is located. The ontology has 500 concepts and 600 formal restrictions, and was derived from best practice guidelines and standards for information security, including Internet Security Glossary (RFC 2828); German IT Grundschutz Manual; The United Nations Standard Products and Services Code; National Institute of Standards and Technology Special Publication 800-12; ISO / IEC 27000; among others.

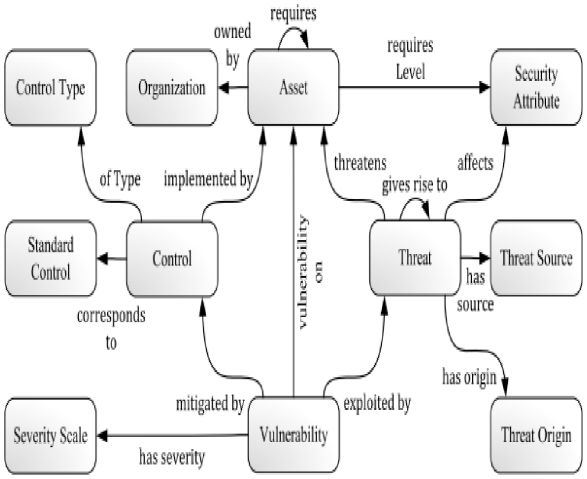

Figure 1. Security relationships [1].

In [2] the authors present a framework composed of several ontologies. These ontologies are used to represent, store and reuse safety requirements. The first ontology presents knowledge for risk analysis following the ISO 27002 standard (see Fig. 2) As a result the ontology identifies five main elements: assets, threats associated with the asset, protection measures to address threats, valuation dimensions (attributes that make an asset valuable) and valuation criteria (measure of the importance of an asset to the organization).

The second ontology has classified requirements according to IEEE standards. In the combination of these two ontologies, each security requirement has an associated asset together with threats and protection measures. Additionally, each requirement has the information of valuation dimensions, and valuation criteria for each asset associated with the requirement

In [3] an ontology of security incidents is proposed. The ontology describes a conceptual framework with the following elements: Agent, Attack, Security Incident, Tools Vulnerability and Access. These elements are related as follows: an agent performs an Attack that can cause a Security Incident. In order to perform an Attack,the agent uses a Tool, which exploits a Vulnerability, in order to get Access. The Incident has a Consequence, on an Asset, and happens at a specific Time.

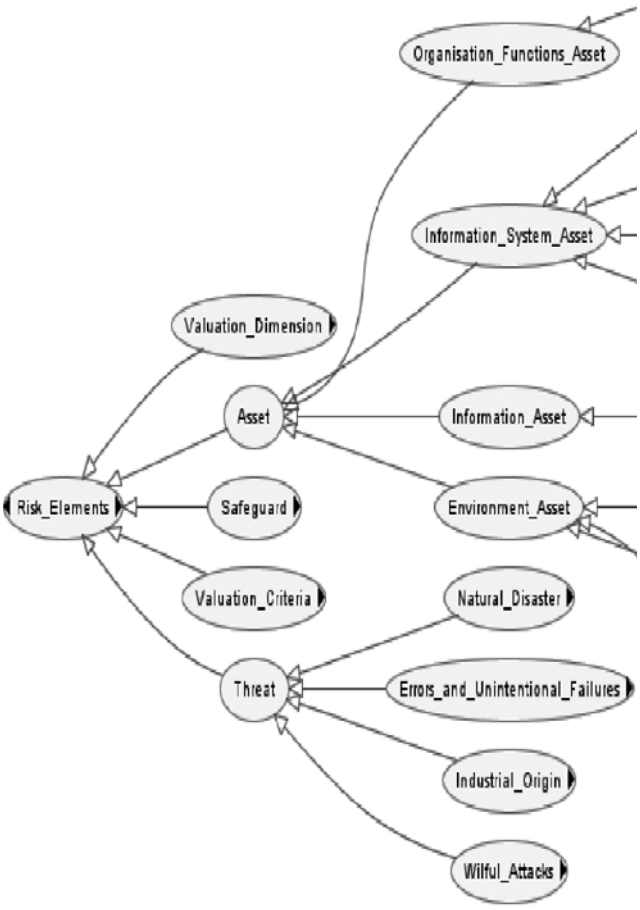

Figure 2. Taxonomy of the elements of the risk analysis ontology [2].

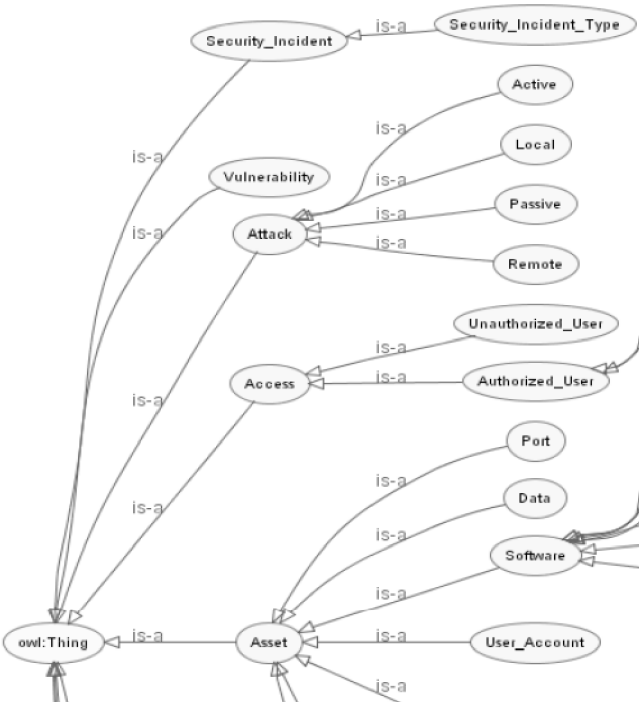

Figure 3. The security incident ontology conceptual model [3].

In [4] an initial work is presented for a unified security ontology. First, the study identifies the basic requirements that the ontology should have.

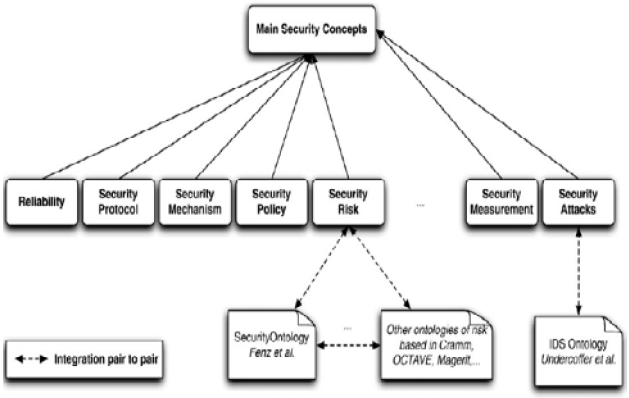

Figure 4. Overlapping between security domains for integrated security ontology [4].

To identify such requirements the study uses OntoMetric to create a comparative analysis of existing proposals. These requirements are: static knowledge, dynamic knowledge and reusability. Second, a process for ontology integration was applied. Following such process overlapping areas between ontologies were identified (see Fig. 4), related concepts and consistency of the result were verified.

3 Design Methodology for the Ontology

For the design of ontology the methodology described in [5] was used. The following steps are carried out.

A. What is the domain the ontology will cover?

Definitions of functional requirements to acquire or develop an application.

B. For what we use the ontology?

To identify information security requirements.

C. What types of questions the information in the ontology should answer?

- What are the vulnerabilities for an asset?
- What are the threats that can exploit a vulnerability?
- What are the controls that can minimize a vulnerability?
- What are the types of risk that may materialize?

a. Enumerate the important concepts in the domain

Information asset, environment, threat, vulnerability, inherent risk, impact, control, residual risk, accepted risk, encryption algorithm, encryption software, cryptographic hash algorithm, cryptographic-summary software, log audit, log traceability, application, application code, database, table, communications link, issuer, receiver, owner of information asset, switch, router, firewall, antimalware, malware.

b. Define the classes and the class hierarchy

Information asset: information; hardware; software; person, service.

Threats: human intentional (STRIDE), human unintentional, hardware, nature, among others.

Vulnerability: unencrypted data, unsigned data, absence of audit, lack of traceability, lack of access control, data without cryptographic summary, absence of capture fields validation, absence of output data validation, among others.

Risk: damage or loss of assets, excessive costs, loss of income, loss of business, image loss, legal sanctions, wrong decisions, impact, level of involvement for the company can be measured in money, percentage levels, among others.

Controls: rules, procedures, policies, implementation guidelines, standards, cryptography, logs, monitoring, access control, disciplinary sanctions, backups, alternate center, redundancy, among others.

4 Results

A. CyberSecurity Ontology

This section describes the ontology that was designed following the methodology described above (see Fig. 5).

We now define in detail the classes (concepts) and properties (relations) of the ontology.

1) Classes

Asset type: Electronic Data, Physical data, Hardware, Software, Service, Person.

Vulnerability: Unrestricted access, Absence of antivirus, Absence of backup, Absence of logging and auditing, Weak passwords, Disgruntled employee, Typos, Lack of secure deletion policy, Installing unauthorized software, Unprotected network point, Vulnerabilities in the operating system and/or PC applications, among others.

Threat: Intrusive access to the PC, Alteration or removal, Accidental damage, Natural disasters, Terrorism or public disorder, Information leakage, Infection or malware, among others.

Control: Enable audit logs, Apply Active Directory policies, User training, Defining roles and user roles, Record file deletion, Implementing encryption in storage, Implement UPS or power plant, Implement firewall, Backups policy, Secure wiring policy, Procedure for defining strong passwords, Record of failed attempts to access network resources, among others.

Risk type: Competitive disadvantage, Loss of image or credibility, Economic risks, Strategic risks, Operational risks, Legal sanctions.

2) Properties

Can have: the relation *can have* is defined as follows: an Asset Type can have a Vulnerability.

Exploits. The relation *exploits* has Threat as domain and a conjunction between Asset Type and Vulnerability as range.

Minimize. In the relation *minimize* the domain is Control and Vulnerability is the range. A control minimizes a vulnerability.

Can be materialized. The relation *can be materialized* has Risk Type as domain and Asset Type as range.

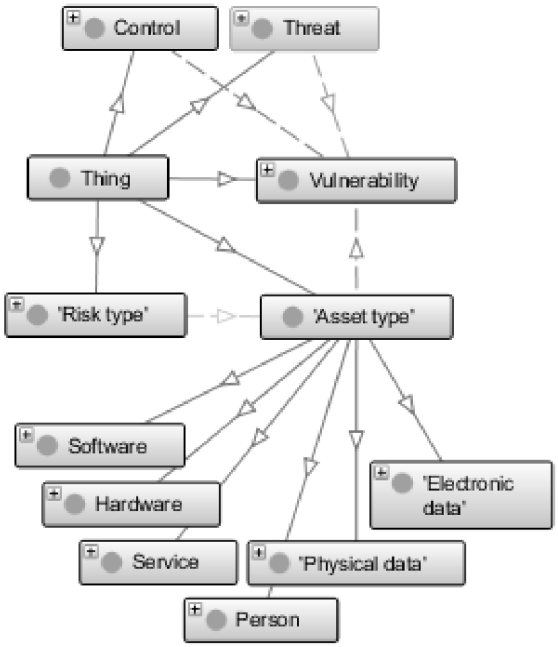

Figure 5. Top level concepts and relationships in the ontology.

A concrete example of the type of information that can be extracted from the ontology is:

The asset type 'Electronic data' can have a vulnerability of 'Absence of backup', which is exploited by the threat 'Alteration or deletion' and is minimized by the control 'Backups Policy'. The risk that can be materialized in this case is 'Operational risk'.

The ontology has 361 concepts, which are broken down into six types of assets, six types of risks, 95 vulnerabilities, 24 threats and 230 controls. In addition, there are 515 relationships between concepts.

The following link shows the complete ontology description that was developed using Protégé [6]: http://webprotege.stanford.edu/#Edit:projectId=1530e338-671c-474f-a01e-4f77808ce63b

B. Knowledge-based Intelligent System

The intelligent system based on the ontology is called "CIBERREQ". CIBERREQ is a tool for the identification and specification of security requirements for projects of software development or software acquisition in government entities.

The system receives as input functional requirements written in natural language and, using the knowledge represented in the knowledge base, supports domain experts or users in the definition and specification of security requirements.

CIBERREQ uses the knowledge base for (Fig. 6):

- *Identify the information assets candidates*, found in the functional requirements.
- *Identify vulnerabilities* related to specific assets.
- *Identify threats* that exploit vulnerabilities.
- *Identify the controls* that minimize vulnerabilities.
- *Identify the types of risks* that may materialize.

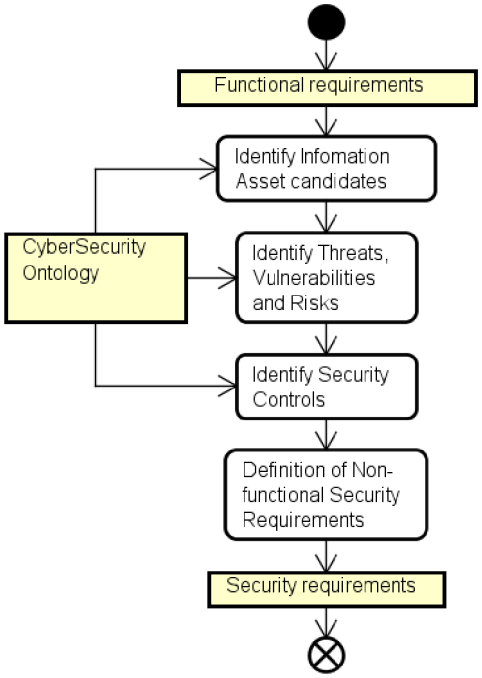

Figure 6. Process for identification of security requirements to CIBERREQ.

Following there is a description of an example of the application of the CIBERREQ tool in a real project:

Given the following functional requirement: "it is required a functionality that allows for the initial loading of parametric tables that make up the databases Base1 and BD2; the system identifies semi-automatically, using natural language processing techniques and validation from security experts, the following information assets: BD1 (unique client base), BD2 (membership database), parametric tables. These three correspond to the asset type 'Electronic Data'.

For these assets the tool identified the vulnerability: "password administration inadequate"; and the threat: "non-authorized access", and the control: "to implement strong passwords in the access control established for the equipment".

For the preceding information the following risks were identified: "damage or loss of assets due to non-authorized access in Base2 due to inadequate password administration"; "legal sanctions for non-authorized access in Base2 due to inadequate password administration".

Considering the previous information, the security expert, using the mentioned tool, defined the following non-functional security requirements: "confidential information that is processed and transmitted must be encoded with strong cryptographic algorithms"; "authorized users must enter the system using authentication based on the specific role"; "a profile, historical and tracing register must be generated".

Figure 7. CIBERREQ tool.

Figure 7 shows the process developed with the CIBERREQ tool for the case previously described.

C. Ontology expansion with Wikipedia

Additionally, with the object of enriching the ontology, some terms were automatically expanded using Wikipedia´s API.

In Wikipedia, every page has one or more associated categories, and each category can have subcategories or supercategories [7]. For this expansion, category trees were extracted of up to 7 depth levels for the asset types Software and Hardware, obtaining 7125 subtypes of Software and 5894 subtypes of Hardware. This allowed an improvement in identifying the information assets candidates of 10%, one of the most important phases in the system.

5 Discussion

The use of ontologies in building the knowledge base facilitates maintenance, expansion and extension of the system to other more specific contexts of information security to improve accuracy throughout the process of identifying security requirements.

Therefore, this knowledge base can be used for other projects that are not government.

6 Conclusion

We have presented the design of an ontology for information security, and a tool that uses the ontology in order to aid in the identification and specification of security requirements. The ontology was developed collaboratively by domain experts and users, and was later expanded automatically with Wikipedia, producing a 10% improvement in the precision of the phase of information asset identification of the CIBERREQ tool. The resulting ontology is a general model that is not specific for a platform or technology. Thus it can be used to develop other intelligent knowledge-based systems.

The knowledge base was validated with different experts and officials from state agencies, in addition, the system was used in a real project of a state entity, but as previously said the knowledge base can be specified and adjusted for other not necessarily governmental contexts, and the fact that part of this ontology is based and expanded with terms in Wikipedia allows validation by the expert community in the subject.

Acknowledgment

This research is financed jointly by Colciencias (Fondo Francisco Jose de Caldas), Ministry of Information and Communication Technologies of Colombia (MinTic), Universidad Manuela Beltrán and Ubiquando – Project No. 1263-668-44906

References

[1] S. Fenz and A. Ekelhart. "Formalizing Information Security Knowledge." In *Proceedings of the 4th International Symposium on Information, Computer, and Communications Security*, 2009, 183-194.

[2] J. Lasheras, R. Valencia-García, J. T. Fernández-Breis, and A. Toval. "Modelling reusable security requirements based on an ontology framework."

Journal of Research and Practice in Information Technology, vol. 41, no. 2, pp. 119–133, 2009.

[3] A. Martimiano and E. Moreira. "An owl-based security incident ontology." in *Proceedings of the Eighth International Protege Conference*, 2005, pp. 43–44.

[4] C. Blanco, J. Lasheras, E. Fernández-Medina, R. Valencia-García, and A. Toval, "Basis for an integrated security ontology according to a systematic review of existing proposals," *Computer Standards & Interfaces*, vol. 33, no. 4, pp. 372–388, 2011.

[5] N.F. Noy and D.L. McGuinness. "Ontology Development 101: A Guide to Creating Your First Ontology". Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March 2001.

[6] Protégé. http://protege.stanford.edu/. [Oct. 01, 2015]

[7] T. Zesch, I. Gurevych and M. Mühlhäuser. "Analyzing and Accessing Wikipedia as a Lexical Semantic Resource." *Data Structures for Linguistic Resources and Applications*, pp. 197-205, 2007.