A Security Ontology for Security Requirements Elicitation

Amina Souag, Camille Salinesi, Raul Mazo, Isabelle Comyn-Wattiau

A Security Ontology for Security Requirements Elicitation

Amina Souag[1], Camille Salinesi[1], Raúl Mazo[1], and Isabelle Comyn-Wattiau[2]

[1] CRI -Paris 1 Sorbonne University
Paris, France
{amina.souag,camille.salinesi,raul.mazo}@univ-paris1.fr
[2] CEDRIC-CNAM & ESSEC Business School
Paris, France
isabelle.wattiau@cnam.fr

Abstract. Security is an important issue that needs to be taken into account at all stages of information system development, including early requirements elicitation. Early analysis of security makes it possible to predict threats and their impacts and define adequate security requirements before the system is in place. Security requirements are difficult to elicit, analyze, and manage. The fact that analysts' knowledge about security is often tacit makes the task of security requirements elicitation even harder. Ontologies are known for being a good way to formalize knowledge. Ontologies, in particular, have been proved useful to support reusability. Requirements engineering based on predefined ontologies can make the job of requirement engineering much easier and faster. However, this very much depends on the quality of the ontology that is used. Some security ontologies for security requirements have been proposed in the literature. None of them stands out as complete. This paper presents a core and generic security ontology for security requirements engineering. Its core and generic status is attained thanks to its coverage of wide and high-level security concepts and relationships. We implemented the ontology and developed an interactive environment to facilitate the use of the ontology during the security requirements engineering process. The proposed security ontology was evaluated by checking its validity and completeness compared to other ontologies. Moreover, a controlled experiment with end-users was performed to evaluate its usability.

Keywords: Security, ontology, concepts, security requirements, elicitation.

1 Introduction

Security has moved from being considered by Information Systems (IS) designers as a technical topic to becoming a critical issue in our society [1]. With the growing digitization of activities, IS are getting more and more complex. They must comply with new usages, varied needs, and are permanently exposed to new vulnerabilities. There is no single week without an announcement indicating that the IS of some

F. Piessens et al. (Eds.): ESSoS 2015, LNCS 8978, pp. 157–177, 2015.

private or public organization was attacked. The cost of cybercrime in 2012 reached $110B in the world [2]. It has been reported recently that attacks to sensitive data increased by 62% in 2013 with 253 incidents observed and 552 million identities stolen [44]. A major obstacle that faces analysts, and requirements engineers, is the fact that knowledge about security is most often tacit, imprecisely defined and non-formalized. Among the challenges for security projects is the difficulty of expressing security requirements and producing exhaustive specifications. A requirement prescribes a condition judged necessary for the system [3]. Security Requirements Engineering (SRE) methods derive security requirements using specific concepts, borrowed from security engineering paradigms [4]. It is well known that ontologies are useful for representing and inter-relating many types of knowledge of a same domain. Thus, the research community of information system security [5] urged the necessity of having a good security ontology to harmonize the vaguely defined terminology, leading to communication troubles between stakeholders. The benefits of such a security ontology would be manifold: it would help requirements engineers reporting incidents more effectively, reusing security requirements of the same domain and discussing issues together, for instance [6]. Several research studies have addressed the issue of knowledge for the field of security [7][8]. The research presented in this paper is part of a larger ongoing research project that aims at proposing a method that exploits ontologies for security requirements engineering [9]. In [9], a small security ontology was first used for the elicitation and analysis of security requirements. Being "small", the ontology used affected the resulting requirements and the whole security requirements analysis process. In a previous research, several security ontologies were compared and classified [7]. The paper concluded that ontologies are good sources for security requirements engineering. However the quality of the resulting security requirements depends greatly on the ontologies used during the elicitation and analysis process. To cope with the aforementioned issues, this paper proposes a core security ontology that considers the descriptions of the most important concepts related to security requirements and the relationships among them. "*Core*" refers to the union of knowledge (high-level concepts, relationships, attributes) present in other security ontologies proposed in the literature. As Massacci et al. claims, "Although there have been several proposals for modeling security features, what is still missing are models that focus on high-level security concerns without forcing designers to immediately get down to security mechanisms"[15]. Meta-models can be useful since they provide an abstract syntax of security concepts. However, we believe that ontologies can be a better option since they allow representing, accessing, using and inferring about that knowledge in order to develop methods, techniques, and tools for security requirements analysis. According to [16], a good security ontology should inter alia include static knowledge (concepts, relationships and attributes), and dynamic knowledge (axioms). It must be reusable (commented in natural language, and formalized in a standard language). The main objective of this paper is to address the following research questions: *What are the concepts and relations that need to be present in a core security ontology? And how to make this ontology easy for requirements engineers to use?* This ontology should make it possible to: (a) Create a generic platform of different security concepts (threats, risks, requirements, etc.). (b) Create a source of reusable knowledge for the elicitation of security requirements in various projects.

The rest of the paper is organized as follows: Section 2 presents the construction of the ontology, its concepts and relationships. Section 3 reports the evaluation of the proposed ontology. Related works are presented in Section 4 through a literature review. Finally, Section 5 concludes the paper and describes future work directions.

2 A Core Security Ontology for Security Requirements Engineering

This section presents the main contribution of this paper, a core security ontology to be used particularly for security requirements elicitation process. The method for constructing the security ontology is adapted from ontology construction methods proposed by Fernandez [25], mixed with key principles of the ones proposed by Jones et al. [26]. The construction process contains six main steps: objective, scope, knowledge acquisition, conceptualization, implementation, and validation. The objective behind the ontology construction must be defined in the beginning, including its intended uses, scenarios of use, end-users, etc. The scope stipulates the field covered by the ontology. The knowledge acquisition step aims at gathering from different sources the knowledge needed for the ontology construction. In the step of conceptualization, the knowledge is structured in a conceptual model that contains concepts and relationships between them. Ontology implementation requires the use of a software environment such as Protégé[1]; this includes codifying the ontology in a formal language (RDF or OWL/XML). Finally, the validation step guarantees that the resulting ontology corresponds to what it is supposed to represent. The details about how the first five steps were applied to construct our ontology are presented in the following sub-sections and the last step is detailed in Section 3.

2.1 Objective

The main objective of the target ontology is to provide a generic platform containing knowledge about the core concepts related to security (threats, vulnerabilities, countermeasures, requirements, etc.). This ontology will be a support for the elicitation of security requirements and the development of SRE methods and tools. The ontology will be a meta-view for the different security ontologies in the literature. It should harmonize the security terminology spread in these ontologies and help requirements engineers communicate with each other.

2.2 Scope of the Ontology

The ontology covers the security domain in its high level aspects (threats and treatments) as well as its organizational ones (security procedures, security management process, assets, and persons). The reader will find details on all security concepts covered by the ontology in section 2.4.below on.

[1] `http://protege.stanford.edu/`

2.3 Knowledge Acquisition

The acquisition of the security knowledge started from standards (e.g. ISO27000). Other knowledge acquisition sources were the different security ontologies that exist in the literature. We analyzed about 20 security ontologies, based on previous literature surveys; the full list of these ontologies can be found in [7] and [8]. These ontologies are of various levels (general, specific, for a particular domain). Relevant concepts and relationships were extracted through a systematic and syntactic analysis of the security ontologies (their concepts and relations). Table 8 in the appendix presents part of them (13 ontologies). For the sake of space, we cannot provide the reader with the description of all the ontologies used as a source of knowledge for the ontology. Brief descriptions of some of them are presented in the following:

- The ISSRM model [27] (top left in Fig. 1.) was defined after a survey of the risk management standards, security related standards, and security management methods. The three groups of concepts proposed in the ISSRM model (asset related concepts, risk related concepts, and risk treatment related concepts) were used to define the three dimensions of the ontology (organization, risk, treatment).
- Fenz et al. [24] have proposed an ontology to model the information security domain. We reused some concepts and relationships of that ontology, in particular the ones related to the infrastructure of organizations (assets, organization), the relationships between threats and assets, and between threats and vulnerabilities. We also reused some standard controls used in Fenz's ontology to define our security requirements.
- Lashras et al.'s security requirements ontology [12] was useful to define the security requirements in our ontology.

Fig. 1 schematizes the knowledge acquisition step and part of the conceptualization phase, starting with the knowledge sources (the different ontologies), the concept

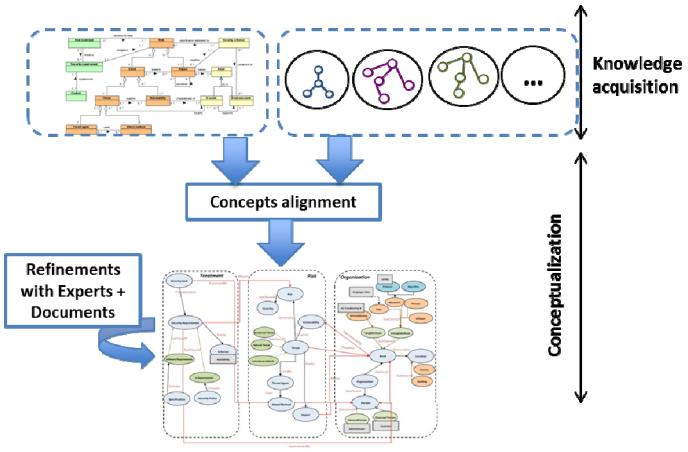


Fig. 1. Knowledge acquisition and conceptualization phases

alignment, and the conceptualization with the help of experts and documents. The concepts of the resulting ontology were derived from the alignments of the different security ontologies in the knowledge acquisition step. The knowledge and the conceptualization steps were performed manually relying essentially on tables to align the concepts and relations of the different source ontologies.

2.4 Conceptualization

Based on the outcomes of the knowledge acquisition step, concepts were organized and structured in a glossary. Various relationships among these concepts were considered, and then were put together in a conceptual model of the ontology (Fig. 4 in the appendix), easy to understand, independently of any implementation language. The names of the concepts and the relationships of the security ontology proposed in this paper were chosen according to the number of occurrences of names in the source ontologies (Table 8 in the appendix). If a concept has different names in the ontologies (e.g. impact or consequence, attack method or deliberate attack, or SessionIP attack); the most generic or easiest to understand name was chosen (here, impact, attack method). Some security experts (5 experts) were consulted to validate the choices that were made. The validation was informal and took the form of email exchanges, phone and direct discussions. The experts acknowledged most of the concepts and relationships between them. Some refinements in the ontology were performed after discussion with them. For example, the concept of "Attack" was removed, since the experts consider it as an Intentional Threat. Discussions also clarified the difference between the concepts of "Security Goal", "Security Criterion", "Security Requirement" and "Control". These concepts are frequently mixed up in the security requirements elicitation phase and the difference between them is often not easy to capture. The concepts were organized around three main dimensions. The latter are: Risk dimension, Treatment dimension, and Organization dimension. In ontology engineering terms [45]: the Risk, Treatment and Organization dimensions are considered as modules. The Risk dimension represents the "dark" face of security; it gathers concepts related to threats, vulnerabilities, attacks, and threat agents. Treatment dimension is concerned with concepts related to the necessary treatments to overcome risks. The concepts are security goals, requirements, controls, and security policies. Finally, security is a multifaceted problem; it is not only about technical solutions or single assets, but also about the environment where threats appear and arise. That is why the Organization dimension is considered. This dimension relates to concepts such as person, location, assets, and organization that must be analyzed and on which assumptions must be match in a security requirements elicitation process. Some ontologies covered only the dimension treatment [12]. The security ontology proposed by Fenz et al. [24] groups concepts into three sets (security, enterprise and location). The classification into these three dimensions (organization, risk and treatment) helps in organizing the knowledge related to security; it has been inspired by the security meta-model proposed in [27]. The concepts and relationships of the ontology are described in the following sub-section. To visualize the different concepts and relations, the reader may refer to Fig.4.

1) Concepts of the Security Ontology

The following summarizes the different concepts identified for the ontology with their respective descriptions. These general concepts together with their relations constitute the ontology, which presents an overview of the information security in a context-independent manner. In the following, we describe the concepts dimension by dimension.

a) Organization dimension: This dimension includes the concepts related to the organization, its assets and its environment. The concepts are:
Organization: a structure including human, hardware, and software resources (assets).
Person: Represents human agents. A person may be internal in the organization (e.g., administrator) or external (e.g., customer).
Asset: a valuable resource, which can be a tangible asset (e.g., air-conditioning, fire extinguisher, computers) or an intangible asset. Intangible assets can be, for example, software, data, and industrial manufacturing processes.
Location: Defines the asset's location. Location can be a brick and mortar physical location such as a classroom, data center or office. It can also consist of collaborative research materials on a file share or financial information stored in a database [28].

b) Risk dimension: The concepts of the risk dimension are:
Risk: a combination of a vulnerability and threat causing harm to one or more asset.
Severity: the level of risk, e.g. high, medium or low.
Threat: a violation of a security criterion. The threat may be natural, accidental, or intentional (attack).
Vulnerability: a weakness of an asset or group of assets that can be exploited by one or more threats [29] (e.g., weak password).
Impact: the impact may vary from a simple loss of availability to loss of the entire information system control. Impact can also be of other types such as harm to the image of the company.
Threat agent: the person (or program) who carries out the threat. The name 'threat agent' was chosen to cover both types of threat, either intentional (carried out by an attacker) or unintentional (carried out by any person, not necessarily an attacker).
Attack method: Refers to the different methods used by threat agents to accomplish their attacks, such as sniffing (which lets threat agents capture and analyze traffic transmitted over a network); spoofing (where the threat agent attempts to impersonate someone or something else); and social engineering (tricking people into giving sensitive information or performing actions on behalf of the threat agent).
Attack tool: The tool used to perform the attack, e.g. sniffing tool (e.g., Wireshark[2]), spoofing tool (e.g. Subterfuge[3]), scan port tool (e.g., Nmap[4]) and others.

[2] http://www.wireshark.org/
[3] http://code.google.com/p/subterfuge/downloads/list
[4] http://nmap.org/

c) Treatment dimension:
Security goal: a security goal defines what a stakeholder/organization hopes to achieve in the future in terms of security [27], it states the intention to counter threats and satisfy security criteria. Security goals are sometimes considered as security objectives [47].
Security Requirement: a condition defined on the environment that needs to be fulfilled in order to achieve a security goal and mitigate a risk. Depending on what we want to protect and on the target security level, we define our requirements. They can be related to databases, applications, systems, organizations, and external environments. For example, "the system shall ensure that personal data can be accessed only by authorized users" and "the system shall deliver data in a manner that prevents further or second hand use by unauthorized people".
Control: a means or a way to secure assets and enable a security requirement, e.g., alarm or password.
Security criterion: defines security properties such as confidentiality, integrity, availability, and traceability. It can also be considered as a constraint on assets.
Requirements document: The document that states in writing the necessary security requirements to protect the assets. Two main documents generally contain security requirements:

- Security policy: a security policy expresses the defense strategy or strategic directions of the information security board of an organization.
- Specification document: it gathers the set of requirements to be satisfied by a material, design, product, or service. The document contains, inter alia, security requirements.

2) Relationships of the Security Ontology

High-level relationships between those concepts were defined. They were categorized into four kinds: IsA, HasA, SubClassOf and AssociatedTo. The relationships between the concepts of the security ontology can be briefly described as follows: An organization has assets (Has_Asset). An asset may have a location (Has_Location). Tangible and intangible assets are subclasses of the asset concept (SubClassOf). An organization also includes persons that it deals with (Has_Person). The persons can be internal or external (SubClassOf). An asset is threatened by one or many threats (Threatens). These threats exploit vulnerabilities in the assets (Exploits). The threat-agent leads an attack (LeadBy) and uses attack methods (UseMethod) or attack tools (UseTool) to achieve an attack. A threat implies an impact (Implies), for example: "A denial of service attack implies a server downtime". The impact affects one or more assets (Affect). A threat can be natural, intentional, or accidental (SubClassOf). A threat generates a risk (Generate) with a certain level of severity (HasSeverity). Security requirements mitigate a risk (Mitigate) and satisfy (Satisfy) security goals expressed by stakeholders (ExpressedBy). Security requirements fulfill (Fulfills) one or more security criteria. For instance, the requirement "The application shall ensure that each user will be able to execute actions for which he/she has permission at any time/every week" satisfies the security criteria Confidentiality and Availability. Controls enable a security requirement (Enable). For example, the control "password"

enables the requirement "The application shall ensure that each user will be able to execute actions for which he/she has permission". Security policies and specifications incorporate (Includes) security requirements; these may either be security software requirements (SubClass), which relate to the security of applications or databases, or security organizational requirements (SubClass), which relate to assets, persons, or buildings.

3) Attributes and Axioms of the Security Ontology

In addition to concepts and relationships, an ontology contains axioms and attributes. Formal axioms are assertions accepted as true about abstractions of a field. The axioms allow us to define the meaning of concepts, put restrictions on the values of attributes, examine the conformity of specified information, or derive new concepts [30]. As stated before, the ontology proposed in this paper was not created from scratch. It was constructed by reusing knowledge of existing security ontologies. In particular, some attributes (see Table 1) of the ontology proposed by [31] were reused. For instance, a person has a phone number (its type is Integer); a requirements document has a version (its type is String).

Table 1. Part of the table of attributes

Concept	Attribute	Value type
Person	Phone number	Integer
Software	Version	String
Requirement Document	Version	String
Password	Minimum length	Varchar

The ontology proposed by [24] was a good source of axioms. Table 2 illustrates some axioms with their descriptions and the related concepts.

Table 2. Part of the table of axioms

Description	Expression	Concepts
A threat can be either intentional or accidental	$\forall x: Threat \Rightarrow Intentional\ Threat\ (x) \vee Natural\ Threat\ (x) \vee Accidental\ Threat\ (x)$	Threat
A requirements document can be either a policy or a specification	$\forall x: Requirements\ Document \Rightarrow Security\ Policy\ (x) \vee Specification\ (x)$	Requirements document Security policy Specification

Fig. 4 in the appendix presents the security ontology proposed in this paper. It includes the three dimensions, including concepts and relationships.

2.5 Implementation of the Ontology

Among the different editors of ontologies (OntoEdit [32], Ontolingua [33] and Protégé [34]). Protégé (version 3.4.8) was chosen since it is an extensible, platform-independent environment for creating, editing, viewing, checking constraints, and extracting ontologies and knowledge bases. Ontologies via Protégé can be developed in a variety of formats. OWL 1.0 (Web Ontology Language) was used for the development of the ontology as recommended by the World Wide Web Consortium (W3C). To test and extract relevant knowledge from the security ontology, SQWRL (Semantic Query-Enhanced Web Rule Language) was used. SQWRL is a SWRL-based (Semantic Web Rule Language) for querying OWL ontologies. The description of SQWRL syntax is beyond the scope of the paper; readers may refer to O'Connor et al. [35] for further details. Some indicative queries are presented later in the next section. Implementing the core security ontology with OWL and Protégé is not enough. The target end-users are requirements engineers who are asked to elicit security requirements for different projects, on which they have a tacit knowledge. The ontology will provide the necessary security knowledge in a formalized and explicit form. It also makes available a set of reusable security requirements. To make it usable even for end users not familiarized with Protégé and SQWRL, an interactive environment based on Eclipse was developed. Fig. 2 illustrates the architecture of the tool. The interactive environment facilitates the exploration of the ontology. It automatically and dynamically generates the necessary. SQWRL queries and rules for obtaining the information related to assets, organization, threats, vulnerabilities, and security requirements. The interactive environment makes it possible to generate a specification (a Word document) that summarizes the result of the analysis. Protégé plays the role of the engine; it is intended to wait for SQWRL queries (it plays a passive role in the communication with the end user). Once a query is received,

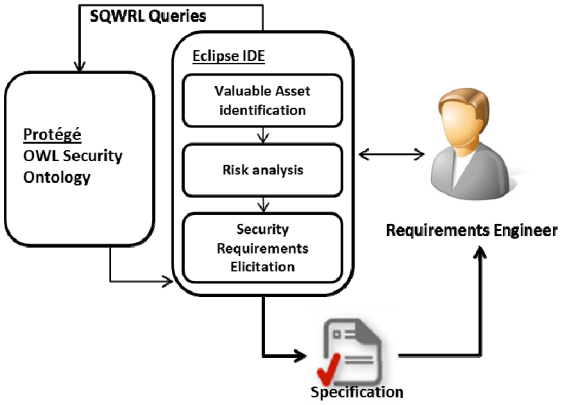


Fig. 2. Architecture of the interactive environment

Protégé processes it and then sends the result to the interactive environment. With this architecture, Protégé is opaque to the requirements engineers; i.e., the requirements engineers do not interact directly with it.

A screenshot of the user interface is presented in Fig. 3. In particular, this figure presents part of the interface; a typical security requirements analysis process was performed, with 3 main windows: valuable asset identification (on the left side), risk analysis, and security requirements elicitation (on the right side).

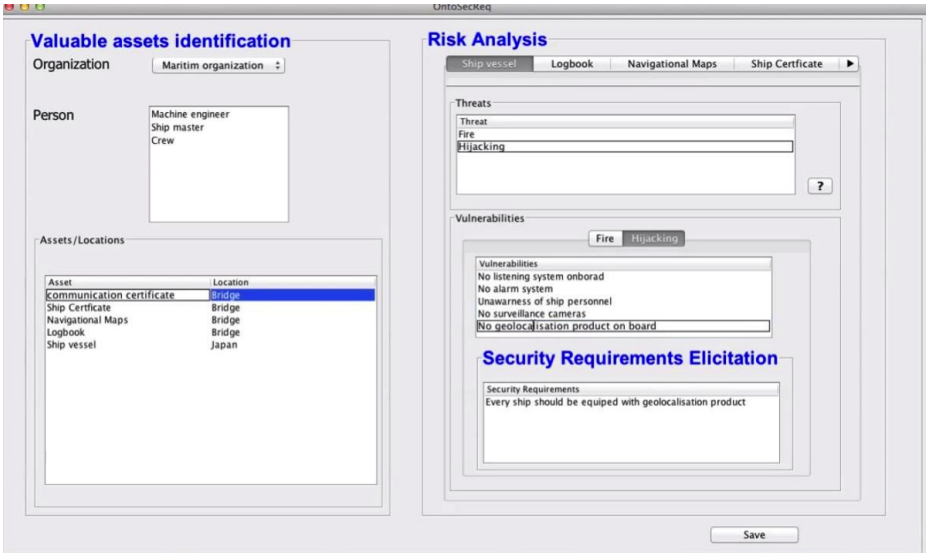


Fig. 3. A screenshot of the interactive environment[5]

The interactive environment allows the user to choose the organization. It displays the persons involved and the list of all assets with their corresponding locations. It also allows the user to choose valuable assets that he/she wants to protect. The latter are displayed on the left. For each asset the environment displays the corresponding threats (threat agents, impact and generated risk of each threat). For each chosen threat, the environment displays the corresponding vulnerabilities. And finally, for all chosen vulnerabilities, the resulting list of security requirements to mitigate them is presented. The "Save" button leads to the generation of the specification document that summarizes the analysis and the relevant security requirements.

3 Evaluation

Given that our goal was to develop an ontology covering the high-level concepts of security, and make it (re)usable by the requirements engineering community, the focus was on the following criteria:

[5] A demonstration video can be viewed at: `http://youtu.be/zwGbe0Z_mTE`

- *Completeness*: this criterion will be evaluated by mapping the target ontology and some other ontologies extracted from literature. The focus was mainly on security ontologies that have been used in security requirements engineering [9][10][11][12].
- *Validity*: Through this criterion, the ability of the ontology to provide reliable answers to a set of questions using its terminology was checked.
- *Usability*: This criterion refers to 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"[6]. In our case, it demonstrates that the ontology can be used for security requirements elicitation, and reused through different projects.

3.1 Completeness

The completeness criterion verifies that our ontology integrates the knowledge that exists in the other ontologies. By completeness, we want to prove that the proposed ontology is 'more' complete than the ones covered by our literature. An alignment table was drawn up, with the concepts of our ontology on one side, and concepts of security ontologies found in security requirements engineering literature on the other side. Table 3 presents the result of the alignment.

Table 3. The alignment table of the proposed security ontology with ontologies used for security requirements elicitation

	Ontologies used for security requirements elicitation				
Concepts of the ontology	*Daramola et al.* [11]	*Ivankina et al.* [10]	*Lashras et al.* [12]	*Salini et al.* [13]	*Dritsas et al.* [14]
Asset	Asset	Asset	Asset	Asset	Asset
Location	-	-	-	-	-
Organization	-	-	-	-	-
Person	-	-	-	Stakeholder	Stakeholder
Threat	Threat/ Active attack	-	Threat	Threat	Threat/ Deliberate attack
Vulnerability	-	Threat causes	-	Vulnerability	Vulnerability
Risk	-	-	Risk	-	-
Severity	-	-	Valuation criteria	-	-
Impact	-	-	-	Impact severity	-
Threat agent	-	-	-	-	Attacker
Attack tool	-	-	-	-	-
Attack method	Code injection	-	-	-	-
Security goal	-	-	-	-	Objective
Security criterion	-	-	-	Security objective	Security requirement
Security requirement	-	Treatment	Security requirement	Security requirement	-
Control	-	-	Safe guard	-	Countermeasure

[6] According to: ISO 9241-11.

Most of the security ontologies used in the SRE contain the concept of "Asset". Given that security issues affect all the infrastructure of organizations, other concepts were introduced (with their corresponding sub-classes): Location, Organization and Person. While many of the other security ontologies take into consideration the concept Threat, most of them neglect the concept Risk generated by a threat, and its Severity. Only the ontology proposed by Dritsas et al. [14] uses the concept of "Attacker". Only the ontology used by Daramola et al. [11] includes the concept of "Attack Method". Our proposed security ontology covers the concept "Objective" used by Dritsas et al. [13]. The concept "Security Criterion", missing in the security ontologies [11], [10] and [12] was used in [13] and [14]. Note that [14] considers as a 'security requirement' what other sources consider a 'security criterion' (availability, confidentiality ...). The concept "Security Requirement" was used in [10], [12] and [13]. These results tend to demonstrate that the proposed security ontology is complete with respect to the union of all the other security ontologies used in security requirements studies, since it incorporates all their concepts.

3.2 Validity

According to Uschold & Gruninger [36], informal and formal questions are one way to evaluate an ontology. The ontology must be able to give reliable answers to these questions using its terminology. The ontology was applied to the maritime domain. For now, the application for domain specific cases is done manually, by instantiating the concepts of the core ontology with domain concepts. Ongoing work is being carried to automatize this instantiation.

This section lists a number of questions that a requirements engineer is likely to encounter during the requirements elicitation phase of a development project. These questions should be regarded as indicative of what the ontology can deal with and reason about. Table 4 summarizes some of these questions. Each of the questions is expressed informally in natural language and formally using SQWRL. The answers to the questions are presented in the last column. These queries guide the requirements engineer during the security requirements elicitation process. The process includes: i) valuable assets identification (what are the assets of the organization? Where are they located? What are the persons involved in the organization?), ii) the risk analysis (what are the threats that threaten the asset? Who leads the attack? What is the attack method used?), and iii) security requirements elicitation (what are the security requirements to mitigate the risk? What are the controls needed to implement those security requirements? What are the security criteria that those requirements fulfill?)

Table 4. Informal and formal questions to the ontology

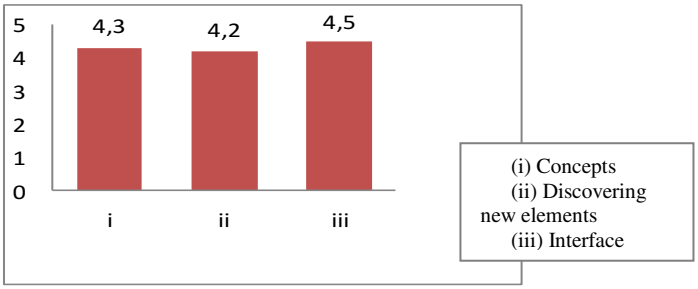
	Queries	Part of result
Valuable asset identification	What are the organizations in the scope of the project? `Organization(?o) → sqwrl:select(?o)`	Maritime organization X, Maritime organization Y
	What are the assets to be protected in the maritime organization X? What is the location of each asset? `Has_Asset(Maritime_organizationX,?a) • Has_Location(?a, ?l) → sqwrl:select(?a, ?l)`	Ship, Navigation maps located in the bridge
Risk analysis	What are threats that threaten the asset "ship"? `threatens(?T,Ship) → sqwrl:select(?T)`	Ship hijacking
	Who is responsible for the threat "Ship hijacking"? `LedBy(Ship_hijacking,?A)→sqwrl:select(?A)`	Hijacker
	What is the method used by the hijacker to attract the ship? `Threat(Ship_hijacking)•Uses(Hijacker,?M) → sqwrl:select (?M)`	Fake distress signal
	What are the impacts of such a threat on the ship? `Implies(Ship_hijacking,?I)→ sqwrl:select(?I)`	Theft of provision, Hostage
Security requirements elicitation	What are the security requirements to consider to mitigate the risk? `Exploits (Ship hijacking, V?) • mitigated_by(?V, ?R) →sqwrl:select(?r)`	**Req1.** Every Ship should be equipped with geolocalization products. **Req2**. Every Ship should be equipped with a listening system on board.

This section has demonstrated how the security ontology could be exploited in the security requirements elicitation phase. This can bring the necessary knowledge to the requirements engineers. This sub-section has illustrated one possible application in the maritime field.

3.3 Usability

To evaluate the usability of the core security ontology, a controlled experiment was performed with end users. The protocol of the experiment was adapted from experimental design and analysis methods [42][43]. In order to obtain a representative group of participants [37], we contacted by mail and phone people from security and requirements engineering communities (laboratories, associations, LinkedIn…). People (industrialists or researchers) not related to the field were intentionally excluded. We used the profile page, and the job position to include/exclude a participant. The day of the experiment, 10 participants were present. The average age was 30 years old. Three participants were certified ISO27000, and three had industrial experience with EBIOS [38] (a well-known French risk assessment method). Four were PhD students working on related subjects. The experiment included a presentation of the security ontology (its main concepts and relations), demonstration of the interactive environment, and a session of manipulation by the participants. At the end of the experiment, participants were asked to fill in a questionnaire[7]. The results extracted from these questionnaires are summed up in Tables 5, 6, 7.

Table 5. Average grading usability



First, the participants were asked to grade the usability of the ontology on a scale of 1 to 5 through three main questions: (i) *Do you find that the security ontology contains the main concepts for security requirements elicitation?* (ii) *Does the security ontology help in finding new elements (threats, vulnerabilities, security requirements, etc.)*? (iii) *Do you find the interface to access to security ontology easy to use?* The scale (1 to 5) corresponds to the degree of agreement to the asked question. Thus (5 = strongly agree, 4 = agree, 3 = neither agree nor disagree, 2= disagree, 1= strongly disagree). Table 5 (page 12) shows a quite high level of satisfaction, which is encouraging. Most participants find that the security ontology includes the main concepts. It helps in discovering new elements even for those who are experts in security since it is not easy to bear in mind hundreds of threats,

[7] The questionnaire can be consulted on: `https://www.dropbox.com/s/cc40n31p3fucf4o/Sec%20Ont%20Evaluation%20form.pdf?dl=0`

vulnerabilities, and their corresponding security requirements. Almost all participants liked the interactive environment, and revealed that is nice to have the code of the ontology (in OWL-Protégé) hidden. Among the positive qualitative feedbacks that were provided by participants: "*I find in the ontology all concepts that are used in risk analysis methods such as EBIOS*". One participant mentioned that: "*The ontology seems to have main concepts and individuals, however it would be nice to update it constantly, there are new threats appearing every day!*". That was an interesting point that could be improved in the future by providing a mechanism to update automatically the individuals of the security ontology.

The second series of questions were particularly devoted to the next stages of the research project and their answers constitute an important input for future work. The participants were asked: (iv) *Does the core security ontology help in building security models*? Secure Tropos models [39] were taken as an example of a security modeling framework. It was presented to participants who did not know it before.

Table 6. Does the core security ontology help in building Secure Tropos models?

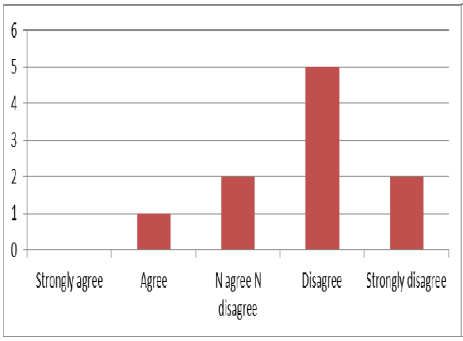


Table 7. Does the security ontology help in eliciting security requirements for other specific domains?

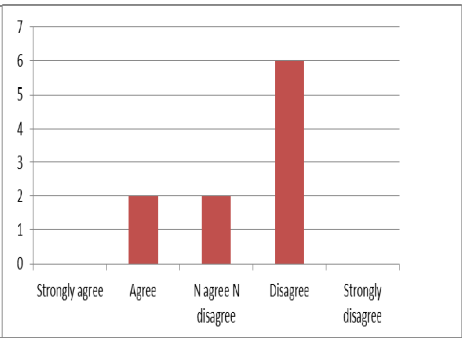


Table 6 reports the results for question (iv). Most participants find it difficult to pass from the concepts of the core security ontology to the concepts of Secure Tropos. A common answer was: "*We understand the existence of connections but the mapping from the core security ontology to Secure Tropos is not straightforward*". The discussion with participants that followed this question shows that, although the security ontology has the main concepts, relations and individuals, this is still not enough for users to build security models with it. More guidelines or mapping rules are necessary, not for the ontology itself but for the process of using it for security requirements elicitation. The last question was: (v) *does the security ontology help in eliciting security requirements for other specific domains (health, military, and bank)?* We wanted to know if the security ontology helps in providing more security domain specific knowledge each time one switches from a domain to another one. Table 7 reports results for question (v) and shows that most participants "disagree" on the fact that the security ontology by itself is sufficient for eliciting security

requirements for different specific domains. One participant mentioned "*something additional is required for the application to different specific domains*". The ontology can be used in different application contexts with some extra collaboration with domain experts, consulting documentation. On the current research phases, we are trying to make the process automatic by using the core security ontology with different domain ontologies.

4 Security Ontologies: Related Works

Considerable works have been devoted to knowledge in the field of security. Schumacher [46] proposed a security ontology and qualifies it as "Core Ontology". This ontology was a good beginning but omits organizational related concepts and some other key concepts such as attack method and attack tool, or security criteria and controls. Undercoffer et al. [17] propose an ontology that characterizes the domain of computer attacks and intrusions. The ontology covers concepts such as host, system component attack, input and consequence. Geneiatakis and Lambrinoudakis [18] propose an ontology for SIP-VoIP (Session Initial Protocol-VoIP) based services. Denker et al. [19][20] develop several ontologies for security annotations of agents and web services, using DAML (DARPA Agent Markup Language) and later OWL. Karyda et al. [21] present a security ontology for e-government applications. Tsoumas et al. [22] define a security ontology using OWL and propose the security framework of an information system which provides security acquisition and knowledge management. Herzog et al. [23] propose an ontology based on the following top-level concepts: assets, threats, vulnerabilities and countermeasures. Some approaches considered modeling security ontologies such as [48]. To our knowledge, ontologies of this kind come close to being meta-models, in that they are used more to share a common understanding of the structure of the modelling language than to enable reuse of knowledge. Fenz and Ekelhart [24] propose an ontology that targets a similar goal but attempts to cover a broader spectrum: their ontology models the information security domain, including non-core concepts such as the infrastructure of organizations. A large part of these studies deal with the development of low-level ontologies limited to a particular domain. A previous survey [7] classifies the existing security ontologies into eight main families: theoretical basis, security taxonomies, general, specific, risk based, web oriented, requirements related and modeling. The analysis of these ontologies reveals that they vary a lot in the way they cover security aspects as reported in previous work [7]. The results converge with those of Blanco et al. who conducted a systematic review of security ontologies [8].

5 Conclusion and Future Work

This paper presents a core ontology for the IS security requirements elicitation and analysis process. The completeness of this ontology was evaluated with regards to existing security ontologies used in security requirements engineering methods. An interactive environment was developed to facilitate its use and reuse. The controlled experiment demonstrated that the ontology helps requirements engineers in eliciting

security requirements by allowing them to exploit security-structured knowledge. This was made possible via the interactive environment that dynamically generates the necessary queries. Despite all this effort, the goal of constructing this kind of security ontologies remains ambitious and was found to be more complex than expected. One single team's work is not enough. This research should be of a more collaborative nature including many teams working on security ontologies. A truly complete security ontology remains a utopian goal. However, in the case of this proposed ontology, it can be improved by considering other sources related to security expertise (not mainly ontologies as was the case in this work). The controlled experiment could be performed with a larger number of participants to improve the validity of the results.

In future work, we plan to integrate the ontology and its reasoning features with existing approaches for security requirements analysis (Secure Tropos, KAOS, and others). We plan to make this security ontology more domains-specific by relying on domain ontologies. On the technical level, the plan is to keep the ontology up to date and perform the necessary migrations to the latest available versions (OWL/Protégé).

References

1. Denker, G., Kagal, L., Finin, T.: Security in the Semantic Web using OWL. Information Security Technical Report 10(1), 51–58 (2005)
2. Norton, 2012 Norton Cybercrime report (July 2012)
3. Kauppinen, M., Kujala, S., Aaltio, T., Lehtola, L.: Introducing requirements engineering: how to make a cultural change happen in practice. In: Proceedings IEEE Joint International Conference on Requirements Engineering (RE 2002), pp. 43–51 (2002)
4. Elahi, G., Yu, E., Li, T., Liu, L.: Security Requirements Engineering in the Wild: A Survey of Common Practices. In: Proceedings of COMPSAC 2011, pp. 314–319 (2011)
5. Donner, M.: Toward a Security Ontology. IEEE Security and Privacy 1(3), 6–7 (2003), http://dlib.computer.org/sp/books/sp2003/pdf/j3006.pdf
6. Souag, A.: Towards a new generation of security requirements definition methodology using ontologies. In: Proceedings of 24th International Conference on Advanced Information Systems Engineering (CAiSE 2012), Gdańsk, Poland, June 25-29, pp. 1–8 (2012)
7. Souag, A., Salinesi, C., Comyn-Wattiau, I.: Ontologies for Security Requirements: A Literature Survey and Classification. In: Bajec, M., Eder, J. (eds.) CAiSE Workshops 2012. LNBIP, vol. 112, pp. 61–69. Springer, Heidelberg (2012)
8. Blanco, C., Lasheras, J., Valencia-Garcia, R., Fernandez-Medina, E., Toval, A., Piattini, M.: A Systematic Review and Comparison of Security Ontologies. In: The Third International Conference on Availability, Reliability and Security, ARES 2008, pp. 813–820 (2008)
9. Souag, A., Salinesi, C., Wattiau, I., Mouratidis, H.: Using Security and Domain Ontologies for Security Requirements Analysis. In: IEEE 37th Annual Computer Software and Applications Conference Workshops (COMPSACW), pp. 101–107 (2013)
10. Salinesi, C., Ivankina, E., Angole, W.: Using the RITA Threats Ontology to Guide Requirements Elicitation: an Empirical Experiment in the Banking Sector. In: First International Workshop on Managing Requirements Knowledge, MARK 2008, pp. 11–15 (2008)

11. Daramola, O., Sindre, G., Moser, T.: Ontology-Based Support for Security Requirements Specification Process. In: Herrero, P., Panetto, H., Meersman, R., Dillon, T. (eds.) OTM-WS 2012. LNCS, vol. 7567, pp. 194–206. Springer, Heidelberg (2012)
12. Velasco, J.L., Valencia-Garcia, R., Fernandez-Breis, J.T.,, T.: Modelling Reusable Security Requirements Based on an Ontology Framework. Journal of Research and Practice in Information Technology 41(2), 119 (2009)
13. Salini, P., Kanmani, S.: A Knowledge-oriented Approach to Security Requirements for an E-Voting System. International Journal of Computer Applications 49(11), 21–25 (2012)
14. Dritsas, S., Gymnopoulos, L., Karyda, M., Balopoulos, T., Kokolakis, S., Lambrinoudakis, C., Katsikas, S.: A knowledge-based approach to security requirements for e-health applications. Electronic Journal for E-Commerce Tools and Applications (2006)
15. Massacci, F., Mylopoulos, J., Zannone, N.: An ontology for secure socio-technical systems. Handbook of Ontologies for Business Interactions. IDEA Group (2007)
16. Blanco, C., Lasheras, J., Fernández-Medina, E., Valencia-García, R.,, T.: Basis for an integrated security ontology according to a systematic review of existing proposals. Computer Standards and Interfaces 33(4), 372–388 (2011)
17. Undercoffer, J., Joshi, A., Pinkston, J.: Modeling Computer Attacks: An Ontology for Intrusion Detection. In: The 6th International Symposium on Recent Advances in Intrusion Detection, pp. 113–135 (2003)
18. Geneiatakis, D., Lambrinoudakis, C.: An ontology description for SIP security flaws. Computer Communications 30(6), 1367–1374 (2007)
19. Denker, G., Kagal, L., Finin, T.W., Paolucci, M., Sycara, K.: Security for DAML Web Services: Annotation and Matchmaking. In: Fensel, D., Sycara, K., Mylopoulos, J. (eds.) ISWC 2003. LNCS, vol. 2870, pp. 335–350. Springer, Heidelberg (2003)
20. Denker, G., Nguyen, S., Ton, A.: OWL-S Semantics of Security Web Services: a Case Study. In: Bussler, C.J., Davies, J., Fensel, D., Studer, R. (eds.) ESWS 2004. LNCS, vol. 3053, pp. 240–253. Springer, Heidelberg (2004)
21. Karyda, M., Balopoulos, T., Dritsas, S., Gymnopoulos, L., Kokolakis, S., Lambrinoudakis, C., Gritzalis, S.: An ontology for secure e-government applications. In: The First International Conference on Availability, Reliability and Security, ARES 2006, p. 5 (2006)
22. Tsoumas, B., Gritzalis, D.: Towards an Ontology-based Security Management. In: 20th International Conference on Advanced Information Networking and Applications, AINA 2006, vol. 1, pp. 985–992 (2006)
23. Herzog, A., Shahmehri, N., Duma, C.: An Ontology of Information Security. International Journal of Information Security and Privacy 1(4), 1–23 (2007)
24. Fenz, S., Ekelhart, A.: Formalizing information security knowledge. In: Proceedings of the 4th International Symposium on Information, Computer, and Communications Security, New York, NY, USA, pp. 183–194 (2009)
25. Fernández-López, M., Gómez-Pérez, A., Juristo, N.: METHONTOLOGY: From Ontological Art Towards Ontological Engineering. In: Proceedings of the Ontological Engineering AAAI-97 Spring Symposium Series, Stanford University, EEUU (1997)
26. Jones, D., Bench-capon, T., Visser, P.: Methodologies For Ontology Development. In: Proceedings IT&KNOWS Conference of the 15th IFIP World Computer Congress, pp. 62–75 (1998)
27. Mayer, N.: Model-based Management of Information System Security Risk. Presses universitaires de Namur (2012)
28. Vogel, V.: Information Security Guide, `https://wiki.internet2.edu/confluence/display/itsg2/Overview+to+the+Guide`

29. ISO/IEC 13335-1:2004 Information technology – Security techniques – Management of information and communications technology security – Part 1: Concepts and models for information and communications technology security management (2004)
30. Staab, S., Maedche, A.: Axioms are Objects, too – Ontology Engineering beyond the Modeling of Concepts and Relations. In: Workshop on Applications of Ontologies and Problem-Solving Methods, ECAI 2000, Berlin (2000)
31. Lekhchine, R.: Construction d'une ontologie pour le domaine de la sécurité: application aux agents mobiles (2009)
32. Sure, Y., Angele, J., Staab, S.: OntoEdit: Guiding Ontology Development by Methodology and Inferencing. In: Meersman, R., Tari, Z. (eds.) CoopIS 2002, DOA 2002, and ODBASE 2002. LNCS, vol. 2519, pp. 1205–2011. Springer, Heidelberg (2002)
33. Farquhar, A., Fikes, R., Rice, J.: The Ontolingua Server: a tool for collaborative ontology construction. International Journal of Human Computer Studies 46(6), 707–727 (1997)
34. Horridge, M., Knublauch, H., Rector, A., Stevens, R., Wroe, C.: A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools Edition 1.0. University of Manchester (2004)
35. O'Connor, M.J., Das, A.K.: SQWRL: A Query Language for OWL. In: OWLED, vol. 529 (2009)
36. Uschold, M., Gruninger, M., Uschold, M., Gruninger, M.: Ontologies: Principles, methods and applications. Knowledge Engineering Review 11, 93–136 (1996)
37. Kitchenham, B.A., Pfleeger, S.L., Pickard, L.M., Jones, P.W., Hoaglin, D.C., El Emam, K., Rosenberg, J.: Preliminary guidelines for empirical research in software engineering. IEEE Transactions Software Engineering 28(8), 721–734 (2002)
38. de la Défense Nationale, S.G.: EBIOS-Expression des Besoins et Identification des Objectifs de Sécurité (2004)
39. Mouratidis, H., Giorgini, P.: Secure Tropos: A Security-Oriented Extension of the Tropos Methodology. International Journal of Software Engineering and Knowledge Engineering 17(02), 285–309 (2007)
40. Kim, A., Luo, J., Kang, M.: Security Ontology for Annotating Resources. In Research Lab, NRL Memorandum Report, p. 51 (2005)
41. Martimiano, A.F.M., Moreira, E.S.: An owl-based security incident ontology. In: Proceedings of the Eighth International Protege Conference, pp. 43–44 (2005)
42. Lawrence, P.S.: Experimental design and analysis in software engineering. Annals of Software Engineering 1(1), 219–253 (1995)
43. Davis, F.D.: Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. MIS Quarterly, 319–340 (1989)
44. Norton, 2013 Norton Cybercrime report (July 2013)

Appendix

Fig. 4 in the appendix presents the core security ontology. Table 8 in the appendix was built up for ontology concepts definition. It includes the ontologies used as an entrance point.

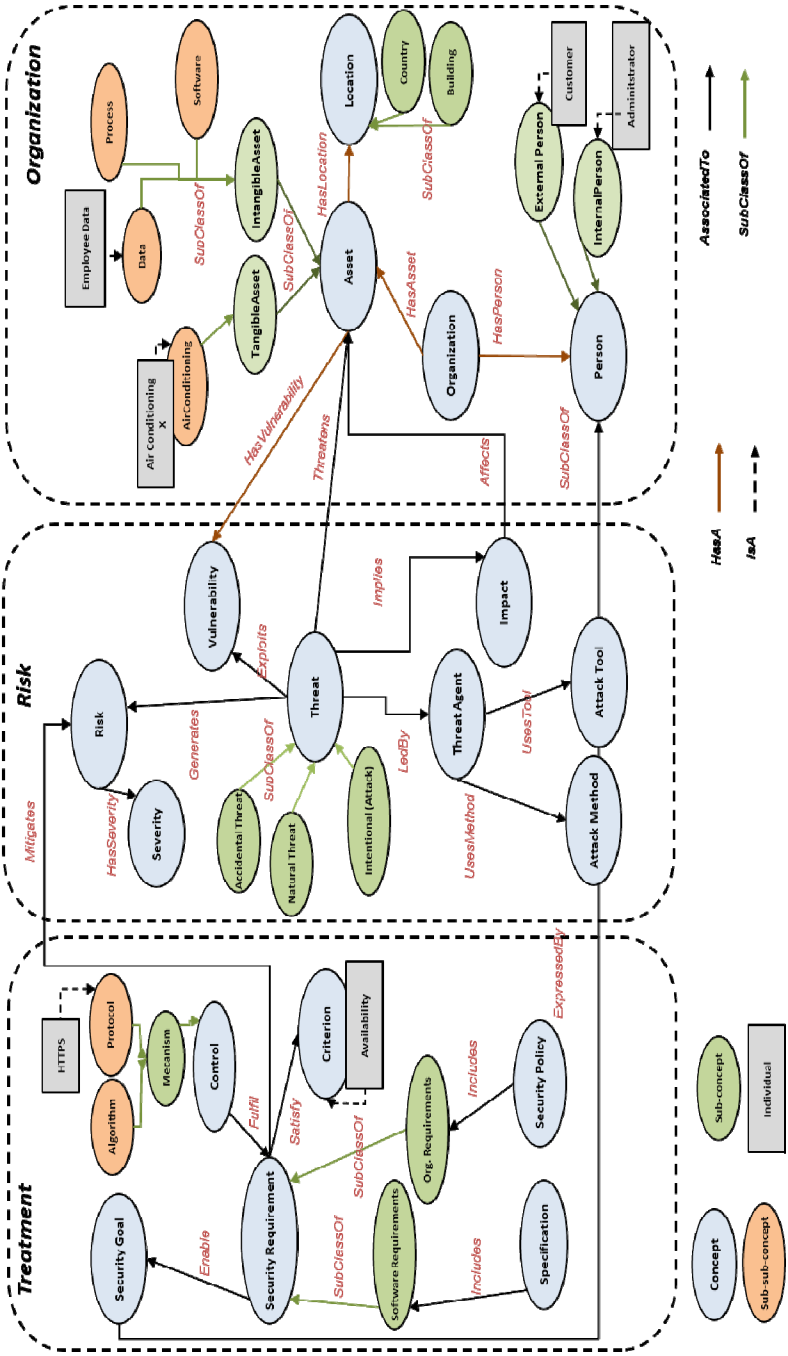


Fig.4. The core security ontology

Table 8. Ontology concepts definition using security ontologies and models from literature

	Concepts of the ontology	Mayer et al. [27]	Tsoumas et al. [22]	Herzog et al.[23]	Fenz et al. [24]	Lashera s et al.[12]	Dritsas et al. [14]	Karyda et al.[21]	Kim et al. [40]	Undercoffer et al. [17]	Geneiataki s et al. [18]	Denker et al.[1]	Lekhchine et al.[31]	Martimiano et al. [41]
Organization dimension	Asset	Business Asset	Asset	Asset	Asset	Asset	Asset	Asset	-	System component	Target	-	Asset	Asset
	Location	-	-	-	Location	-	-	-	-	Location	-	-	-	-
	Organization	-	-	-	Organization	-	-	-	-	-	-	-	Organization	-
	Person	-	Stakeholder	-	Person	-	Stakeholder	Person	-	-	-	-	Person	Supplier
Risk dimension	**Threat**	Threat	Threat	Threat	Threat	Threat	Threat	Threat	-	-	-	-	Threat	Attack
	Vulnerability	Vulnerability	Vulnerability	Vulnerability	Vulnerability	-	Vulnerability	-	-	Means	-	-	Vulnerability	Vulnerability
	Risk	Risk	Risk	-	-	Risk	-	-	-	-	-	-	-	-
	Severity	-	-	-	Attribute	-	-	-	-	-	-	-	-	-
	Impact	Impact	Impact	-	-	Valuation criteria	-	-	-	Consequence	-	-	-	Consequence
	Threat agent	Threat agent	Threat agent	-	Role	-	Attacker	-	-	-	-	-	-	Agent
	Attack method	Attack method	Attack	-	-	-	Deliberate attack	-	-	Intrusion	SIP_attack	-	-	-
	Attack tool	-	-	-	-	-	-	-	-	-	SIP message	-	-	Tool
Treatment dimension	Security goal	-	-	-	-	-	-	-	-	-	-	-	-	-
	Security criterion	Security criterion	-	goal	Security attribute	-	Objective	Security objective	Security objective	-	-	Security notation	Objective mechanism	Acess
	Requirements document	-	Security policy	-	-	-	-	-	-	-	-	-	-	-
	Control	Control	Countermeasure	Countermeasure	Asset	Safeguard	Countermeasure	Counter-measure	Security mechanism/ Credential	-	-	Security mechanism/ Credential	Countermeasure/ Protocol/ Algorithm	Correction
	Security requirement	Security requirement	Control	-	Control	Security requirement	-	-	-	-	-	-	-	-