Ontology-based Risk Control for the Incident Management

Tung Ju Chiang and Jen Shiang Kouh and Ray-I Chang,

Dept. of Engineering Science and Ocean Engineering, National Taiwan University, Taipei Taiwan

Summary
Both non-profit and commercial organizations rely heavily on information to process their daily activities. The information security management standards are widely used and advocated by researchers and practitioners to reduce security incidents and lower down risk. One problem of information security management is in compliance with new and never-ending best practices, regulation and legislation. In this work we proposed an ontological mapping of the ISO/IEC 27001 standard, IT security EBK and its control countermeasure in combination with our Security Ontology approach. For the purpose of the reusability, interoperability, aggregation and reasoning of the security knowledge.

Key words:
Ontology, owl, protégé, information security, iso 27001.

1. Introduction

Cyber-security is vital to the operation of safety critical systems, such as emergency response, and to the protection of infrastructure systems, such as the national power grid. Due to the impact of the Enron failure and the implementation of the Sarbanes-Oxley Act, many companies within the financial sectors have to comply with Sarbanes Oxley (SoX) Act. The Gramm-Leach-Bliley Act (GLBA), an Act of the United States Congress, open up competition among banks, securities companies and insurance companies. GLBA compliance is mandatory; whether a financial institution discloses nonpublic information or not, there must be a policy in place to protect the information from foreseeable threats in security and data integrity. So many standards exist in the company mean the existing problems of maintain and integration. In response to the legal compliance problem, there is need for a storage of information security legislation, standards, and case authority, which will provide a user with a single resource for the investigation of relevant legal influences. In this paper, we try to apply ontology to be the central storage of knowledge. It can use a security management framework of an information system which builds upon knowledge-based resources by security ontology to provide reusable security knowledge interoperability, aggregation and reasoning exploiting security knowledge from diverse sources.

2. Ontology

There are some reasons to develop and to use ontologies are [1]:

- sharing common understanding of the structure of information among people or software, i.e. mainly the structure of the components, generics and evidences;
- reusing the domain knowledge, i.e. using the same specification means in different projects and deriving its new variants from the previously defined ones;
- making explicit assumptions for a domain; it concerns predefined parameters and predefined mapping relations between specification items;
- separating the domain knowledge, expressed by the specification means as a whole, from the operational knowledge allowing to use these means to compose the ST of the given IT products or systems;
- providing the domain knowledge analyses concerning: variants, semantics, risk, relationships of the developed specification means, etc.

Property Restrictions: Object property restrictions are used to create constraints on individuals that belong to a particular class. Restrictions fall into three categories: Quantifier, Cardinality and hasValue restrictions. An existential ($\exists$) restriction requires at least one relationship for a given property to an individual that is a member of a specific class. A universal ($\forall$) restriction mandates that the only relationships for the given property that can exist must be to individuals that are members of the specified class. A property restriction effectively describes an anonymous or unnamed class that contains all the individuals that satisfy the restriction. When restrictions are used to describe classes they specify anonymous superclasses of the class being described. When building the domain ontology, it’s important to define the class properties (slots) and their restrictions (facets) which describe or limit the set of possible values for the given slot. There are three standard kinds of slots that can be inherited [1]:

- object (instance-type) slots which represent relationships between an individual of the given class and other individuals, expressing parts of the structured concepts or other complex properties;

Manuscript received November 5, 2009
Manuscript revised November 20, 2009

• data-type slots of integer, byte, float, time, date, enumeration or string values, expressing simple properties of the individuals of the most elementary classes;
• annotation slots (documentation) which represent the meaning of the given concept.

Classes are interpreted as sets of individuals and can be organised into a super class-subclass hierarchy. For example, Protocol is a class that represents the set of all individual protocols and its subclasses include TCP and UDP classes. Subsumption represents the superclass subclass hierarchy, for example, TCP $\subseteq$ Protocol indicates that TCP is a subclass of Protocol.

$$\text{Man} \sqsubseteq \text{Person}, \text{Woman} \sqsubseteq \text{Person}$$
$$\text{Bob} \in \text{Man}, \text{Mary} \in \text{Woman}$$

2.1 Protégé and OWL

There are three dialects of OWL: OWL-Lite, OWL DL and OWL Full. OWL-Lite is the syntactically simplest sublanguage. OWL-DL is much more expressive than OWL-Lite and is based on Description Logics (hence the suffix DL). Description Logics are a decidable fragment of First Order Logic and are therefore amenable to automated reasoning. It is therefore possible to automatically compute the classification hierarchy and check for inconsistencies in an ontology that conforms to OWL-DL. OWL-Full is the most expressive OWL sub-language. It is intended to be used in situations where very high expressiveness is more important than being able to guarantee the decidability or computational completeness of the language. It is therefore not possible to perform automated reasoning on OWL-Full ontologies. OWL was selected in order to enhance our model with automated reasoning facilities. Reasoning will permit us to derive new knowledge based on an initial set of rules.

Prot´eg´e is an open-source tool developed at Stanford Medical Informatics. It has a community of thousands of users. Although the development of Prot´eg´e has historically been mainly driven by biomedical applications, the system is domain-independent and has been successfully used for many other application areas as well [2]. There are many benefits for using the Prot´eg´e tool:

- free, open-source ontology editor and knowledge-base framework.
- Can be used by domain experts
- Better scalable than visual UML modeling
- Reasoning support at edit-time
- Rapid prototyping of models
- Individuals can be acquired using forms
- Open architecture / adaptability
- Start one's application as a plugin

2.2 SWRL

Rules are widely used in business applications including workflow management, awareness, training, education, diagnostic fact finding, compliance monitoring, and process control. Rule-based Systems are common in many domains: 1) Engineering: Diagnosis rules; 2) Commerce: Business rules; 3) Law: Legal reasoning; 4) Medicine: Eligibility, Compliance; 5) Internet: Access authentication. The Semantic Web Rule Language (SWRL) is one way to define a rule language. SWRL [3] allows users to write Horn-like rules expressed in terms of OWL concepts to reason about OWL individuals. The rules can be used to infer new knowledge from existing OWL knowledge bases. SWRL is used to query the knowledge base to check whether specific security requirements are fulfilled.

The SWRL Editor is an extension to Protégé-OWL that permits the interactive editing of SWRL rules. The editor can be used to create SWRL rules, edit existing SWRL rules, and read and write SWRL rules. It is accessible as a tab within Protégé-OWL.

There are two ways of interacting with the SWRL Editor in Protege-OWL:

1) The primary mechanism is through the SWRL Rules tab. This tab shows all the SWRL rules in a loaded OWL knowledge base in tabular form.
2) A second mechanism allows users to find rules relating to a selected OWL class, property, or individual in the respective Protege-OWL tabs for those entities.

SWRL is an acronym for Semantic Web Rule Language and intended to be the rule language of the Semantic Web. It includes a high-level abstract syntax for Horn-like rules. All rules are expressed in terms of OWL concepts (classes, properties, individuals).

SWRL also provides so-called "built-ins" that allow user-defined methods to be used in rules. In the rules below we use some core built-ins swrlb :startsWith which returns true if the first argument starts with the second argument or swrlb : greaterThanOrEqual which compares two values.

SWRL also supports the common same-as and different-from concepts. For example, the SWRL sameAs atom can determine if two OWL individuals ISO27001 and ISO17799:2005 are the same individual. The SWRL rule is shown below and the OWL code is below and The OWL form code is shown next.

sameAs(ISO27001, ISO17799:2005)

```
<owl:Class rdf:ID="ISO27001">
  <owl:equivalentClass>
    <owl:Class rdf:about="#ISO17799:2005"/>
  </owl:equivalentClass>
 </owl:Class>
```

A SWRL rule is composed of an antecedent (body) part and a consequent (head) part, both of which consist of positive conjunctions of atoms. In this syntax, a rule has the form:

antecedent ⇒ consequent

For example, the requirement: servers hosting ssh based business services protected by a firewall require that firewall to open port 22 is expressed in SWRL as:

Server(?n)^hasHosted(?n,?s)^hasPort(?s,ssh)^hasFirewall(?n,?f)->hasPortOpen(?f,ssh)

Benefits of OntoSec

Benefits of OntoSec in Information Security Management

It can be pointed out the following advantages of using ontologies to assist the information security management:

1) The development of ontologies creates a conceptual model that makes it possible to the organization to know better its security incidents domain.

2) The ontologies can facilitate the interoperability among different security tools, creating a unique way to represent security data and, for instance, allowing that security alerts from any security tool is mapped into an ontology.

3) The Security Incident Ontology imports the Vulnerability Ontology, allowing the reuse of knowledge and information. Other ontologies about security domain could be imported, such as a Virus Ontology or a Worm Ontology. The same reuse can be scaled up in such a way that security information can be treated in a more abstract level.

4) The querying and inference process helps the security administrators to be more confident of the decisions made about the security information management, because the ontology developed is knowledge bases about security incidents. The ontology allows the security administrators to learn from previous security problems, assisting them in solving and preventing new problems.

Table 1: Taxonomy Hierarchy of the ISO/IEC 27001 Standard

	Number	Title	Description
Category	A.9	Physical and environmental security	
Objective	A.9.1	Secure areas	To prevent unauthorized physical access, damage and interference to the organization’s premises and information.
Control	A.9.1.1	Physical security perimeter	Security perimeters (barriers such as walls, card controlled entry gates or manned reception desks) shall be used to protect areas that contain information and information processing facilities.
Control	A.9.1.2	Physical entry controls	Secure areas shall be protected by appropriate entry controls to ensure that only authorized personnel are allowed access.

In this paper we model the firewall configuration expertise with ontologies, using OWL for knowledge representation. Our formal representation will allow formally disambiguating and structuring the represented knowledge. Further, we employ the Semantic Web Rule Language (SWRL) for policy validation i.e. detecting firewall conflicts. The power of our tools will reside in a dual usage of reasoners. Ontology reasoners will allow integration of new rules with existing ones, while SWRL knowledge reasoners will allow validation and conflict detection.

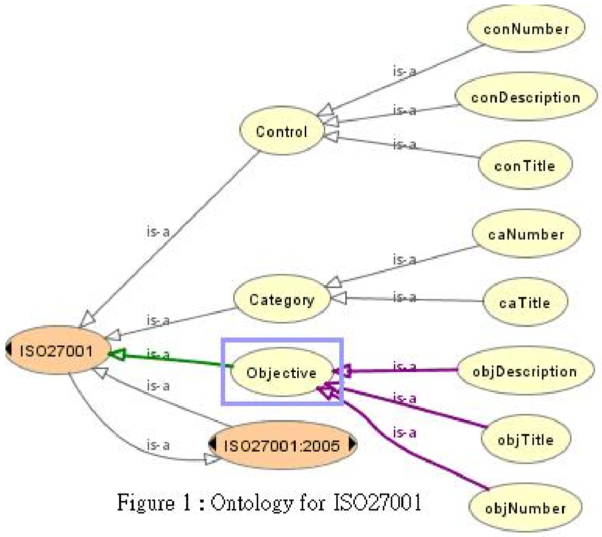


Figure 1 : Ontology for ISO27001

3. ISO/IEC27001

The aim of this paper is to explain how security ontologies can be used for a tool to support the ISO/IEC 27001 certification, providing pivotal information for the preparation of audits and the creation and maintenance of security guidelines and policies [4]. Thus, we propose an ontological mapping of the ISO/IEC 27001 standard to increase the degree of automation within the certification process, lowering the financial costs and time required for the certification procedure [4].

Standard ISO/IEC 27001 [5] defines the best practice code in the area of information security. The entire standard is based on eleven security categories (chapters) which cover all the aspects of information security. These categories are:

As shown in Figure 1, our ontology is comprised of a number of different concepts. Each individual concept has a relationship with one or more other concepts. The objects Chapter, Section, Guideline and Guideline Step provide representation of content from the ISO27001 information security standard. An individual Guideline can be associated with a particular Asset by way of the 'hasSubject' relation. Otherwise if a Guideline is broken down into more

Datatype Property

All attributes are represented in OWL as a Datatype Property. This property also defines which are the Domain Resource and the Range Resource. In this case, the Domain Resource is the class which has the attribute and the Range Resource is the type of attribute. Similarly to the Object Property, the Datatype Property also can have a restriction as cardinality, representing how many instances the property can have. It is also possible to predefine instances. For example, the attribute has severity can have only one of the following instances: Low, Medium or High.

Due to the very flat structure of the ISO/IEC 27001 standard, described in Table 1, we were able to map the entire standard to the ontology using only three classes: Category, Objective, Control and four relations: hasCategoryObj and its inverse relation hasObjCategory, hasObjControl and its inverse relation hasControlObj.

```
<owl:ObjectProperty rdf:about=”#implies_to_aConsequence”>
      <rdfs:domain rdf:resource=”#Security_Incident”/>
       <rdfs:range rdf:resource=”#Consequence”/>
</owl:ObjectProperty>
```

Security Incident acting on Asset can be expressed below:

```
<owl:ObjectProperty rdf:about=”#acts_onAsset”>
      <rdfs:domain rdf:resource=”#Security_Incident”/>
       <rdfs:range rdf:resource=”#Asset”/>
</owl:ObjectProperty>
```

4. IT security EBK of Homeland Security

To assist organizations and current and future members of this workforce, the Department of Homeland Security National Cyber Security Division (DHS-NCSD) worked with experts from academia, government, and the private sector to develop a high-level framework that establishes a national baseline representing the essential knowledge and skills IT security practitioners should possess to perform. DHS-NCSD developed the IT Security Essential Body of Knowledge (EBK): A Competency and Functional Framework for IT Security Workforce Development as an umbrella document that links competencies and functional perspectives to IT security roles fulfilled by personnel in the public and private sectors [7]. There are two way to implement IT security EBK in the Protégé tool. One way is to use three tiers structure like the ISO 27001 framework we used in the Table 1. This is a good way to use the same structure for the purpose of the standard integration and comparison of the different standards. Another way is to set the chapter and section as the class, not the individual as ISO27001 done before. It can be a easy way to implement the standard or best practice to the OWL format and the relational database format. Figure 2 shows “2.4 incident management” of the IT security EBK.

It do not convey a lifecycle concept of task or program execution as is typical of a traditional system development lifecycle (SDLC), but are used to sort functions of a similar nature. The functional perspectives are defined as follows:

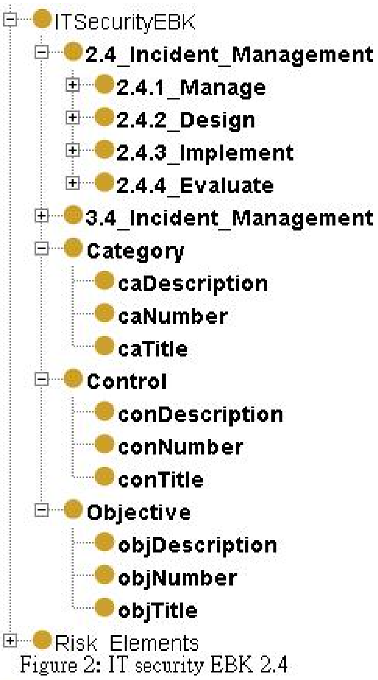
Manage: Functions that encompass overseeing a program or technical aspect of a security program at a high level, and ensuring currency with changing risk and threat environments.

Design: Functions that encompass scoping a program or developing procedures, processes, and architectures that guide work execution at the program and/or system level.

Implement: Functions that encompass putting programs, processes, or policies into action within an organization.

Evaluate: Functions that encompass assessing the effectiveness of a program, policy, process, or security service in achieving its objectives.

Incident Management (2.4):



```
ITSecurityEBK
    2.4_Incident_Management
        2.4.1_Manage
        2.4.2_Design
        2.4.3_Implement
        2.4.4_Evaluate
    3.4_Incident_Management
    Category
        caDescription
        caNumber
        caTitle
    Control
        conDescription
        conNumber
        conTitle
    Objective
        objDescription
        objNumber
        objTitle
Risk Elements
```

Figure 2: IT security EBK 2.4

Refers to knowledge and understanding of the process to prepare and prevent, detect, contain, eradicate, and recover, and the ability to apply lessons learned from incidents impacting the mission of an organization.

Manage (2.4.1)

• Coordinate with stakeholders to establish the incident management program

• Establish relationships between the incident response team and other groups, both internal (e.g., legal department) and external (e.g., law enforcement agencies, vendors, and public relations professionals)

• Acquire and manage resources, including financial resources, for incident management Functions

• Ensure coordination between the incident response team and the security administration and technical support teams
• Apply lessons learned from information security incidents to improve incident management processes and procedures
• Ensure that appropriate changes and improvement actions are implemented as required
• Establish an incident management measurement program.

Design (2.4.2)

• Develop the incident management policy, based on standards and procedures for the organization
• Identify services that the incident response team should provide
• Create incident response plans in accordance with security policies and organizational goals
• Develop procedures for performing incident handling and reporting
• Create incident response exercises and penetration testing activities
• Develop specific processes for collecting and protecting forensic evidence during incident response
• Specify incident response staffing and training requirements
• Establish an incident management measurement program.

Implement (2.4.3)

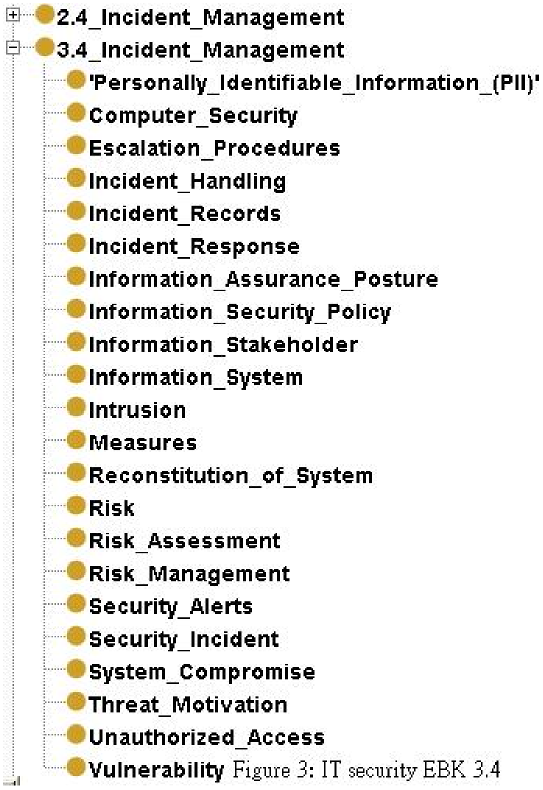
• Apply response actions in reaction to security incidents, in accordance with established policies, plans, and procedures
• Respond to and report incidents
• Assist in collecting, processing, and preserving evidence according to standards, procedures, directives, policies, regulations, and laws (statutes)
• Monitor network and information systems for intrusions
• Execute incident response plans
• Execute penetration testing activities and incidence response exercises
• Ensure lessons learned from incidents are collected in a timely manner, and are incorporated into plan reviews
• Collect, analyze, and report incident management measures
• Coordinate, integrate, and lead team responses with internal and external groups according to applicable policies and procedures.

Evaluate (2.4.4)

• Assess the efficiency and effectiveness of incident response program activities, and make improvement recommendations
• Examine the effectiveness of penetration testing and incident response tests, training, and exercises
• Assess the effectiveness of communications between the incident response team and related internal and external organizations, and implement changes where appropriate
• Identify incident management improvement actions based on assessments of the effectiveness of incident management procedures.

Incident Management (3.4)

Refers to knowledge and understanding of the process to prepare and prevent, detect, contain, eradicate, recover, and apply lessons learned from incidents impacting the mission of an organization. The content of the IT security EBK 3.4 is shown on Figure 3.



```
⊞ 2.4_Incident_Management
⊟ 3.4_Incident_Management
    'Personally_Identifiable_Information_(PII)'
    Computer_Security
    Escalation_Procedures
    Incident_Handling
    Incident_Records
    Incident_Response
    Information_Assurance_Posture
    Information_Security_Policy
    Information_Stakeholder
    Information_System
    Intrusion
    Measures
    Reconstitution_of_System
    Risk
    Risk_Assessment
    Risk_Management
    Security_Alerts
    Security_Incident
    System_Compromise
    Threat_Motivation
    Unauthorized_Access
    Vulnerability
```

Figure 3: IT security EBK 3.4

The Information Security Officer (ISO) specializes in the information and physical security strategy within an organization. The ISO is charged with the development and subsequent enforcement of the company's security policies and procedures, security awareness program, business continuity and disaster recovery plans, and all industry and governmental compliance issues.

The OWL individual can be asserted as a member of the class Role:

(assert (Role (name ISO)))

(assert (Job (name CyberSecurityOfficer)))

sameAs(ISO, CyberSecurityOfficer)

• Incident Management: Manage, Design, Evaluate

Example Job Titles:

- Cyber Security Officer
- Chief Information Security Officer (CISO)
- Enterprise Security Officer
- Information Security Officer
- Senior Agency Information Security Officer

5. Ontology of the Risk Management

5.1 Security Incident Ontology

The main classes of the Security Incident Ontology are described as following [8]:

- Access: This class represents the type of accesses an agent can have.
- Agent: This class represents the entity that performs one or more attacks in order to cause a security incident.
- Asset: This class represents the target of a security incident. Assets are information or resources which have value to an organization or person. A stakeholder is an organization or person who places a particular value on assets. Anything that provides value to the organization is belong to Asset. It is classified within a hierarchy of types of assets. MAGERIT [13] distinguishes nine disjoint types of assets: Service, Media, Communication, Software, Hardware, Data Information, Auxiliary Equipment, Installations and Personnel. These assets have been grouped in four disjoint classifications: Organisation Functions, Information System, Information, and Environment assets.
- Attack: This class represents the attack itself performed by the agent. An attack is an action that violates the security of an asset. An attacker is the entity which carries out attacks.
- Consequence: This class represents the consequences a security incident can imply.
- Security Incident: This is the most important class. It represents the security incident caused by an agent through an attack.
- Time: This class represents information about when the security incident happened.
- Tool: This class represents the means, used by an agent, of exploiting a computational system.
- Vulnerability: Vulnerability is a flaw or weakness that could be exploited to breach the security of an asset. This class represents the types of vulnerabilities a system can have and it imports the vulnerability ontology.
- Security Objective is a statement of intent to counter threats and satisfy identified security needs.
- Threat is the possible threats associated to the assets in an information system. Four main disjoint types of threats have been identified: Natural disasters, Industrial Origin, Errors and Unintentional Failures (unintentional failures caused by people) and Wilful Attacks (deliberate failures caused by people). A threat is a potential for a security breach of an asset.
- Countermeasure is an action taken in order to protect an asset against threats and attacks. Every information asset is associated with certain threats, which can be mitigated by a set of countermeasures. **[9]**
- The risk is the probability that a successful attack occurs.
- Safeguard: the safeguards that allow threats to be faced. For example, access control, record of actions or back-up copies.
- Valuation dimension: the features or attributes that make an asset valuable. This is the measurement of the loss caused by damages in an asset in a certain dimension: availability, integrity, confidentiality, authenticity and accountability. For example the availability dimension of an asset means “How important would it be if the asset was not available?”
- Valuation criteria: The criteria which made an asset valuable for an organization, in other words, how interesting the asset is for the system. We must protect the most valuable asset. It includes a numerical value (1-10) of their importance, and a rationale for this.

Ontology for Vulnerability Management (OVM) is developed with populating all the software products vulnerability information from National Vulnerability Database (NVD). OVM captures the relationships between IT products, vulnerabilities, attackers, security metrics, countermeasures, and other relevant concepts. As information security is such a complex field that the scope and volume of security data overwhelm security professionals and administrators, the importance of the OVM manifests itself in the practice of building automated tools for system security [11].

The risk analysis ontology contains relations, constraints, axioms and rules. For example, there is a binary relationship between the assets and the threats to represent which threat can affect which asset. The semantics of this generic relation is completed at each concept (of the assets and threats taxonomies) by adding the corresponding range and domain constraints in OWL. For example, the Errors and Unintentional Failures (threat) affect the Software (asset) but not the Personnel (asset), and Traffic Analysis (threat) only affects Communication (asset). Moreover, other constraints such as disjointedness or cardinality can also be defined in OWL and have been very useful for the construction of this ontology (and the following ones, described in the next sections). The new properties, and their OWL modelling, are described below [12]:

- has_asset: every security requirement has to be related to one asset. Thus, an Object property has been added to the concept security requirement to represent its associated asset. The range of the Object property is the class Asset defined in the risk analysis ontology. This Object property is inherited and restricted throughout the hierarchy. Furthermore storing the owner, the person and the unit responsible for the asset are relevant. This information is an objective in ISO 27002 (2005).

- has_threats: it represents possible threats associated to the non-fulfilment of the requirements. This property is represented by an Object property over the hierarchy of threats of the risk analysis ontology and so its range is the class Threat defined in this ontology. The risk analysis ontology has constraints of which threat can be occurred to which asset, so in the security requirements ontology we can infer if a threat associated to a requirement can be inconsistent with the asset associated to this requirement.
- has_valuation_dimensions: the features that make an asset valuable. There exist five valuation dimensions modelled using an Object property: “Availability”, “Integrity”, “Confidentiality”, “Accountability” and “Authenticity”.
- has_safeguards: This Object property associates the related requirement to the safeguard. Information about the efficacy to confront a threat and its state of implantation must be stored. With this combination, the Source of the requirement becomes essential. It specifies the security

standards or current legislation a requirement has been derived from. The current version is available at http://dis.um.es/~jolave/securityRequirements.owl.

In this case, the asset defined for the requirement is consistent with the threat that affects it. However this check could not be performed in Protégé with OWL, so we have to do some semantic queries to verify. We can use the semantic web query language SPARQL [10] by means of two queries:

Query#1: to select the requirements and assets in the sense that all the assets associated to every requirement have to be the same asset on which the threat of the requirement acts.

```
SELECT distinct ?x ?asset
WHERE{ ?x :has_asset ?asset ; :has_threat ?threat.
?threat magerit:has_asset ?b. filter(sameTerm(?asset,?b) )
} orderby ?x ?asset
```

Query#2: to check that the asset associated to the requirement can be found in the assets that are affected by the threat. So, the next query selects all the requirements and their assets.

```
SELECT distinct ?x ?asset
WHERE{ ?x :has_asset ?asset ;
} orderby ?x ?asset
```

In Protégé Tool, the hierarchical relations are called Asserted Hierarchy and the non-hierarchical relations and the attributes are called Properties. The DataType Properties represent the attributes and the ObjectType Properties represent the non-hierarchical relations, which in the Security Incident Ontology represent the events between classes. For instance, the main ObjectType Properties of the class Security Incident are [8]:

- acts_on with the class Asset;
- happens_on with the class Time;
- implies_to_a with the class Consequence;
- proceeds and precedes, which are self-relations.

Table 2:Ontology of the Risk Management

Sub Class Name	Value
RiskNumber	R09001
Standard	ISO27001
Class	HumanResourcesSecurityControl
Target	employees, contractors, third party users
exePriod	[now | Month | Year]
Constraint	{Time, place, and Subject constrains}
Attributes	[Confidentiality | Integrity | Availability | |Authentication|AccessControl|Non-repudiation]
controlType	[Managerial | Procedural | Technical]
RiskMitigation	[High | Medium | Low]
ControlCost	[High | Medium | Low| CreditCost | Fees]
Desciption	As part of their contractual obligation, employees, contractors and third party users shall agree and sign the terms and conditions of their employment contract, which shall state their own and the organizations responsibilities regarding information security.
controlDate	2009.1.2
finishDate	2009.3.2
isImplemented	Yes
controlFrom	A.8.1.3 (ISO/IEC 27001)
prevents	SocialEngineering
LinkPolicy	Information Security Policy
LinkProcedure	Human Management Procedure
controlPurpose	[Security | Audit]
processOwner	Kevin
RiskBefore	High
RiskAfter	Low
Comment	

Ontology of the Risk Management

ISO 31000, a risk management guidance standard, is a generic standard set that intended to support the existing standards to deal with specific risks. Risk management, according to ISO 31000 draft report, involves applying logical and systematic methods for (www.iso.org/rm): [6]

1) communicating and consulting throughout this process; 2) establishing the organization's context for identifying, analyzing, evaluating, treating, and monitoring risk associated with any activity, product, function or process; and 3) reporting the results appropriately.

The risk management plan is made after the risk assessment, incident happened or audit of the organization. The sub-class and it's individual of the ontology of the risk control is shown on the Table 2. It is needed a risk plan item number for identifying and easily retrieved the content. For the purpose of control and evidence, the control happens and finish date is required. The constraint is the restriction of the implementation of the control item. When considering the security attributes, there are two mnemonics commonly used to summarize services which a network security system should provide: 'CIA' and 'Triple A'. CIA provides a key to remember three important security services (Confidentiality, Integrity and Availability), but really another three services should be added (Authentication, Access Control and Non-repudiation). The Link Policy and Link Procedure is the relationship of the risk management plan mapping to the policy and procedure of the organization. The process owner is the person who is responsible to finish the processes necessary to achieve the objectives of this risk-management plan. The risk value is needed to record before and after the implementing of the security control.

Conclusion

There are three mayor reasons for using the ontology: Share common understanding, Reusing knowledge and Interoperibility. Several Web sites contain information about vulnerabilities. The Security Incident Ontology reuses some concepts and relations of the Vulnerability Ontology, asset ontology. It can use the concepts and the relations defined by an Ontology to ease the interoperability that allows sharing data among different applications. In this paper, we proposed the methods of implementing the information security standard (ISO/IEC 27001) and best practice (IT security EBK) by the ontology. Furthermore, this contribution proposes the risk control ontology of the risk management. Combining the ontology built before, it can be a holistic framework tried to resolve the information security problem. In the future, we will focus on the Semantic Web Rule Language (SWRL) [3], W3C recommendation, based on a combination of OWL with the Unary/Binary Datalog RuleML sub languages of the Rule Markup Language, can be used to infer new knowledge from an existing OWL knowledge base. SWRL is a good solution for moving property values from one individual to another. Summarizing our main requirements for such an ontological framework results in the following list:

- **Rule based**: We emphasize the development of rule based systems, especially in domains where the underlying logic changes often. The rule language used must be highly expressive due to the complexity of the compliance statements;
- Maintainability: A clear separation of components (rules, business logic, and interfaces) strongly supports this attribute;
- OWL Knowledge Base: The framework has to operate directly on OWL files as this W3C standard has high potential and is now widely used
- **Compliance**: The framework of the information security management was built by the language of OWL in order to utilize the existing ontologies of threat, vulnerability, information security standards or best-practice guidelines for the purpose of the requirement of compliance by legislation requirement.

References

[1] N.F. Noy, D.L. McGuiness, Ontology Development 101: A Guide to Creating Your First Ontology, Knowledge Systems Laboratory, March, 2001. http://www-ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness-abstract.html

[2] Protégé Ontology Editor and Knowledge Acquisition System, Stanford University. http://protege.stanford.edu/

[3] SWRL: http://www.daml.org/rules/proposal/

[4] S. Fenz, G. Goluch, A. Ekelhart, and E. Weippl, Information Security Fortification by Ontological Mapping of the ISO/IEC 27001 Standard, In 13th Pacific Rim International Symposium on Dependable Computing, PRDC2007. IEEE Computer Society, December 2007.

[5] ISO/IEC-27001:2005, Information technology – security techniques – information security management systems – equirements.

[6] D.C. Chou, A.Y. Chou, Information systems outsourcing life cycle and risks analysis, Computer Standards & Interfaces 31 (2009) 1036–1043

[7] Office of Cybersecurity and Communications National Cyber Security Division, Information Technology (IT) Security Essential Body of Knowledge (EBK): A Competency and Functional Framework for IT Security Workforce Development, http://www.us-cert.gov/ITSecurityEBK/EBK2007.pdf

[8] A.F.M. Martimiano, E.S. Moreira, An OWL-based Security Incident Ontology, In: Proceedings of the Eighth International Prot´eg´e Conference. (2005) 43–44 Poster.

[9] M. Ahmed, A. Anjomshoaa, T.M. Nguyen, A.M. Tjoa, Towards an Ontology-based Organizational Risk Assessment in Collaborative Environments Using the SemanticLIFE, "The Second International Conference on Availability, Reliability and Security", IEEE Computer Society, (2007)

[10] SPARQL Query Language for RDF, W3C Recommendation 15 January 2008, http://www.w3.org/TR/rdf-sparql-query/

[11] J.A. Wang, M. Guo, OVM: An Ontology for Vulnerability Management. CSIIRW ’09, April 13-15, Oak Ridge, Tennessee, USA.

[12] 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, May 2009

[13] MAGERIT (2006): Methodology for information systems risk analysis and management. http://www.csi.map.es/csi/pg5m20.htm.

Tung Ju Chiang is a PhD student of the Department of Engineering Science and Ocean Engineering, National Taiwan University. He has 3 international certifications, ISO27001 Lead Auditor, MCSE and MCSD. He is a senior technician of Computer and Information Networking center of the National Taiwan University. His current research interests include information security management, ontology, risk management, Fuzzy.

Jen Shiang Kouh is a Professor of the Department of Engineering Science and Ocean Engineering, National Taiwan University. He is an expert in Computer Graphics, Geometrical Modeling, Computational Fluid Dynamics, Computer Simulation, knowledge management, X3D. He has many projects about the topics of the wind power plants and shape design of the underwater vehicle.

Ray-I Chang received his Ph.D. degree in Electrical Engineering and Computer Science from National Chiao Tung University in 1996. Then, he joined the Computer Systems and Communications (CSCL) Laboratory in the Institute of Information Science, Academia Sinica, Taiwan, ROC. In 2003, he joined the Department of Information Management, National Centural University. Now, he is an Associate Professor of the Department of Engineering Science and Ocean Engineering, National Taiwan University. Dr. Chang was the editorial board member of International Journal of Applied Metaheuristic Computing, Management and Information Science, and Journal of the Chinese Institute of Industrial Engineers. His current research interests include network security, wireless sensor networks, data mining, real-time scheduling and multimedia systems. Dr. Chang is a member of IEEE.