Security Requirements Engineering: Analysis and Prioritization

A. Daya Gupta¹, B. Shruti Jaiswal², and C. Anupriya Tewari³
¹,²,³Computer Engineering Department, Delhi Technological University, Delhi, India

Abstract - with the increase in the use of software system, security requirement engineering becomes an emergent area of study. Security requirements are constraints to a system which must be satisfied for consistent system. Most of the software engineering processes deals with security constraints during the design or implementation phases which may result into unnecessary constrained system. So the need for a new process arises which deals with security issues in requirement engineering phase and then take appropriate design decisions so that security mechanism used are optimal to some extent resulting in efficient secure system. Therefore the requirement engineers must discover security requirement along with functional and non functional requirement, so that security requirement can be dealt effectively. In this paper, we present a method for security requirement analysis and prioritization along with the other activities of security requirement engineering. Analysis is based on the technique of ontology that will automate the process and prioritization is based on risk analysis. The resultant system will be a cost effective in nature as well as it lay a foundation for further activities so that designer will adopt the most efficient technique for the implementation of security requirements.

Keywords: Security Engineering, Security Requirements Analysis, Security Requirements Prioritization, Ontology, Risk Analysis

1 Introduction

Security engineering is a field of engineering which focuses on the security aspect of a system here we are talking about software systems. The security goals are traditionally classified into confidentiality, integrity and availability of information in an organization [24]. In early approaches, the security measures are taken during design phase, which could result in unseen constraints which can affect the cost and availability of the system and sometime failure of system. In response to this Software engineering community has proposed different methodology to elicit security requirements and then enforce measures to meet these requirements. Some of the proposals for eliciting security requirements are abuse case [4], misuse case [1, 2, 10, 11], common criteria [3, 14] and attack trees [5]. Also, methodologies like secure tropos extension of tropos methodology [19], intentional anti model extension of KAOS methodology with security requirement oriented construct [22] are also discussed. Also, there are proposals for security engineering which takes the risk and threats into consideration and then apply measures to enforce security on threats like OCTAVE [17], CORAS [16], and CRAMM [18]. Firesmith [6, 7] has given proposal to classify security requirements and identify them to mitigate threats that causes risk for a system. Methods like EBIOS [20], MEHARI [21], etc. are proposed for risk assessment and management.

In our previous work, we have proposed software engineering framework that integrates security engineering activities [15, 26] in SDLC. The process starts with the process of elicitation of security requirements along with functional and non functional requirements. These elicited requirements must be analyzed and prioritized so that proper design decisions can be taken [26]. This paper will focus on techniques for analysis and prioritization of security requirement. Process of analysis is based on Ontology and prioritization will be done using risk analysis.

Ontology is a formal, explicit specification of a shared conceptualization. Ontology consists of various domain specific concepts, their properties and relationships between them. Ontology is a hierarchical arrangement of knowledge related to a domain which can be used as a centralized dictionary. The domain relevant to our discussion is security. The advantage of using ontology as a tool for requirements analysis is that it allows the requirements engineers to analyze the requirements with respect to the semantics of the domain. The requirements are generally discovered and documented in natural language. Natural language statements may have different interpretations by different human engineers trying to analyze them, hence making the analysis difficult. Our approach of using ontology will overcome this problem as it helps in semantic analysis of the requirements. Our approach will also provide automation of the analysis process and will save human effort.

Risk analysis is important as it would tell what could possibly go wrong? What is the likelihood of it happening? How will it affect the project? So risk analysis is considered as an important factor. Further paper is organized as follows section 2 provides activities of security requirements
engineering. Section 3 provides proposed framework for analysis and prioritization of security requirements; next section will provide a case study for illustration of proposed technique; and finally section 5 concludes the paper and provides the future scope for research.

2 Activities of Security Requirement Engineering

Security engineering framework proposed in our previous work consists of four main phase that are security requirements engineering, security design engineering, security implementation and security testing. Framework representing the overall procedure for Security Requirement Engineering is shown in Fig 1. Different activities in security requirement engineering are:

2.1 Requirements Discovery and Definition

Here our aim is to discover first the functional, non functional requirements and then security requirements which mitigates threats that affects the assets used by functional requirements. We have considered twelve types of security requirements as defined by Firesmith [6] and extended view point oriented method of Sommerville [12, 13] to define security requirements which are associated with functional requirements.

Different steps in this activity are:
- Identify various stakeholders of the system using view-point analysis [12, 13]. We have identified the various abstract classes of actors as direct and indirect actors. Direct actors are those who directly interact with the system such as human, software system and hardware devices. Indirect actors refer to Engineering personals who develop software and people who regulate application domain. For this paper our interest is in direct actor.

2.2 Analysis and Validation of the Requirements

Analysis is not an easy task it comes with lot of problems as said by Sommerville [13] due to change of requirements, stakeholder don’t know what they really want, etc. If requirements will change, it will in turn affect everything. So analysis and validation needs special attention in security requirement engineering.

Analyze the identified security requirements for consistency and completeness. If any conflict occurs it must be resolved and must reach to an agreement to avoid any further conflicts.

Completeness Checking ensures inclusion of all necessary security requirements identified to protect the assets of an organization that are affected by threats. Here we check whether all the identified threats are mitigated by security requirements or not.

Fig 1. Framework for Security Requirements Engineering
Consistency Checking ensures that there are no contradictory security requirements. The two security requirements should not conflict. It ensures that two security requirements never conflict. Example, customer wants to authorize more number of customers to access the confidential data but want the cost incurred in providing privacy for individual and organizational data will get decreased. So, both of the requirements can’t be satisfied simultaneously.

The steps involved in analysis of security requirements are discussed in the next section.

2.3 Prioritization of Security Requirements

Prioritize the security requirements so that it help the designers and other members involved in later phases to take design decisions. It tells about which security requirement is more critical and need to be dealt first over other. The proposal devised for prioritization of security requirements is given in the next section

2.4 Management of the Requirements

Keep trace of each security requirements and its associated attributes such as requirement identity, view point identity, functional requirement, nonfunctional requirements, threats design constraint, other security requirement, design constraints. The techniques for requirement management presented in [12, 13] can be used for this activity. Details of this work will be dealt in the future.

3 Proposed Techniques for analysis and Prioritization

This section will be divided into two sub sections. The first part will cover the approach for security requirements analysis using ontology. And the next part deals with the steps involved in prioritization of security requirements using the concept of risk analysis.

3.1 Security Requirements Analysis Framework

We propose a framework for the analysis of security requirements that are discovered during security requirements elicitation. The framework is represented in Fig 2. Now various steps involved in the framework will be discussed.

Steps Involved in Analysis:

- Creation of the Domain Ontology: This is a major task, as it will serve as the main knowledge base for analysis. Ontology can be created using any of the various approaches available at present such as Methontology [26], OTK (On-To-Knowledge) [27], etc. Ontology will be constructed in hybrid fashion out of top down, bottom up and hybrid approaches because in the hybrid fashion, initially the ontology is constructed by a domain expert and further modifications can be made in it when needed. Protégé [30] is used for ontology development. Reason for using Protégé as a tool is that it is intuitive, easy to use, interactive, easily scalable and extensible due to plug-ins. And OWL (Web Ontology Language) is used for creation of our ontology.

- Creation of the Application Ontology: Application ontology will represent the elicited security requirements and their relationships. From the given set of elicited security requirements, application ontology can be created through following steps:

![Fig 2: Ontology Based Security Requirements Analysis Framework](image-url)
i. Now classes will be converted into concepts of the ontology. As class consists of a class name, properties and relationships to other classes.

ii. Next properties of a class will become properties of corresponding concepts in ontology.

iii. The Subclass and Supersclass inheritance relation will be maintained in the ontology, using Subclass and Supersclass concepts.

iv. Other relations in the class diagram will be converted into properties in the ontology. For example: aggregation relation becomes PartOf property.

Various tools are available to automate the above task such as AToM3 [31], ATL [32], etc. tools can be used.

- **Mapping Application Ontology with Domain Ontology:** Ontology mapping module maps the classes and relationships in the application ontology with the corresponding classes and relationships of the domain ontology. The knowledge obtained through this mapping will serve as a knowledge base for the inference engine. The mapping process can be performed using existing ontology mapping algorithms [33]. The above tasks can be automated by tools such as OMEN [34], CROSI [35], etc.

- **Knowledge Extraction by the Inference Engine:** We can select any inference engine for querying and drawing out useful conclusions from the security requirements analysis process. In our proposal we are using the Prolog inference engine. Prolog Queries will be used to extract useful knowledge from the knowledge base. Here the knowledge base is the mapping result of the above step. For example, a Prolog query can be generated to obtain from the knowledge base, all the threats that can be mitigated by a particular security requirement.

- **Application of Completeness Checking Rules:** The following rules can be applied on the knowledge extracted by the inference engine for checking completeness of the security requirements:

  i. If a concept c1 is related to concept c2 by relation R and the same concept c1 is related to concept c3 by the same relationship R in the domain ontology, then add the concept c3 to the given set of defined requirements and relate it to c1 by relation R (if it is not already in application ontology.)

  ii. If c3 is already in the application ontology then just relate c3 and c1 with relation R.

  iii. Repeat the above check for all the concepts of the application ontology.

For example, if we have **identification** and **privacy** as security requirements and **snooping** as threat concept in the application ontology and **snooping** is related to **authentication** in the domain ontology with relation “is mitigated by”. Then **authentication** is added to our security requirements and it is related with **snooping** by “is mitigated by” relation.

- **Merging Application Ontology and Domain Ontology:** To merge the Application ontology with the Domain ontology we can use any existing ontology merging algorithm. The description of the algorithm is not given here because of the space constraints. Interested users can refer to [36]. Various tools such as Chimaera [37] and PROMPT [29] are available for automated merging of ontologies.

- **Consistency Checking:** The above merging process makes consistency checking very easy for the analyst. To check consistency of the security requirements we run the merged ontology on the Protégé platform. The recent versions of Protégé include Jese Inference Engine and Pellet reasoner [38]. The Jese Inference Engine automatically checks all the concepts for consistency and the Pellet reasoner automatically checks all the rules for consistency. The output of this step is the result phrase “consistent security requirements” or “inconsistent security requirements”.

### 3.2 Security Requirements Prioritization Framework

As we have the list of security requirements, it’s better to prioritize them. As prioritized list of security requirement would be helpful for designers and other members involved in later phases to take decisions. As it tells us about which security requirement is more critical and need to be dealt first over other.

The steps followed in security requirement prioritization are discussed in this section. Prioritization of security requirements will be done with the help of risk analysis. Risk is normally defined as the chance or likelihood of damage or loss [25, 39]. That is, it is a function of two separate components, the **likelihood** that an unwanted incident will occur and the **impact** that could result from the incident.

So risk can be calculated as :

\[
\text{Risk} = (\text{Probability of Occurrence of Threat} \times \text{Impact of Threat Occurrence on Assets}) \tag{1}
\]

**Probability of Occurrence of Threat = Threat Rating** and is represented by **TR**

**Impact of Threat Occurrence on Assets = Summation of importance level of assets affected by threat, as one threat may affect more than one asset.**
Represented by \( \sum_{k=1}^{n} IL \) where \( n \) is the maximum number of assets affected. Hence (1) will become

\[
\text{Risk} = TR \times \sum_{k=1}^{n} IL
\]

(2)

Steps involved in Security Requirement Prioritization

- Assign threat rating to each threat identified. Threat rating is the number which represent occurrence frequency of a threat in a system. The scale for threat rating is 1- 10. Assign lower value to threat whose occurrence frequency is low and higher to higher frequency threat.

- Assign assets a value in range of 1- 10, representing its importance to the organization. This importance level will show how much cost and resources are required to protect a particular asset.

- List various assets affected by each threat.

- Calculate the risk value using (2).

- Now the threats will be prioritized based on risk value calculated in step4. Higher the risk value higher will be the priority of threat.

- Finally prioritize security requirements based on threats priority. For prioritization of security requirement following rules will be used:

  i. If the security requirement is mitigating single threat then in that case threat priority is simply assigned to security requirement and it acts as its priority.

  ii. If the security requirement is mitigating more than one threat in that case we add up all the corresponding threat priorities and assign the calculated value to the security requirement as its priority.

4 Case Study

The whole process of security requirement engineering is explained here with the help of a case study of “Railway Reservation System”. It covers the full detail of the whole procedure that is from the identification of stakeholders to the final prioritization of security requirements.

- Various direct stakeholders or viewpoints involved are:
  i. Traveler
  ii. Railway Management
  iii. Database

- List of functional and non functional requirements is listed in Table1. And, various assets involved are:
  i. Traveler Information
  ii. Ticket Information
  iii. Credit Card/ Bank Details
  iv. IT Infrastructure (Communication Channel, System Information, etc)
  v. Employee and its Details

Various threats involved are identified using the concept of common criteria [3, 14]. We will be developing the repository of the threats. For extraction of threats from repository, one need to fill Actor Profiles that contains the seven fields as defined in [15]. All the threats identified are listed in Table1.

Various security requirements to mitigate threats are shown in Table 1.

Table 1. An Example “Railway Reservation System” Explaining our Process

<table>
<thead>
<tr>
<th>Viewpoints</th>
<th>Services</th>
<th>Non Functional Requirements</th>
<th>Threats</th>
<th>Security Requirements</th>
</tr>
</thead>
</table>

Table 2 Asset Importance Level

<table>
<thead>
<tr>
<th>ASSET</th>
<th>IMPORTANCE LEVEL (1-10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traveller Information</td>
<td>7</td>
</tr>
<tr>
<td>Ticket Information</td>
<td>5</td>
</tr>
<tr>
<td>Credit Card</td>
<td>9</td>
</tr>
<tr>
<td>IT Infrastructure</td>
<td>4</td>
</tr>
<tr>
<td>Employee and its Details</td>
<td>6</td>
</tr>
</tbody>
</table>
- Assigned threat rating values on the scale of 1 – 10 to various threats is given in Table 3.

Table 3 Threats with Threat Rating

<table>
<thead>
<tr>
<th>THREAT</th>
<th>THREAT RATING (1 – 10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change Data</td>
<td>9</td>
</tr>
<tr>
<td>Repudiate Receive</td>
<td>5</td>
</tr>
<tr>
<td>Spoofing</td>
<td>5</td>
</tr>
<tr>
<td>Insider</td>
<td>3</td>
</tr>
<tr>
<td>Privacy Violated</td>
<td>7</td>
</tr>
<tr>
<td>Outsider</td>
<td>6</td>
</tr>
<tr>
<td>Disclose Data</td>
<td>4</td>
</tr>
<tr>
<td>Social Engineer</td>
<td>6</td>
</tr>
<tr>
<td>Impersonate</td>
<td>8</td>
</tr>
</tbody>
</table>

- Identify the assets affected by each threat. The identified assets corresponding to each threat is given in Table 4.

Table 4 Assets Affected by Threat

<table>
<thead>
<tr>
<th>THREAT</th>
<th>ASSET THAT CAN BE AFFECTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change Data</td>
<td>Traveler Information, Ticket Information, Employee Details</td>
</tr>
<tr>
<td>Repudiate Receive</td>
<td>Credit Card Information</td>
</tr>
<tr>
<td>Spoofing</td>
<td>Credit Card Information</td>
</tr>
<tr>
<td>Insider</td>
<td>IT Infrastructure</td>
</tr>
<tr>
<td>Privacy Violated</td>
<td>Ticket Information, Credit Card Information, Traveler Information</td>
</tr>
<tr>
<td>Outsider</td>
<td>IT Infrastructure, Traveler Information, Employee Details</td>
</tr>
<tr>
<td>Disclose Data</td>
<td>Traveler Information, Ticket Information, Employee Details</td>
</tr>
<tr>
<td>Social Engineer</td>
<td>Traveler Information, Employee Details</td>
</tr>
<tr>
<td>Impersonate</td>
<td>Traveler Information, Employee Details</td>
</tr>
</tbody>
</table>

Final computation of priorities is given in Table 6. As Security Requirement Authorization is mitigating three threats so for its priority we add the priority value of corresponding threats. And, in case of Privacy Security Requirement it is mitigating only one threat so whatever will be the priority value of threat it is directly assigned to security requirement.

Table 6 Priority of Security Requirements

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Authorization</td>
<td>Change Data</td>
<td>8</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Disclose Data</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Insider</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Privacy</td>
<td>Privacy Violated</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Non repudation</td>
<td>Repudiate Receive</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Spoofing</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Integrity</td>
<td>Social Engineer</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Identification</td>
<td>Change Data</td>
<td>8</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Outsider</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Authentication</td>
<td>Privacy Violated</td>
<td>7</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>Outsider</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Security Auditing</td>
<td>Impersonate</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Intrusion Detection</td>
<td>Outsider</td>
<td>6</td>
<td>6</td>
</tr>
</tbody>
</table>

If in case two Security Requirements has same priority in that case designer will have to take decision which one to deal first. Throughout the computation higher number represent higher priority or higher value.
5 Conclusions and Future Work

In this paper we have presented techniques for analysis and prioritization of security requirements based on ontology and threat analysis respectively. This method is improvement over the method presented in [23], as it tries to quantify the value of risk value so that will get correct and consistent result. Further complexities in risk analysis are under processing which covers other factors then threat rating. Here analysis is done using the manual method of our previous paper [23] and will provide a detailed framework for it with proper example in our next paper. We are also developing a computer based tool to incorporate these techniques. Method block presented in [9] will be extended to incorporate the security characteristics also the CAME tool MERU [8] will be initiated in the construction of method which includes the security engineering.

6 References

[1] Alexander IF, “Modelling the interplay of conflicting goals with use and misuse cases”. In: Proceedings of the 8th international workshop on requirements engineering: foundation for software quality (REFSQ’02), Essen, Germany, 2002.
[18] The Logic behind CRAMM’s Assessment of Measures of Risk and Determination of Appropriate Countermeasures available at www.cramm.com
[38] Evren Sirin a, Bijan Parsia a, Bernardo Cuenca Grau a,b, Aditya Kalyanpur a, Yarden Katz Pellet: A Practical OWL-DL Reasoner.