A Goal Oriented Approach for Modeling and Analyzing Security Trade-offs

Lovepreet Kaur, Kailash Bahl
Computer Science and Engineering Department
Patiala Institute of Engineering & Technology for Women

ABSTRACT

In software system designing, security is only one major design factor among many. It may be more probable with other major factors such as working, usability, and performance. Security mechanisms like network protection and encryption are invaluable without explicit recognition of competing design objectives and their origins in stakeholder interests. Through this paper, we examine the explicit and systematic support provided by conceptual modeling for analyzing security trade-offs.

Keywords
Security Trade-offs, Trade-off Analysis, Goal Modeling, Goal Model Evaluation

1. INTRODUCTION

All software system requires increasingly significant amount of security. For example, if usability concerns are not addressed in the design of a secure system. Security safeguards may conflict with usability, performance, and users respond by circumventing security mechanisms. For designing successful secure software systems, goods balance maintains between security mechanism and usability goals. Security goals can have their own contradictions because confidentiality, integrity, privacy, accountability, availability, and recovery from security attacks often conflict fundamentally.

For example, accountability requires a strong audit trail and end-user authentication, which conflicts with privacy needs for user. Security is about balancing the trade-offs among the competing goals of multiple actors. In current practice, security designers often adopt security mechanisms such as firewalls, access control, or encryption without explicit recognition of, and systematic treatment of competing design objectives originating from various stakeholders. This motivated question arise that which conceptual modeling techniques can be more help. Full for designers to analyze security trade-offs to obtained valuable security?

2. Conceptual Modeling Criteria for Security Trade-offs Analysis

A good software design follows security trade-off to maintaining the right balance among many competing objectives. When some goals are not sufficiently satisfied, designers need to explore further alternatives that can better achieve those goals without detrimentally hurting others. A careful and systematic process for security trade-off analysis can be very challenging, because a wide range of security mechanisms, solutions and frameworks need to be considered.

There are three kind of concepts should be followed by conceptual modeling technique to support security trade-offs are as:

I). Goals: Security trade-offs are conflicts among design objectives that originate from stakeholder goals. The more fundamental problem is that designers need to decide about alternatives security mechanisms subject to multiple factors such as cost, time-to market, non-functional requirements (NFRs), security policies, standards, and individual goals of various stakeholders. The measures could be quantitative or qualitative. Quantitative approaches can greatly simplify decision making, but can be difficult to apply due to lack of agreed metrics or unavailability of accurate measures. The modeling technique should be able to support analysis despite incomplete knowledge about goals.

II). Actors: Design objectives typically come from multiple sources and stakeholders such as system’s users, administrators, top managers, project managers, and customers. The conceptual modeling technique should be able to model multiple actors that impose competing goals on the designer, and should provide means to trace back goals to the actors. The modeling technique should be able to model tradeoffs that occur within a single actor or across multiple actors.

iii) Security Specific Concepts: The conceptual modeling technique that enables security trade-off analysis should model security specific concepts such as threats, vulnerabilities, and safeguards. Threats can be viewed as malicious actors’ goals. The modeling technique should provide means to model to what extent attacks are successful, how attacks influence on goals.
3. Existing Approaches to Security Trade-off Analysis

Many approaches have been proposed to model security aspects of the software systems. The notion of “abuse case” [14] and UML sec modeling language [15] are examples of security specific conceptual modeling approaches for modeling security requirements and aspects of the system. Examples of such frameworks are KAOS [1], the NFR framework [10], the i* framework and tropos.

3.1 ATAM

Bass et al. [11] introduces a framework to model quality attributes and architectural options using the notion of scenarios and tactics respectively. The result of the analysis is an “Architectural Approach Analysis” table for each quality scenario.

3.2 SVDT/AORDD Approach

The SVDT approach using UML sec for modeling security solutions. A complementary framework on AORDD provides a risk assessment process and cross-benefit tradeoff analysis. AORDD and SVDT use BBN to compute Return on Security Investment.

Fig. 2 illustrates the relationship between the main concepts involved in AORDD risk assessment. This list, alternative security treatments, and fixed trade-off parameters such as budget, time-to-market, and policies are fed into the BBN to compute the ROSI.

3.3 Secure tropos \ i*

The proposed approaches in [3] take advantage of the i* and tropos frameworks. The meta-model of related concepts to the Tropos goal model, which is the core of all these approaches, is depicted in Fig. 3.

3.4 Limitations of Existing Approaches

In ATAM, trade-offs among quality scenarios and tactics in the “Architectural Approach Analysis” table are indirect and implicit, since trade-off and risk points, instead of referring to quality scenarios, refer to affected quality properties.

In SVDT and AORDD, the trade-off inputs and information are given to a BBN, and the final ROSI is computed automatically, which makes the analysis efficient. Table 1 summarizes a comparison of the studied approaches based on the evaluation criteria from section 2. A comparison of existing approaches based on the criteria of the conceptual modeling technique for security trade-off analysis.

<table>
<thead>
<tr>
<th>Method Requirement</th>
<th>ATAM</th>
<th>SVDT/AORDD</th>
<th>i*Tropos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goals</td>
<td>Expressed in terms of scenarios</td>
<td>Limited to security requirements and fixed BBN parameters</td>
<td>Explicit goals</td>
</tr>
<tr>
<td>Relations of goals</td>
<td>Not modeled explicitly</td>
<td>Limited to UMLsec models</td>
<td>Modeled using contribution links</td>
</tr>
<tr>
<td>extents of goal satisfaction</td>
<td>Not expressed</td>
<td>Quantitatively</td>
<td>Qualitatively</td>
</tr>
<tr>
<td>Goal contribution structure</td>
<td>Utility tree doesn’t capture the contributions of scenarios</td>
<td>Not modeled</td>
<td>Modeled in terms of sub-goals and contribution links</td>
</tr>
<tr>
<td>Multiple actors</td>
<td>Expressed implicitly by multiple stimuli sources</td>
<td>Not modeled</td>
<td>Modeled in terms of agents/actor roles/positions</td>
</tr>
<tr>
<td>trade-off with single actor or crossed actors</td>
<td>Single actor</td>
<td>Single actor</td>
<td>Single and multiple actors</td>
</tr>
<tr>
<td>Security Specific Trade-off Concepts</td>
<td>Not modeled</td>
<td>Some concepts are modeled</td>
<td>Some concepts are modeled</td>
</tr>
<tr>
<td>Trade-off analysis method</td>
<td>Qualitative analysis</td>
<td>Quantitative analysis</td>
<td>Qualitative and quantitative analysis</td>
</tr>
</tbody>
</table>

4. The Security Trade-offs Modeling Notation

We propose a meta-model of security concepts for systematically addressing security trade-offs (Fig. 4), considering the limitations of existing approaches and reviewing well known security knowledge sources such
as NIST’s guidelines. The core of the meta-model is the concepts of goals and actors guided by the criteria of the conceptual modeling technique that enables security trade-offs analysis.

The proposed notation builds upon the framework which provides a notation to model actors and competitions among the actors. Actors achieve goals on their own or depend on each other for goals to be achieved, tasks to be performed, and resources to be furnished.

4.1 Malicious Actor, Goals and Tasks
Actors depend on, or compete with each other to achieve their goals. Meanwhile, malicious actors try to achieve their own goals. Representing a malicious actor with a different modeling construct in i* was first employed in [3] by highlighting them with a black shadow rectangle.

4.2 Assets, Services and Vulnerabilities Points
An asset is anything that has a value for the organization [13]. Physical resources, information, and people can be counted as assets. In this way, the asset concept is well matched with the “resource” modeling element in i*. Assets can be the services an organization offer or receive, and in this case, can be represented by tasks or goals that actors offer to the “dependent” actors. In security analysis, a vulnerability point is any weakness in, or back door to the system [13].

4.3. Relation between Attacks and Security Mechanism
In the i* notation, relation between soft goals and other elements is modeled by contribution links [7]. If an element hurts a soft goal, yet not enough to prevent it, the contribution link type is “-”.

4.4. Expressing Trade-offs by the Proposed Conceptual Structure
The proposed approach provides the means to model goals, and trace them back to the source actors. In this approach, trade-offs among goals are modeled by contribution links. The i* notation offers the conceptual structure to model trade-offs between refined sub-goals of high level goals as well. It has negative influence on the Usability soft goal consequently. In this way, the trade-off among usability and security is modeled through relationships among their refined sub-goals.

5. Trade-off Analysis and Decision Making
In the first step, evaluator assumes that attackers are successful in performing tasks and satisfying their goals, since attackers are usually external actor that designer has no sure knowledge of their abilities and skills. A goals that operational security mechanism (step 7). This label indicates the evaluator’s judgment about the success of the actor in performing a security task or achieving a security goal. In step 9, the goal model indicates which goals are fully or partially satisfied or denied for the examined security solution.

6. Case Studies
In developing the proposed notation, we modeled a number of NIST guidelines. In addition to example cases, we applied the notation to three example cases originally used to illustrate other approaches to security trade-offs.

Table 2. Evaluation labels and propagation rules from [10]

<table>
<thead>
<tr>
<th>Child Node</th>
<th>Contribution Type (Prevent)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satisfied</td>
<td>+</td>
</tr>
<tr>
<td>Weakly Satisfied</td>
<td>/</td>
</tr>
<tr>
<td>Conflict</td>
<td></td>
</tr>
<tr>
<td>Unknown</td>
<td></td>
</tr>
<tr>
<td>Weakly Denied</td>
<td></td>
</tr>
<tr>
<td>Denied</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 7 gives a part of the trade-off models and analysis of GA system using the proposed approach in this paper. The goal model evaluation yields a fully satisfied Privacy goal with Confidentiality partially satisfied, while Performance is partially denied.

7. Conclusions and Future Work
In this paper, we began by considering the criteria for a conceptual modeling technique that enables designers to model and analyze security trade-offs among competing
goals of multiple actors to achieve a good-enough security level. We studied existing approaches to trade-off analysis, and identified limitations of these approaches. Based on the evaluation criteria and limitation of previous works, we proposed extensions to the $^*$ notation.

8. References
2. Castro, J., Kolp, M., Mylopoulos, J., A requirements-driven development methodology, In Proc. of the 13th Int. Conf. on Advanced Information Systems Engineering, CAiSE'01 (2001)