Applying a requirement engineering based approach to evaluate the security requirements engineering methodologies

Sravani Teja Bulusu, Romain Laborde, Ahmad Samer Wazan, François Barrère, Abdelmalek Benzekri

Applying a Requirement Engineering Based Approach to Evaluate the Security Requirements Engineering Methodologies

Sravani Teja Bulusu, Romain Laborde, Ahmad Samer Wazan, Francois Barrere, Abdelmalek Benzekri
IRIT/University Paul Sabatier, 118, Route de Narbonne, 31062, Toulouse, France
sbulusu@irit.fr, laborde@irit.fr, ahmad-samer.wazan@irit.fr, Francois.barrere@irit.fr, Abdelmalek.benzekri@irit.fr

ABSTRACT

Considering the multitude of security requirements engineering methodologies available today, selecting a security requirement engineering methodology that fits the security engineering context becomes a promising task. In previous work, we outlined a generic evaluation methodology to elicit and evaluate the anticipated characteristics of a security requirements engineering methodology according to the stakeholders' working context. In this paper, we detail each step of our methodology using an example context of network security requirements engineering.

1 INTRODUCTION

Security requirements engineering is an important activity since bad security requirements can lead to ineffective security or worth security holes. However, given the plethora of security requirements engineering methodologies available today, choosing a good SRE methodology is still a complex task. Many comparative and evaluation studies exist. However, they are not reusable due to various issues such as: ad-hoc criteria, lack of consideration of all the phases of RE process; and finally non-consideration of the working context of the security requirement engineers.

To help address these issues, in our previous paper[1], we proposed an evaluation methodology built on the classical idea of requirements engineering approach. Figure 1 depicts an overview of our approach. It includes three main steps: 1) identifying the problem context and eliciting initial high-level characteristic goals. This is done by coupling the stakeholder's working SRE context as well as the quality criteria of good security requirements; 2) refining the high-level characteristic goals into final requirements of the SRE methodology-to-be (R^M); 3) the final step deals with evaluation of the existing SRE methodologies using the elicited requirements (R^M).

Our discussion in the previous paper focused mainly on outlining the generic idea of our evaluation methodology. In this paper, we will explain how our methodology is applied to the SRE context of the stakeholders. More specifically, we emphasize our discussion on how goal modelling approach is used to elicit and refine the characteristic goals at step1 and step2. For SRE context, we consider an example use case related to the, working context of the security experts involved in network security engineering process. In end, we will briefly explain step3 with some sample evaluation results.

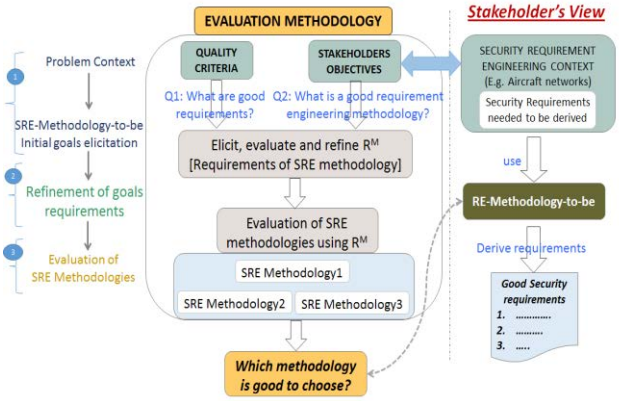

Figure 1: Our evaluation methodology

The rest of the paper is structured as follows. Section 2 introduces the SRE context of our evaluation. In Section 3 we discuss the application of our evaluation methodology to the SRE context considered. Finally, we conclude our work in Section 4.

2 Scenario context

The scenario concerns a situation related to the maintenance of the aircraft to anticipate the health of the on-board aircraft system by verifying specific parameters of the On-board aircraft system (Figure 2).

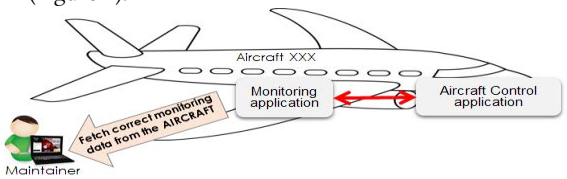

Figure 2: Example scenario context

The On-board aircraft system is integrated the aircraft monitoring application and the aircraft control application that are connected to each other via an internal avionic bus network. The maintenance people are allowed to connect their laptops to the monitoring application in order to fetch the monitored parameters. The security goals are expressed in terms of

protecting the integrity and availability of the monitored parameters.

Hence, the aircraft network security design must ensure a trusted transmission of parameters from the aircraft control application to the monitoring application and then to the laptop of maintenance people. In addition to these high-level security requirements, network design choices are dependent to more technical security requirements. E.g., maintenance people can potentially connect to the aircraft using an Ethernet cable or a wireless connection. This scenario gathers network security requirements context information in an unstructured format. It provides some insights on what kind of network security requirements can be elicited. However, the question of SRE methodology goodness from the point of view of the security experts is still open. Without this information it will difficult to anticipate what kind of SRE methodology would be interesting to the security experts.

3 Application of our methodology

3.1 Step1: Problem context and initial requirement analysis

The initial step of our approach allows analysing the security problem context of the example scenario given in section 2. Accordingly, this step deals with interviewing the people involved in the security engineering process. We have developed an elicitation tools using a consolidated list of 20 characteristics of good security requirements provided in our previous paper[2], see Figure 3. The first three columns contain a unique identifier, a quick one-line definition and corresponding synonyms found in the literature. The last column describes the quality criteria via a set of questions, each reflecting different perspectives of the respective criterion definition. This elicitation tool facilitates to trigger the discussions. It acts as a common platform to discuss as well as to obtain common understanding of their perspectives and expectations.

No	Abstract criterion abstract definition	Criterion Names in use	QUESTIONAIRES To what extent does the SRE methodology facilitate?
C2	Compatible, non-contradictory requirements	Consistent	1) Does the SRE methodology allow to verify the conflicts between the requirements, goals, assumptions and the domain properties?
C3	Accomplishable within the given financial, time, legal, technical constraints	Feasible/affordable legal Achievable	1) Does it allow to capture all the constraints pertinent to a security requirement? such as technical? Legal? Time? Financial? And time and costs of the implementation? 3) How long does it take to learn the methodology? Is it within the time constraints? 4) what are the training costs? do they exceed the financial constraints?

Figure 3: Sample of the elicitation tool

At the end of step1, every quality criteria of the elicitation tool must have been analysed and their respective interpretations should be agreed. We used KAOS goal modelling notation to represent the SRE-methodology-to-be goal refinement hierarchy. In Figure 4, the high-level goals are the quality criteria from our elicitation tool (goals in orange). The elicited interpretations are the immediate sub-goals (in green). We used different colouring to the goal nodes to facilitate the understanding of the readers; these colours are not compliant with KAOS.

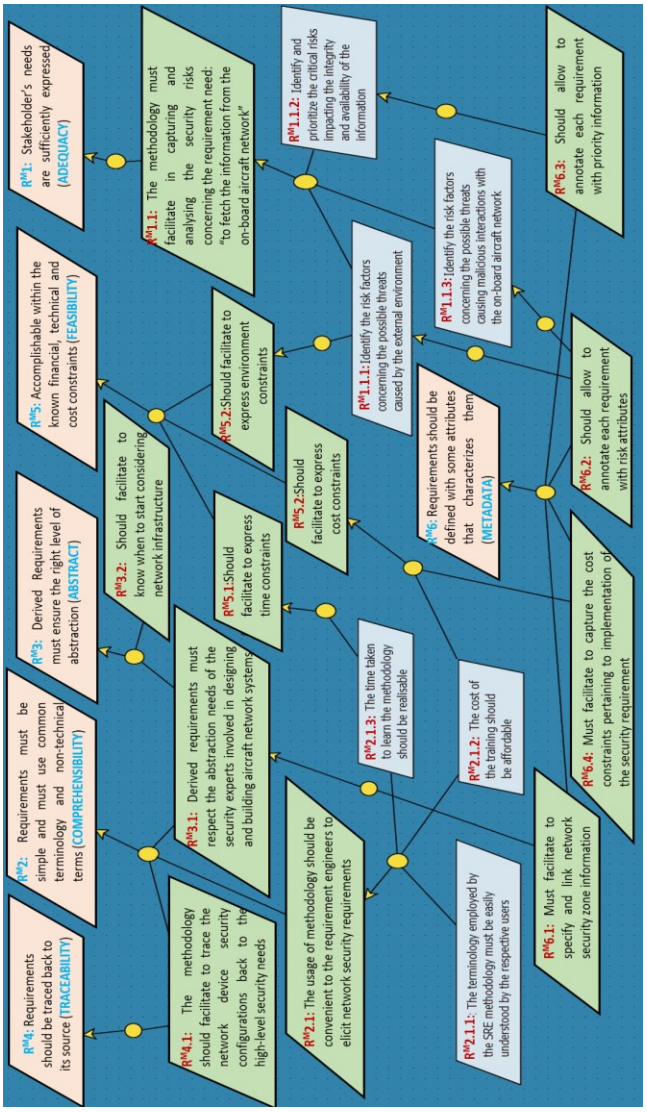

Figure 4: Sample of the SRE-methodology-to-be goals refinement

3.2 Step2: Refinement requirement analysis

In this step2 the high-level abstract goals realized in step 1 are coarse-grained into refined sub-goals fitting to specific demands of the security experts. The refinement is expressed using AND-construct. In some cases it is possible that a sub-goal can be refined from multiple high-level goals. E.g. $\mathbf{R^M3}$ can be derived from $\mathbf{R^M3}$ and $\mathbf{R^M2}$ with a justification stating that the requirements not respecting the abstraction requirements of the stakeholders are not comprehensible. This type of refinement patterns explains the semantic dependencies between the quality characteristics. It also explicitly reflects the merging of the different security experts' points of views. The refinement is

performed until the final refined goals are realized as verifiable objectively. Only then the final sub-goals (leaf nodes) qualify as the evaluation criteria. However, the type of verification method and expected performance metrics differs with respect to the type of evaluation criteria. For instance, let's consider the evaluation criterion $R^M6.2$ It states "the SRE methodology-to-be allows to annotate each requirements with risk attributes". The verification method must facilitate to evaluate the supportability of the SRE-methodology-to-be in capturing risk attributes related to environmental constraints and interaction dependency constraints, risk priority information. Respectively, the performance metrics to measure the supportability of this criterion is defined as in Table 1:

Table 1: Verification method for $R^M6.2$

Verification method	Performance measure
Requirement cannot be annotated with any risk information	nil
Requirements can be annotated with at least one of the attributes	low
Requirements can be annotated with risk priority and threat events	medium
The annotation feature is extensible. Requirements can be annotated with any risk attributes.	high

Finally, our approach has showed its benefits. We found new evaluation criteria (namely $R^M3.2$, $R^M6.1$ and $R^M6.4$) that were related to network security requirements engineering context.

3.1 Step3: Evaluation of the SRE methodologies

We tested the evaluation criteria on three distinct SRE methodologies Secure KAOS (a goal-oriented methodology – noted KAOS) [3], Secure Socio-Technical System (an agent-oriented methodology – noted STS) [4] and Security Engineering Process using Patterns (a problem-oriented methodology – noted SEPP) [5]. We derived security requirements for the example scenario given in section 2. During this practical experiment, we analyzed the supportability of SRE methodologies using the verification methods for each of the evaluation criteria. At the end of our evaluation study, we found that there are some criteria that are not supported by any of the above three methodologies, see Table 2. The criteria $R^M3.2$, $R^M6.1$ and $R^M6.4$ are notably related to the network security requirements engineering context. The evaluation measure we used is qualitative and defined as *high* for highly supportable, *medium* for partially supportable, *low* for less likely supportable and *nil* for not supportable. Due to the limitation of space we have not included the detailed discussion on the performance of the SRE methodologies and the evaluation results.

Table 2: Sample of the evaluation results

Evaluation criteria list (R^M)	STS	Secure KAOS	SEPP
$R^M3.2$: Should facilitate to know when to start considering network infrastructure	nil	nil	nil
$R^M6.1$: Must facilitate to specify and link network security zone information	nil	nil	nil
$R^M6.4$: Must facilitate to capture the cost constraints pertaining to implementation of the security requirement	nil	nil	nil

4 Conclusion and Future Perspectives

In this paper, we discussed the three steps of our evaluation methodology and applied to the SRE context of network security engineering. We used KAOS goal modelling notation to express the refinement of the characteristic goals of SRE methodology. The leaf nodes will eventually become the evaluation criteria. We have tested the criteria to evaluate three SRE methodologies (KAOS, STS and SEPP). For future work, we would like to apply our evaluation approach to other security engineering contexts. This will help us to determine which SRE methodology evaluation criteria are common and which are specific to security context.

Acknowledgement

This work is part of project IREHDO2 funded by DGA/DGAC. The authors thank all the security experts at Airbus and the anonymous reviewers for providing their useful comments.

REFERENCES

[1] S. T. Bulusu, R. Laborde, F. Barrère, A. Benzekri, and A. samer Wazan, 'Which Security Requirements Engineering Methodology Should I Choose? Towards a Requirements Engineering-based Evaluation Approach', presented at the ARES'2017, Reggio Calabria, ITALY, 2017.

[2] S. T. Bulusu, R. Laborde, F. Barrère, A. Benzekri, and A. samer Wazan, 'Towards the weaving of the characteristics of good security requirements', in *CRISIS 2016*, Roscoff, France, 2017.

[3] A. van Lamsweerde, 'Elaborating security requirements by construction of intentional anti-models', in *26th International Conference on Software Engineering, 2004. ICSE 2004. Proceedings*, 2004, pp. 148–157.

[4] E. Paja, F. Dalpiaz, and P. Giorgini, 'Sts-tool: Security requirements engineering for socio-technical systems', in *Engineering Secure Future Internet Services and Systems*, Springer, 2014, pp. 65–96.

[5] D. Hatebur, M. Heisel, and H. Schmidt, 'A pattern system for security requirements engineering', in *Availability, Reliability and Security, 2007. ARES 2007. The Second International Conference on*, 2007, pp. 356–365.