Towards Empirical Evaluation of Automated Risk Assessment Methods*

Olga Gadyatskaya[1], Katsiaryna Labunets[2], and Federica Paci[3]

[1] SnT, University of Luxembourg, Luxembourg
olga.gadyatskaya@uni.lu
[2] DISI, University of Trento, Italy
katsiaryna.labunets@unitn.it
[3] ECS, University of Southampton, UK
f.m.paci@soton.ac.uk

Abstract. Security risk assessment methods are numerous, and it might be confusing for organizations to select one. Researchers have conducted empirical studies with established methods in order to find factors that influence their effectiveness and ease of use. In this paper we evaluate the recent TREsPASS semi-automated risk assessment method with respect to the factors identified as critical in several controlled experiments. We also argue that automation of risk assessment raises new research questions that need to be thoroughly investigated in future empirical studies.

Keywords: Security risk assessment, empirical studies, TREsPASS, CORAS

1 Introduction

Security risk assessment (SRA) is an integral part of operations in many companies. A recent report by PWC states that as many as 91% of surveyed companies have adopted a risk-based cybersecurity framework, often based on guidelines provided in ISO 27001 and NIST Cybersecurity Framework [22]. Risk assessment methodologies are the core part of such risk-based cybersecurity frameworks, as they allow to identify, prioritize, mitigate and communicate security risks. Yet, the sheer variety of existing security risk assessment methodologies makes it difficult for organizations to understand which methodology is more beneficial in their context. Thus, the security community recently started to pay more attention to empirical studies of risk assessment methodologies, in order to discover what are the benefits and drawbacks of existing methods, and to provide guidelines to CISOs on what kinds of approaches provide better results. Typically, these empirical studies take the form of controlled experiments, in which participants apply the investigated methods, or their aspects, to realistic scenarios, and researchers observe these exercises and evaluate the outcomes [24, 10, 18, 12, 14, 13, 15, 2, 25, 27].

* This work was partially supported by the European Commission under grant agreement n. 318003 (TREsPASS) and by the SESAR JU WPE under contract 12-120610-C12 (EMFASE).

The body of knowledge already accumulated from these experiments can, and should, be considered when new methods are designed. Furthermore, emerging SRA methods can be immediately evaluated based on the criteria identified as important in empirical studies. Moreover, recent advances in automation bring about new types of SRA methods that aim to identify, prioritize and treat security risks in (semi-) automated manner. We argue that these next-generation approaches pose new challenges to method designers, which should be empirically investigated in controlled experiments.

In this paper we benchmark[4] the recent TREsPASS socio-technical risk assessment method [26] with the established CORAS method [1] based on the criteria identified in previous empirical studies [15, 12]. Furthermore, we outline the challenges that the emerging type of automated risk assessment methods poses, and ponder about research questions that can be investigated in future empirical studies with these new methods.

2 Criteria for Security Risk Assessment Methodologies

Labunets et al. [15, 14, 12, 2] have conducted a series of controlled experiments to investigate which are the main features of SRA methods that are behind the method's success. Success of a method is typically measured according to its *actual efficacy* in identifying threats and security controls and the *perceived efficacy* that participants have of the method, e.g. if they find the method easy to use or useful [19]. To identify the features, Labunets et al. have applied qualitative analysis techniques from grounded theory to the interviews conducted with participants during the experiments. Four main features were identified that can determine the actual success of an SRA method.

Clear Process. Clear process means that the steps to identify assets, threats and security controls are well-defined and guidelines on how to apply the steps are provided to the analysts. If an SRA method has clear process, this positively affects the actual effectiveness of the method and the perception that the analysts have of the method. On the contrary, if the analysts do not know how a step of the process should be executed, the method will not be effective and will not be perceived as easy to use.

Visualization of risk models. Risk model visualization gives an overview of results of SRA, and thus may have a positive impact on an SRA methods success. However, if the visual notation does not scale for complex scenarios, it no longer provides a big picture of the risks threatening the target of analysis, and therefore it negatively affects the methods' effectiveness and perception.

Catalogues of threats and security controls. Catalogues can facilitate the identification of threats and controls especially for the analysts with limited security knowledge. As reported in [2], domain experts without security expertise using domain-specific catalogues achieve better results than domain and security experts. Finding, sharing and validating threats and controls with catalogues is

[4] Notice that in this paper the evaluation was performed by the authors. No controlled experiments with the TREsPASS method were executed yet.

more efficient and effective, and, thus, the actual and perceived efficacy of an SRA method is higher.

Tool support. Tool can automatize the execution of an SRA process (e.g., computation of risk level) or can facilitate reporting of the results using an appropriate format (e.g. provide a set of tables that match methods steps). A well-designed tool can thus have a positive effect on methods success. In contrast, a primitive or buggy tool can only have a negative impact on the analysts perception of the method.

In addition to these main features, other important factors identified were:

Help in identifying threats and controls. Even if catalogues may not be included by default, the analysts appreciate if the methodology supports brainstorming and communication, and helps to elicit relevant threats and controls.

Change management and evolution support for SRA elements. The analysts appreciate if the method helps to ensure consistency across SRA elements (e.g. via traceability) when changes are introduced or the system evolves. This is especially important when dealing with large or evolving systems.

Scalability. For visual methods, such as CORAS and TREsPASS, scalability of diagrams becomes a challenge that, if not handled, can worsen method's effectiveness and perception.

3 CORAS Evaluation Findings

CORAS is an established model-driven risk analysis approach based on the ISO 31000 standard on risk management [1, 17]. It offers a customised language for threat and risk modelling and guidelines on how to use the language. Furthermore, CORAS provides a software tool to be used together with the CORAS method [1]. The CORAS process consists of eight different steps, where the first four steps focus on context establishment and the last four steps are about risk identification, estimation, evaluation and possible risk treatments. The CORAS modelling language defines four kinds of diagrams (asset, threat, risk and treatment diagrams) as part of its model-based approach to support visualisation in all steps of the process. A detailed description of the CORAS steps can be found in [1] and [17, Chap. 3].

With respect to the criteria listed in the previous section, controlled experiments with CORAS have resulted in the following conclusions [12, 15].

Clear Process. The participants found the CORAS process to be clear and easy to use: "good methodology, not difficult to use. It is much clear to understand the security case there" in [12]. CORAS provides different types of diagrams that help practitioners to model the system and possible attack scenarios. However, some participants regarded that CORAS has redundant steps: "I think CORAS has some duplications" in [15].

Visualization of risk models. CORAS enables a visual overview of the assets, possible sources of threats, threat scenarios and security controls, and helps the analysts to check that nothing has been overlooked: "diagrams are

useful. You have an overview of the possible threat scenarios and you can find links among the scenarios" in [12].

Catalogues of threats and security controls. CORAS does not include catalogues of threats and security controls. However, it can be used together with existing catalogues, e.g., BSI IT-Grundschutz or NIST-800-53.

Tool support. CORAS is supported by a diagram editor that helps to draw CORAS diagrams. However, the participants reported that the tool had low usability and was poorly developed. Thus, some of the participants acknowledged that they switched to an alternative solution for diagram drawing due to issues in using the tool [15].

Help in identifying threats and controls. CORAS threat and treatment diagrams support the analysts in brainstorming threats and security controls.

Change management and evolution support for SRA elements. Once the analysis with CORAS is over, it can be hard for the analyst to update created diagrams. There is no traceability between diagrams in the tool. The participants in [15] reported that in the CORAS tool "objects have no references between the diagrams. Changes on an object in a diagram are not reflected on the same object in other diagrams". Manual changes are time consuming and should be done carefully as it may affect many different diagrams.

Scalability. The participants of studies [12, 15] found the scalability issue to be relevant for CORAS: "these diagrams are getting soon very huge and very complex" in [15].

4 Evaluation of TREsPASS

TREsPASS. We start by briefly introducing the TREsPASS approach, which has recently emerged as a more automated methodology for risk assessment. The TREsPASS toolset assists a security analyst in finding attacks and ranking them [26, 21]. The core phases of the TREsPASS approach are preparation, analysis, and assessment [21, 26]. In the **preparation** phase the analyst gathers company- and sector specific data, and populates the knowledge base with relevant information (e.g. probability of employees to fall victim of a phishing attack) and attack scenarios (for example, a social-engineering attack on an employee expressed as an attack tree). She may also perform exploratory sessions with the stakeholders to understand their most pressing needs and the context (TREsPASS offers exploratory modelling sessions using Lego [21]). In the **analysis** phase, which takes advantage of the automatization, the analyst together with stakeholders designs a socio-technical model of the company, which could be at several layers of granularity: from a satellite view (only core infrastructure elements and assets) to a detailed view comprising employees, servers, virtual machines, relevant files, etc. The analyst and the stakeholders will then identify relevant abstract attack scenarios (expected attacker profiles and assets that could be compromized). Afterwards, the TREsPASS toolset generates a set of concrete threat scenarios represented as attack trees [9, 4], which are then extended and annotated with data using a knowledge base populated at the

preparation phase. The extended and annotated trees are then analyzed to identify critical attack scenarios [6], which are traced back to socio-technical model elements affected and visualized to the stakeholders. Finally, in the **assessment** phase, the analyst with the stakeholders can brainstorm on risk treatment elements (which security controls can be implemented to eliminate the threats) or decide to repeat the analysis with another set of scenarios or with a redesigned model.

Evaluation. We now evaluate the TREsPASS methodology based on the criteria listed in Sec. 2 and identify research questions to be investigated in follow-up experiments.

Clear process. The studies [12, 15, 14] mainly included novices in particular SRA methods (but not in information security), thus, the requirement of method and process clarity refers more to the question whether it is easy to master the method, than to whether a seasoned professional is able to achieve with it better results than with another method also familiar to him. For TREsPASS it is currently not known how steep is its learning curve or how easy it is to apply the method in the field.

Recommendation. We recommend to conduct controlled experiments to evaluate how comprehensible is the TREsPASS process to novices in the method.

Visualization of risk models. The TREsPASS toolset supports hierarchical visualization of the system model and advanced visualization of attack scenarios (as paths on the system model, as well as attack trees) [16].

Recommendation. There are ongoing efforts to evaluate the TREsPASS visualization capabilities with security practitioners [7]. These can be further strengthened by conducting ethnological studies with security analysts using the TREsPASS method for actual SRA tasks and capturing their reflections on the visualizations.

Catalogues of threats and security controls. In TREsPASS the role of catalogues is played by the knowledge base incorporating databases with relevant data, attacker profiles, and attack pattern library (tree banks). Thus, TREsPASS provides (limited) support for using existing knowledge in risk assessment.

Recommendation. Established catalogues of threats and controls (e.g., BSI IT-Grundschutz or NIST-800-53 catalogues) can be incorporated in the TREsPASS tool, as a part of the knowledge base. Introduction of catalogues can be also useful for automating controls selection and attack scenarios suggestion [5].

Tool support. The TREsPASS methodology is supported by the TREsPASS toolset that provides great support to the security analyst, as it automates some steps in risk assessment, as well as visualizes attack scenarios and the organization model. Yet, as studies [12, 15, 14] reported, tools should not hinder the work of the analyst, and a buggy or unreliable tools may worsen the risk assessment results. Quality of the TREsPASS toolset has not yet been independently evaluated.

Recommendation. The TREsPASS toolset can be empirically evaluated.

Help in identifying threats and controls. One of the main features of TREsPASS is to automatically find attack scenarios based on the system model.

This is of great value to the practitioners, as their workload is significantly reduced [5]. However, currently TREsPASS does not yet include automated selection of security controls, or automated attack scenario identification.

Recommendation. The TREsPASS method can be further improved by introducing automated suggestions for preferable security controls and relevant scenarios (assets, high-level attack goals and attacker profiles).

Change management and evolution support for SRA elements. Risk assessment artifacts, such as threat diagrams in case of CORAS or system models in case of TREsPASS, are not stable but often need to be modified, e.g., when some earlier mistake or wrong assumption is identified. As both CORAS and TREsPASS are model-based, and they rely on model transformations as a part of their processes, change management is crucial. In this respect, TREsPASS includes some evolution support, but not very advanced. Since the attack generation part is automatic, if the organization model is changed or the considered attack scenario is revised, the discovered attack paths and analysis of those will be automatically re-computed. The identified critical attack paths will be mapped back to the organization model. Therefore, minor changes can be accommodated in the TREsPASS process seamlessly to the analyst. Yet, evolution support can still be improved by maintaining more explicit traceability links among different underlying models and by improving the change management, e.g., via maintaining logs of changes.

Recommendation. The TREsPASS toolset can be enhanced by improving the change management capabilities following, for instance, the suggestions outlined in [3] for security modelling artifacts.

Scalability. The TREsPASS tool adopts a scalable visualization approach, as it is able to zoom in and out the organization model. Moreover, the visualization of attacks is also scalable, as the analyst is presented with not a full generated attack tree, but with only the most important its parts (the zoom out feature for attack trees), or even only the critical attack paths, which are laid down in the organization model. Thus, the TREsPASS methodology, in principle, is able to deal with large use cases. The only critical point still to investigate is design of fine-grained organization models, when the analyst together with the stakeholders need to introduce many intricate details of the organization, while being able to keep track of all different bits and pieces.

Recommendation. The TREsPASS toolset and methodology need to be empirically evaluated on large case studies, in order to understand whether design of fine-grained socio-technical models is scalable.

Summary. Both CORAS and TREsPASS are visual methods and they have comparable processes, therefore it is possible to draw conclusions about these methods based on the criteria identified. Following the results of empirical studies with CORAS [12, 15, 14], and the evaluation of TREsPASS done above, we can summarize that the automation introduced in TREsPASS contributes to improvement of such features as *scalability, help in identifying threats and controls*, and *change management*. Furthermore, TREsPASS seems to provide better support for *catalogues of threats and controls*, because it incorporates the knowledge

base comprising threat-relevant information. However, TREsPASS could be further improved by ensuring integration with established catalogues in order to support controls selection.

TREsPASS and CORAS are comparable in their support for *visualization of risk models*. A comparative study can be organized to assess which visualization approach is better comprehensible and thus more suitable for communication with the client. For the other important factors, i.e., *clear process* and *tool support*, they can be evaluated for TREsPASS only in practical setting (by running case studies, or, better, controlled experiments). Thus, for these two factors, it is currently not possible to compare TREsPASS with CORAS.

5 Discussion

The emerging class of automated and assisted risk assessment methods calls for new types of research questions to be posed about these approaches. Evidently, the usability dimension needs to be explored more in deep. *Tool quality* becomes of the utmost importance for the success of a new automated method. We can propose to apply extended approaches of usability studies from the Human-Computer Interaction domain [8], especially usability studies focused on systems security in this domain [23]. These approaches will allow to evaluate how usable are the different components responsible for various steps in the risk assessment process, how well-balanced is their interplay, and what particular features are exemplary or can be further improved.

At the same time, one important question to answer now is *What is the ideal balance of human expertise and tool support?* in a risk assessment method. *Would well-thought and automated security catalogues or patterns be able to compensate for involvement of domain experts, or analyst's lack of security knowledge?* is an immediate question. Controlled experiments involving experts performing risk assessment manually versus students exercising elaborate tools will not be able to answer this question. We need first to perform exploratory studies of currently established risk assessment techniques and best practices (e.g., interviews with security managers as in [20]), in order to evaluate what can and cannot be successfully automated without new breakthroughs in artificial intelligence.

Another dimension that needs to be taken into account by new studies is comprehensibility and usefulness of the risk assessment results. *Do the stakeholders understand and trust the outcomes and recommendations presented by an automated tool?* Indeed, risk assessment methods are considered to be an important communication means among security managers and the decision makers [22]. Interaction among different stakeholder groups in the process ensures that everybody shares the same view and has better situational awareness. Thus, automating risk assessment should not remove this communication channel, and the recommendations proposed by the tool should be justified and explained *in context*. Therefore, to have a better view on this dimension, we call for new empirical studies on comprehensibility of risk models and risk treatment recom-

mendations. Our own ongoing work is focused on comprehensibility of different risk modelling approaches [11].

6 Conclusions

In this paper we have reviewed the results of empirical studies with SRA methodologies [12, 15] and we have evaluated the recent TREsPASS risk assessment method on these criteria. Our main conclusions from comparison with the results for CORAS obtained in controlled experiments is that automation directly improves such important factors as *help in identifying threats* by including a knowledge base, and *scalability of visual risk models*. To understand better how TREsPASS fares in the *clear process* and *tool support* factors we need to have more results from practical exercises with this methodology.

Automation in risk assessment raises new research questions, which can be answered by conducting empirical studies with the emerging methodologies, by applying user study techniques from the Human-Computer Interaction domain, and by conducting ethnographical studies in the security risk assessment community. We hope that we can start a discussion on this topic that will result in better awareness in the security research community, and ultimately in better security posture of organizations through more thorough security risk assessment.

References

1. CORAS: `http://coras.sourceforge.net/` (2016)
2. De Gramatica, M., Labunets, K., Massacci, F., Paci, F., Tedeschi, A.: The role of catalogues of threats and security controls in security risk assessment: An empirical study with ATM professionals. In: Proc. of REFSQ. LNCS, vol. 9013, pp. 98–114. Springer (2015)
3. Felderer, M., Katt, B., Kalb, P., Jurjens, J., Ochoa, M., Paci, F., Tran, L.M.S., Tun, T.T., Yskout, K., Scandariato, R., Piessens, F., Vanoverberghe, D., Fourneret, E., Gander, M., B.Solhaug, Breu, R.: Evolution of security engineering artifacts: A state of the art survey. Int. J. of Secure Soft. Engineering 5 (2014)
4. Gadyatskaya, O.: How to generate security cameras: Towards defence generation for socio-technical systems. In: Proc. of GraMSec. LNCS, vol. 9390. Springer (2015)
5. Gadyatskaya, O., Harpes, C., Mauw, S., Muller, C., Muller, S.: Bridging two worlds: Reconciling practical risk assessment methodologies with theory of attack trees. In: Proc. of GraMSec. LNCS, Springer (2016)
6. Gadyatskaya, O., Jhawar, R., Kordy, P., Lounis, K., Mauw, S., Trujillo-Rasua, R.: Attack trees for practical security assessment: Ranking of attack scenarios with ADTool 2.0. In: Proc. of QEST. LNCS, vol. 9826. Springer (2016)
7. Hall, P., Coles-Kemp, L., Heath, C.: Visualisation in cyber-security: Towards a critical practice. In: Proc. of Electronic Visualisation and the Arts Australasia (EVAA) (2016)
8. Helander, M.G.: Handbook of human-computer interaction. Elsevier (2014)
9. Ivanova, M.G., Probst, C.W., Hansen, R.R., Kammuller, F.: Transforming graphical system models to graphical attack models. In: Proc. of GraMSec. LNCS, vol. 9390. Springer (2015)

10. Karpati, P., Redda, Y., Opdahl, A.L., Sindre, G.: Comparing attack trees and misuse cases in an industrial setting. Inf. and Soft. Technology 56(3), 294–308 (2014)
11. Labunets, K., Li, Y., Massacci, F., Paci, F., Ragosta, M., Solhaug, B., Stølen, K., Tedeschi, A.: Preliminary Experiments on the Relative Comprehensibility of Tabular and Graphical Risk Models. In: SESAR Innovation Days (2015)
12. Labunets, K., Massacci, F., Paci, F., et al.: An experimental comparison of two risk-based security methods. In: Proc. of ESEM. pp. 163–172. IEEE (2013)
13. Labunets, K., Paci, F., Massacci, F.: Which security catalogue is better for novices? In: Proc. of EmpiRE. pp. 25–32 (2015)
14. Labunets, K., Paci, F., Massacci, F., Ragosta, M., Solhaug, B.: A first empirical evaluation framework for security risk assessment methods in the atm domain. In: SESAR Innovation Days (2014)
15. Labunets, K., Paci, F., Massacci, F., Ruprai, R.: An experiment on comparing textual vs. visual industrial methods for security risk assessment. In: Proc. of EmpiRE. pp. 28–35. IEEE (2014)
16. Li, E., Barendse, J., Brodbeck, F., , Tanner, A.: From A to Z: Developing a visual vocabulary for information security threat visualisation. In: Proc. of GraMSec. Springer (2016)
17. Lund, M.S., Solhaug, B., Stølen, K.: Model-Driven Risk Analysis - The CORAS Approach. Springer (2011)
18. Massacci, F., Paci, F.: How to select a security requirements method? A comparative study with students and practitioners. In: Secure IT Systems, pp. 89–104. Springer (2012)
19. Moody, D.L.: The method evaluation model: a theoretical model for validating information systems design methods. In: In Proc. of ECIS. pp. 1327–1336 (2003)
20. Pettigrew III, J.A., Ryan, J.J.: Making successful security decisions: A qualitative evaluation. IEEE Security & Privacy (1), 60–68 (2012)
21. Probst, C.W., Willemson, J., Pieters, W.: The Attack Navigator. In: Proc. of GraMSec. pp. 1–17. LNCS, Springer (2015)
22. PWC: The global state of information security survey `http://www.pwc.com/gx/en/issues/cyber-security/information-security-survey.html` (2016)
23. Sasse, M.A., Brostoff, S., Weirich, D.: Transforming the weakest linka human/-computer interaction approach to usable and effective security. BT Technology J. 19(3) (2001)
24. Scandariato, R., Wuyts, K., Joosen, W.: A descriptive study of Microsofts threat modeling technique. Requirements Engineering 20(2), 163–180 (2015)
25. Stålhane, T., Sindre, G.: Identifying safety hazards: An experimental comparison of system diagrams and textual use cases. In: Enterprise, Business Proc. and Inf. Sys. Mod. (2012)
26. The TREsPASS Project: Technology-supported Risk Estimation by Predictive Assessment of Socio-technical Security: `http://www.trespass-project.eu/` (2016)
27. Wuyts, K., Scandariato, R., Joosen, W.: Empirical evaluation of a privacy-focused threat modeling methodology. J. of Syst. and Soft. 96, 122–138 (2014)