Human Aspect of Threat Analysis: A Replication

Katja Tuma
k.tuma@vu.nl
Vrije Universiteit Amsterdam
The Netherlands

Winnie Mbaka
w.mbaka@vu.nl
Vrije Universiteit Amsterdam
The Netherlands

ABSTRACT

Background: Organizations are experiencing an increasing demand for security-by-design activities (e.g., STRIDE analyses) which require a high manual effort. This situation is worsened by the current *lack of diverse (and sufficient)* security workforce and inconclusive results from past studies. To date, the deciding human factors (e.g., diversity dimensions) that play a role in threat analysis have not been sufficiently explored.
Objective: To address this issue, we plan to conduct a series of exploratory controlled experiments. The main objective is to empirically measure the human-aspects that play a role in threat analysis alongside the more well-known measures of analysis performance.
Method: We design the experiments as a differentiated replication of past experiments with STRIDE. The replication design is aimed at capturing some similar measures (e.g., of outcome quality) and additional measures (e.g., diversity dimensions). We plan to conduct the experiments in an academic setting.
Limitations: Obtaining a balanced population (e.g., wrt gender) in advanced computer science courses is not realistic. The experiments we plan to conduct with MSc level students will suffer this limitation. We plan to (at least) measure the self-reported team dynamics, though we can not control its effects on the analysis outcomes.

KEYWORDS

Threat Analysis, Human Aspects, Empirical Software Engineering, Replication, Controlled Experiment

ACM Reference Format:
Katja Tuma and Winnie Mbaka. 2022. Human Aspect of Threat Analysis: A Replication. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Security-by-design techniques [8, 29] have been used to prevent costly security fixes to software in later stages of the development life-cycle by analyzing security already during the design phase. Practitioners use threat analysis [35] to look for potential security threats in their product's software architecture. For instance, STRIDE [32] is a popular technique developed by Microsoft.

There is an increasing need to perform such architectural security analyses (e.g., latest BSIMM study reports an increased investment by more than 65% [10]) as the threat landscape evolves. However, threat analysis requires a high manual effort [30], demands the involvement of security and domain experts [6], and has been proven time and again difficult to fully automate [38].

Threat analysis practices are set back by a globally recorded shortage of the security workforce [4, 7]. In addition, the current security workforce is not diverse (e.g., with respect to gender) which may be viewed as an opportunity for a change.

Risk decisions (which are core to threat analysis) are made in face of uncertainty [3], thus there is space for subjective (and possibly biased) judgement [5, 14]. Empirical evidence of threat analysis performance indicators is a crucial piece of the puzzle to improve the situation. But, past empirical studies were either inconclusive about some performance indicators [37] or have focused on measuring performance indicators irrespective of the human factors [30, 36, 39]. Yet measuring such human factors is pivotal to understanding how to close the security workforce gap in the future.

To address these issues, we plan to conduct a series of exploratory controlled experiments with the aim of empirically measuring the human-aspects that play a role in threat analysis. In particular, we design a differentiated replication [20], where we capture some similar measures used in previous experiments [37] but also different measures (e.g., participant gender, nationality, type of outcomes, etc).

2 RELATED WORK

We positioned our contributions with respect to existing literature on empirical studies of STRIDE and related replication studies.
Empirical studies of threat analysis. In addition to the replicated study [37], several works have investigated STRIDE empirically. Scandariato et al. [30] performed a descriptive analysis measuring the productivity, precision, and recall of STRIDE in an academic setting. Their study reports similar values for a version of STRIDE-per-element, however the conditions of the descriptive study were different compared to our controlled experiment, therefore the results can not be directly compared.

Two studies [2, 6] conducted case studies investigating the challenges of STRIDE. Bernsmed et al. [2] conducted semi-structured interviews (with transcription code analysis) with agile organizations and recorded the perceived challenges. The authors report that practitioners see value in performing STRIDE despite the high manual effort it requires. Other discovered challenges were related to the lack of expertise by developers conducting the analysis, and uncompatibility of systematic approaches with the Agile workflow.

Stevens et al. [33] conducted a qualitative case study to investigate the efficacy the Center of Gravity (CoG) technique in an industrial setting. The CoG, originally conceived as a military strategy,

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

is a risk-first threat analysis technique but has not been extensively used to analyze software security. The authors designed surveys and classroom sessions and involved 25 practitioners in the study. Similar to other studies conducted with experts, they report a very high accuracy of the participant results.

Replications. We frame our plan as a series of experimental replications. Generally, software engineering replication studies apply similar experimental procedures as the original study, on a different participation pool. This process is aimed at generating new data [28] (as opposed to re-analyzing the same data in reproduction studies). We briefly mention some related replication studies.

Labunets et al [19] conducted a controlled experiment that was replicated in [18] using student participants to compare two risk assessment methods, a visual and a textual method. The first study found that the visual method was more effective for identifying threats than the textual one. In contrast, the replicated experiment showed that the two techniques were (statistically) equivalent in terms of the quality of identified threats and security controls.

Several studies have empirically compared [16, 17, 22] and conducted replications [15, 23, 25] requirement engineering techniques (e.g., requirements elicitation). For brevity, we direct the interested reader a comprehensive review by Ambreen et al. [1].

3 RESEARCH QUESTIONS

Due to the academic setting we limit this study on observing gender, background, and nationality diversity dimensions (and exclude seniority). The main goal of this study is to measure the existence (or absence) of diversity effects on the actual and perceived analysis outcomes. Accordingly, we developed two research questions and hypotheses about each measure.

RQ1. *What is the effect of gender, background, and nationality on the* ***actual*** *threat analysis outcomes?*

To investigate RQ1, we pose hypotheses about the **equivalence of the sample means for the analysis outcomes**.

$H1_1 : Comp.Sci_F = Comp.Sci_M$

Regarding gender, we expect that the outcomes reported by women are equivalent to the outcomes reported by men. Studies of risk perception suggest that women perceive certain risks differently compared to men. Though we do not foresee strong differences, we might find some effects when it comes to risk priority.

$H1_2 : Comp.Sci_1 = Comp.Sci_2$

Regarding education, we expect that the students of various specialization tracks report equivalent outcomes for the same system under analysis.

$H1_3 : Comp.Sci_{Na} = Comp.Sci_{Nb}$

We expect that the students of various race and nationality report statistically equivalent outcomes.

RQ2. *What is the effect of gender, background, and nationality on the* ***perceived*** *threat analysis outcomes?*

To investigate RQ2, we pose hypotheses about the **equivalence of the sample means for the perceived analysis outcomes**.

$H2_1 : Perc(Comp.Sci_F) < Perc(Comp.Sci_M)$

Due to low confidence levels of female computer science students, we expect that the perceived quality of outcomes reported by women is lesser compared to the perceived quality of outcomes reported by men (regardless of the actual outcomes by both groups).

$H2_2 : Perc(Comp.Sci_1) = Perc(Comp.Sci_2)$

Regarding education, we expect that overall the students of various specialization tracks do not differ in their perceived quality of the outcomes they produced. We may find higher confidence levels of perceived quality for students that are following a security specialization track.

$H2_3 : Perc(Comp.Sci_{Na}) = Perc(Comp.Sci_{Nb})$

We expect that the students of various nationality do not differ in their perceived quality of the outcomes they produced.

4 REPLICATION PROTOCOL

4.1 Variables

Table 1 shows the variables of the study.

4.1.1 Independent. **Gender** is an individual's own gender identity, which is typically, man and woman, but can also be non-binary. Rodriguez et al. [24] found evidence of bias against women in some software engineering communities, and sometimes negative perceptions about women working in teams. Thus, this is an interesting dimension to further investigate in the context of security.

Education is an individuals' achieved level and topic of specialization (e.g., computer security vs AI) of academic studies. Risk-based decisions have to be made in organizations by the managerial layers, who typically have a good understanding of the product, but do not necessarily posses the technical skills of security experts or engineers. Therefore, it is interesting to investigate this dimension and include participants from a different domain (e.g., with background in communication sciences). Education turned out to be a non-significant variable in the study of the impact of commercial Antivirus on people's awareness of security incidents [13]. However, it is not clear whether this dimension has an impact in performing a RA task.

Nationality is the country of origin, which is often coupled with the culture and language that categorizes social groups. Race is a social construct linked with individual's physical characteristics such as skin color and is used to categorize populations. Determining the effect of nationality bias in security practices is to date an open question. Thomas et al. [34] conducted semi-structured interviews with 14 Black women in computing and report that Black women experienced isolation (though it is not clear whether due to gender or race or nationality). But, few studies have focused on nationality diversity in the software engineering discipline [24].

4.1.2 Dependent. Since the quality of analysis lacks a formalised definition (e.g., often natural language is used to describe attack scenarios and informal notations are used for modeling [35]), we will use measures that can be easily reproduced. Namely, we can observe how diversity dimensions effect the *type of analysis outcomes*. Table 1 (dependant variables) shows various outcomes types that we observe.

Threats. We use the STRIDE threat categories to distinguish different type of threats. Analyses conducted by experts tend to be more balanced in terms of their analysis of different threat categories, while novices tend to report more tampering, denial of service and information disclosure threats [36, 37]. We are interested to observe whether category distribution patterns emerge for other diversity dimensions.

Assumptions. Assumptions are statements about the domain that may or may not be true. Assumptions are often implicit and dynamic in nature (i.e., they can be invalidated and modified as the project evolves). Van Landuyt and Joosen [39] find that the majority of assumptions (created by students during STRIDE) were used to either justify an existence of threats or are used to eliminate threats. In [39] a substantial subset (78%) of the assumptions was in direct reference to security-related concepts (i.e., security assumptions), however also domain assumptions (statements about component functionalities) were made. Thus, we are interested to investigate the effect of diversity dimensions on the type of assumptions.

Attack surface. Security analysts often rely on defining the *attack surface* and the required attacker profile to exploit it to determine the feasibility of an attack scenario. Determining feasibility is subjective, domain-specific and not always obvious, therefore we include it in the dependent variables.

Risk priority. Since the number of identified threats explodes in realistic projects, practitioners must choose which threats are most urgent to mitigate. Thus they prioritize them based on estimations of risk. We refer to risk as a product of threat probability and impact. How individuals assess risk priorities may be related to their risk perception which is already well understood [11].

Mitigations. Mitigations of a security risk can be preventative (e.g., implementation of two-factor authentication), detective/reactive (e.g., using intrusion detection and access revocation techniques) and corrective (such as maintaining audit trails or restoring from a secure state). Multiple strategies can be adopted to counter a security threat, and the final choice may depend on domain-related factors, as the cost of implementing the mitigation has to be reasonable. A category of experts (e.g., man vs woman) may underestimate the ease with which a mitigation is actually implemented, as observed in [40]. Thus, we are interested to observe how diversity dimensions effect the type of mitigations that are identified during the analysis.

4.2 Material

Training. In the first part of the training the participants will be introduced to some key security topics (such as CIAA triad, security threats, attack surface and vulnerabilities, security controls and risk mitigations). The second part of the training will prepare the students to actually perform a threat analysis using one of the technique variants. The third part of the training will introduce the participants to the case study which will be the object of their analysis.

Case study documentation. We will use the same case study as in the original study. The home monitoring system (HomeSys) is an automated surveillance system designed for residential places. Its main objective is to enable the home-owner to remotely monitor their property. A detailed documentation of the case (requirements, architectural design, etc) will be made available to the participants.

Ground truth analysis. We will use one 'golden standard' data flow diagram and its' corresponding ground truth STRIDE analysis of the HomeSys case study from [37]. Since we do not aim to measure the quality of the diagrams created, and the DFD building is less time consuming compared to threat identification, we will provide a model to the participants. This will significantly simply the comparison of the identified security threats. Similarly, we will provide the ground truth analysis to the participants that will be prioritizing threats and identifying security mitigations.

4.3 Task

The participants will be asked to individually fill-in a survey. The survey consists of three parts. First, a few questions about the students gender, background, nationality. Half of the participants will be asked to perform a STRIDE analysis (i.e., identify security threats). In contrast to previous studies, our participants will *be given the same graphical model* of HomeSys to analyze and they will analyze it using the same STRIDE technique. The other half of the participants will be asked to prioritize a list of security threats and identify security mitigations to high-priority threats. In contrast to previous studies, our participants will be given the graphical model *and the list of security threats*. To guide the threat identification the participants will use the documentation of STRIDE. Similar to the past studies, we will hand out a threat template csv to standardize the format of the outcomes reported. The participants will submit the files using the same survey. Finally, they will be asked a few questions regarding their perception of the task. Time taken to complete the task was captured using an online survey tool.

4.4 Participants

Our population is computer science students, with some differences in the elective courses and program choices (e.g., we plan to include students from various master programs, such as IA, computer Security, and Software Engineering). All participants are students enrolled in a course taught by the experimenters. At the beginning of the course we plan to hand out an entry survey to measure participants' background and areas of expertise relevant to the study. We expect most to be new to secure design techniques (e.g STRIDE, threat modeling, Data Flow Diagrams, misuse cases, attack trees etc). In addition, we expect the participants are unfamiliar with architectural modeling techniques (e.g sequence, component and deployment diagrams).

4.5 Execution plan

Work division. The participants will be randomly divided into two groups (A and B). Group A will be tasked with analysing a provided data flow diagram of the HomeSys case study using STRIDE. Group B will be tasked with prioritizing a provided list of security threats and identifying security mitigations for high-priority threats. These are not treatment groups. Rather, the groups are formed only to divide the work to avoid overloading the participants performing an overly complex task individually.

Training. The participants will undergo an obligatory training lectures (about 3 hours) covering the topics mentioned above.

Hand-outs. After the training, participants will be given digital copies of all the support material (inc. lecture slides, case documentation, technique documentation, etc).

Physical labs. The experiment will be conducted during a four hour

Name	Description	Scale	Operationalization
Independent variables (design)			
Gender	obtained from the gender of participants	nominal	multiple choice
Background	the program specialization and extra curriculum activities	nominal	multiple choice
Nationality	obtained from the nationality of participants	nominal	multiple choice
Dependent variables			
Different measures compared to existing literature			
Type of identified threats	distribution of categories of threats (spoofing, tampering, information disclosure, denial of service, elevation of privilege) that have been identified by the participants	nominal	see Section 4.2
Type of assumptions	distribution type of assumptions (domain, security) that have been reported by the participants	nominal	see Section 4.2
Type of attacks surface	distribution attack surfaces (physical, close-proximity, remote) of the identified threats	nominal	see Section 4.2
Risk priorities	distribution of risk priorities (high, medium, low) assigned to identified threats	nominal	see Section 4.2
Type of mitigations	distribution of type of identified mitigations (preventative, detective/reactive, corrective)	nominal	see Section 4.2
Treated/Measured variables			
Time spent on task	time (in hours) each team took to complete the task using the prescribed technique	ordinal	automatically measured by the submission tool
Perceived precision (PP)	self-reported ratio between the number of correctly identified threats and all *threats identified*	ordinal	5-point Likert scale
Perceived recall (PR)	self-reported ratio between the number of correctly identified threats and all *existing* threats identified	ordinal	5-point Likert scale
Perceived usefulness (PU)	self-reported usefulness of the prescribed technique	ordinal	5-point Likert scale
Experience with security and modeling	self-reported experience in number of years or previously completed courses	ordinal	5-point Likert scale
Experience with STRIDE	self-reported experience in number of years or previously completed courses	ordinal	5-point Likert scale
Experience with domain of application	self-reported experience in number of years or previously completed courses	ordinal	5-point Likert scale
Different measures compared to existing literature			
Perceived cognitive load	the reported cognitive load (complexity) of the task using the prescribed technique	ordinal	5-point Likert scale
Perceived team dynamics	the reported quality level of group work	ordinal	5-point Likert scale

Table 1: Variables of the differentiated replication experiments

physical lab. The teams will be separated into different classrooms depending on their treatment group (to avoid spillover effects). Each classroom will be supervised by either a teaching assistant or the experimenters. Only questions about the experiment protocol will be answered.

Reports. The data will be collected through an online survey tool.

4.6 Analysis plan

Data cleaning. We will perform a preliminary check of the collected data. This will include removing submissions for which we did not get explicit consent by the participant. Second, we will remove clearly insincere submission attempts (if any).

TOST analysis of equivalence. We will use both difference and equivalence statistical tests. As some of our data are ordinal and comes from independent samples, we will perform Mann-Whitney test. For the equivalence test we will use TOST, which was initially proposed by [31] and is widely used in pharmacological and food sciences to answer the question whether two treatments are equivalent within a particular range δ [9, 21]. Wherever possible (e.g., for Likert-scale questions) we will define the delta empirically. For instance by pooled variance σ_p across several samples reported in the literature on security risk analysis (e.g., in a four year interval) on variables ranging over a 5-item Likert scale for demographic statistics as to account for natural variability of the data.

Validity threats. There is typically around 20% (or less) female students enrolled in computer science programs. We are aware of the validity threats caused by an unbalanced population sample, which is omnipresent in all gender diversity studies in STEM disciplines [24]. To partially mitigate this threat, we will rally female computer scientist students towards participation through local steminist groups and similar community organized channels.

Since we do not include practitioners in this study, we can not observe the full complexity of the diversity effects (e.g., including seniority) that are actually present in organizations where threat analysis is routinely performed. Still, studies have shown [12, 26, 27] that the differences between the performance of professionals and graduate students are often limited.

We considered the threat of overloading the participants with a complex task. We mitigate this threat by splitting the participants into two groups, so individual participants get to either only focus on finding threats or focus on mitigating risks.

5 ACKNOWLEDGMENTS

REFERENCES

[1] Talat Ambreen, Naveed Ikram, Muhammad Usman, and Mahmood Niazi. 2018. Empirical research in requirements engineering: trends and opportunities. *Requirements Engineering* 23, 1 (2018), 63–95.

[2] Karin Bernsmed and Martin Gilje Jaatun. 2019. Threat modelling and agile software development: Identified practice in four Norwegian organisations. In *2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*. IEEE, 1–8.

[3] Vicki Bier. 2020. The Role of Decision Analysis in Risk Analysis: A Retrospective. *Risk Analysis* 40, S1 (2020), 2207–2217.

[4] Borka Jerman Blažič. 2021. Cybersecurity Skills in EU: New Educational Concept for Closing the Missing Workforce Gap. In *Cybersecurity Threats with New Perspectives*. IntechOpen.

[5] Mario P Brito and Ian GJ Dawson. 2020. Predicting the validity of expert judgments in assessing the impact of risk mitigation through failure prevention and correction. *Risk analysis* 40, 10 (2020), 1928–1943.

[6] Daniela Soares Cruzes, Martin Gilje Jaatun, Karin Bernsmed, and Inger Anne Tøndel. 2018. Challenges and experiences with applying microsoft threat modeling in agile development projects. In *2018 25th Australasian Software Engineering Conference (ASWEC)*. IEEE, 111–120.

[7] CyberSeek. 2019. *Cybersecurity Supply/Demand Heat Map*. Retrieved April 21, 2022 from https://www.cyberseek.org/heatmap.html

[8] Chad Dougherty, Kirk Sayre, Robert C Seacord, David Svoboda, and Kazuya Togashi. 2009. *Secure Design Patterns*. Technical Report. Carnegie-Mellon University Pittsburgh, Software Engineering Institute.

[9] Food and Drug Administration. 2001. Guidance for industry: Statistical approaches to establishing bioequivalence.

[10] Synopsys Software Integrity Group. 2021. *Building Security In Maturity Model (BSIMM12)*. Retrieved April 20, 2022 from https://www.bsimm.com

[11] Per E Gustafsod. 1998. Gender Differences in risk perception: Theoretical and methodological erspectives. *Risk analysis* 18, 6 (1998), 805–811.

[12] Martin Höst, Björn Regnell, and Claes Wohlin. 2000. Using students as subjects—a comparative study of students and professionals in lead-time impact assessment. *Empirical Software Engineering* 5, 3 (2000), 201–214.

[13] Eric Jardine. 2020. The Case against Commercial Antivirus Software: Risk Homeostasis and Information Problems in Cybersecurity. *Risk Analysis* 40, 8 (2020), 1571–1588.

[14] Johannes G Jaspersen and Gilberto Montibeller. 2015. Probability elicitation under severe time pressure: A rank-based method. *Risk Analysis* 35, 7 (2015), 1317–1335.

[15] J. Jung, K. Hoefig, D. Domis, A. Jedlitschka, and M. Hiller. 2013. Experimental comparison of two safety analysis methods and its replication. *International Symposium on Empirical Software Engineering and Measurement* (2013), 223–232. https://doi.org/10.1109/ESEM.2013.59

[16] Peter Karpati, Andreas L Opdahl, and Guttorm Sindre. 2011. Experimental comparison of misuse case maps with misuse cases and system architecture diagrams for eliciting security vulnerabilities and mitigations. In *Availability, Reliability and Security (ARES), 2011 Sixth International Conference on*. IEEE, 507–514.

[17] Peter Karpati, Guttorm Sindre, and Raimundas Matulevicius. 2012. Comparing misuse case and mal-activity diagrams for modelling social engineering attacks. *International Journal of Secure Software Engineering (IJSSE)* 3, 2 (2012), 54–73.

[18] Katsiaryna Labunets, Fabio Massacci, and Federica Paci. 2017. On the equivalence between graphical and tabular representations for security risk assessment. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, 191–208.

[19] Katsiaryna Labunets, Fabio Massacci, Federica Paci, et al. 2013. An experimental comparison of two risk-based security methods. In *2013 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*. IEEE, 163–172.

[20] R Murray Lindsay and Andrew SC Ehrenberg. 1993. The design of replicated studies. *The American Statistician* 47, 3 (1993), 217–228.

[21] Michael Meyners. 2012. Equivalence tests–A review. *Food quality and preference* 26, 2 (2012), 231–245.

[22] Andreas L Opdahl and Guttorm Sindre. 2009. Experimental comparison of attack trees and misuse cases for security threat identification. *Information and Software Technology* 51, 5 (2009), 916–932.

[23] M. Riaz, J. King, J. Slankas, L. Williams, F. Massacci, C. Quesada-López, and M. Jenkins. 2017. Identifying the implied: Findings from three differentiated replications on the use of security requirements templates. *Empirical Software Engineering* 22, 4 (2017), 2127–2178. https://doi.org/10.1007/s10664-016-9481-1

[24] Gema Rodríguez-Pérez, Reza Nadri, and Meiyappan Nagappan. 2021. Perceived diversity in software engineering: a systematic literature review. *Empirical Software Engineering* 26, 5 (2021), 1–38.

[25] S. Rueda, J.I. Panach, and D. Distante. 2020. Requirements elicitation methods based on interviews in comparison: A family of experiments. *Information and Software Technology* 126 (2020). https://doi.org/10.1016/j.infsof.2020.106361

[26] Per Runeson. 2003. Using students as experiment subjects–an analysis on graduate and freshmen student data. In *Proceedings of the International Conference on Empirical Assessment in Software Engineering*. 95–102.

[27] Iflaah Salman, Ayse Tosun Misirli, and Natalia Juristo. 2015. Are students representatives of professionals in software engineering experiments?. In *Proceedings of the International Conference on Software Engineering-Volume 1*. IEEE Press, 666–676.

[28] Adrian Santos, Sira Vegas, Markku Oivo, and Natalia Juristo. 2019. A procedure and guidelines for analyzing groups of software engineering replications. *IEEE Transactions on Software Engineering* 47, 9 (2019), 1742–1763.

[29] Joanna C. S. Santos, Katy Tarrit, and Mehdi Mirakhorli. 2017. A Catalog of Security Architecture Weaknesses. In *Proceedings of the International Conference on Software Architecture Workshops (ICSAW)*. IEEE Computer Society, 220–223. https://doi.org/10.1109/ICSAW.2017.25

[30] Riccardo Scandariato, Kim Wuyts, and Wouter Joosen. 2015. A descriptive study of Microsoft's threat modeling technique. *Requirements Engineering* 20, 2 (2015), 163–180.

[31] DL Schuirmann. 1981. On hypothesis-testing to determine if the mean of a normal-distribution is contained in a known interval. *Biometrics* 37, 3 (1981), 617–617.

[32] Adam Shostack. 2014. *Threat modeling: Designing for security*. John Wiley & Sons.

[33] R Stevens, D Votipka, and E.M. Redmiles. 2018. The Battle for New York: A Case Study of Applied Digital Threat Modeling at the Enterprise Level. In *SEC'18: Proceedings of the 27th USENIX Conference on Security Symposium*. USENIX Association, 621–637.

[34] Jakita O Thomas, Nicole Joseph, Arian Williams, Jamika Burge, et al. 2018. Speaking truth to power: Exploring the intersectional experiences of Black women in computing. In *2018 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT)*. IEEE, 1–8.

[35] Katja Tuma, Gül Calikli, and Riccardo Scandariato. 2018. Threat analysis of software systems: A systematic literature review. *Journal of Systems and Software* 144 (2018), 275–294.

[36] Katja Tuma, Christian Sandberg, Urban Thorsson, Mathias Widman, Thomas Herpel, and Riccardo Scandariato. 2021. Finding security threats that matter: Two industrial case studies. *Journal of Systems and Software* 179 (2021), 111003.

[37] Katja Tuma and Riccardo Scandariato. 2018. Two architectural threat analysis techniques compared. In *European Conference on Software Architecture*. Springer, 347–363.

[38] Katja Tuma, Laurens Sion, Riccardo Scandariato, and Koen Yskout. 2020. Automating the early detection of security design flaws. In *Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems*. 332–342.

[39] Dimitri Van Landuyt and Wouter Joosen. 2021. A descriptive study of assumptions in STRIDE security threat modeling. *Software and Systems Modeling* (2021), 1–18.

[40] George Wright, Fergus Bolger, and Gene Rowe. 2002. An empirical test of the relative validity of expert and lay judgments of risk. *Risk Analysis: An International Journal* 22, 6 (2002), 1107–1122.