Threat modeling: from infancy to maturity

Koen Yskout
koen.yskout@cs.kuleuven.be
imec-DistriNet, KU Leuven
Leuven, Belgium

Thomas Heyman
thomas.heyman@toreon.com
Toreon
Antwerp, Belgium

Dimitri Van Landuyt
dimitri.vanlanduyt@cs.kuleuven.be
imec-DistriNet, KU Leuven
Leuven, Belgium

Laurens Sion
laurens.sion@cs.kuleuven.be
imec-DistriNet, KU Leuven
Leuven, Belgium

Kim Wuyts
kim.wuyts@cs.kuleuven.be
imec-DistriNet, KU Leuven
Leuven, Belgium

Wouter Joosen
wouter.joosen@cs.kuleuven.be
imec-DistriNet, KU Leuven
Leuven, Belgium

ABSTRACT

Threat modeling involves the systematic identification and analysis of security threats in the context of a specific system. This paper starts from an assessment of its current state of practice, based on interactions with threat modeling professionals. We argue that threat modeling is still at a low level of maturity and identify the main criteria for successful adoption in practice. Furthermore, we identify a set of key research challenges for aligning threat modeling research to industry practice, thereby raising the technology-readiness levels of the ensuing solutions, approaches, and tools.

ACM Reference Format:
Koen Yskout, Thomas Heyman, Dimitri Van Landuyt, Laurens Sion, Kim Wuyts, and Wouter Joosen. 2020. Threat modeling: from infancy to maturity. In *New Ideas and Emerging Results (ICSE-NIER'20), May 23–29, 2020, Seoul, Republic of Korea.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3377816.3381741

1 INTRODUCTION

The goal of threat modeling is to identify security threats to a system, assess their risk, propose mitigations, and follow-up on their implementation. According to Shostack [15], the threat modeling process entails systematically answering the following questions: (1) What are we building? (2) What can go wrong? (3) What are we going to do about it? (4) Did we do a good enough job?

Systematic processes for threat modeling, initiated by the STRIDE approach at Microsoft around twenty years ago [8], have been gaining traction in both academic and industrial settings since then [22, 27]. This paper argues that, despite these twenty years, the state of practice has largely remained ad-hoc and based on 'whiteboard hacking', relying to a large extent on the experience of the people involved. Academic efforts do not find their way into practice, and industry-driven practices are of a pragmatic nature rather than being based on scientific insights.

The central thesis of this paper is based on collaborations between researchers from KU Leuven and practitioners from Toreon, a company specialized in threat modeling and training. This paper instantiates the first two steps of the technology transfer model of Gorschek et al. [7], namely identifying areas of improvement and formulating a high-level research agenda. The ultimate goal of the paper is to foster new and industry-relevant research tracks on threat modeling and form a community around them.

ICSE-NIER'20, May 23–29, 2020, Seoul, Republic of Korea

2 THREAT MODELING

We start by sketching the threat modeling process as practiced by Toreon. Complemented with the coverage in the academic literature, we argue that threat modeling is still an immature field, posing several research challenges.

2.1 Threat modeling in practice

The insights in this section are based on a sample of 20 typical threat modeling projects performed for 9 different customers by Toreon in the period of 2014–2019.

Every project in the sample followed the same structure (Figure 1). Stakeholders agree on the scope and the goal of the project in a kick-off meeting. After that, a model of the system (typically a whiteboard draft of a data flow diagram) is created in a modeling session and manually finalized by a threat modeling expert afterwards. Subsequently, in a threat elicitation session, this system model is used to uncover threats, which are analyzed and ranked by an expert afterwards. Then, the expert prepares a review meeting, while a colleague performs quality assurance checks on the result. Finally, the results are presented to all stakeholders in a review meeting. Some of the sessions may be repeated if required.

Each project typically involves around ten different stakeholders, ranging from business owners, security experts, threat modeling experts, software architects, developers, and IT infrastructure specialists; not all of these are necessarily involved in all project phases. The threat modeling experts play a key role throughout the entire project, as both analysts and facilitators during the sessions. These sessions are highly interactive and very lively; the model goes through multiple iterations, conflicting views arise, and the overall focus dynamically switches from one part of the system to another.

The time spent on each phase varies between projects, influenced by factors such as prior experience of the stakeholders with threat modeling, complexity and maturity of the system and the application domain, regulatory constraints, and so on. Figure 1 shows the detailed time spent by threat modeling experts (45 hours in total) for one concrete project where all stakeholders were familiar with the process, thus excluding unforeseen project management overhead. Across all 20 sampled projects (including some projects in which

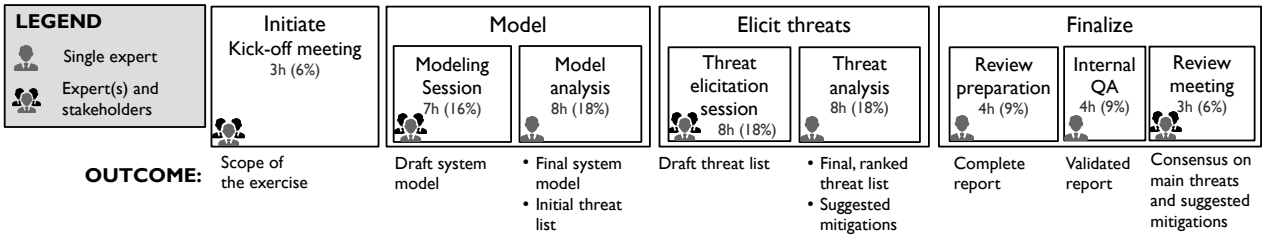

Figure 1: Graphical overview of a typical threat modeling project, including the involved people, the outcomes of each activity, and the combined time spent on that phase by threat modeling experts.

this overhead was significant), the average total time is slightly higher (about 69 hours per project, with a standard deviation of 32).

Note that the time spent on each phase does not indicate the *value* derived from that phase. One of the main benefits of threat modeling is establishing a shared understanding among the stakeholders about the business value and security risks, which is cultivated mainly during the interactive sessions. Shortening or removing those sessions is therefore not necessarily desirable.

Typically encountered problems in those sometimes chaotic sessions are that some details or system parts are overlooked, or stakeholder input is not captured accurately. Hence, more in-depth model and threat analysis is typically performed afterwards by a threat modeling expert in isolation. The key concern here is a qualitative outcome. Typical issues are the error-prone nature of manually iterating through a model, and a lack of specific knowledge, for example about a particular technology used in the system.

Finally, the review preparation and internal QA phases verify the correctness and completeness, the ranking of the threats, and whether the key risks are clearly communicated. These phases are again performed by a threat modeling expert in isolation. Typical problems in these phases are again the error prone nature of manual verification, and the ambiguity of assigning risk and ranking threats.

When the threat model of the system needs to be revised at a later time, repeating the entire process is often the only practical solution. Documenting the outcome of the previous run (including the assumptions and rationale) is therefore essential to increase the efficiency, especially if the group of involved stakeholders and threat modeling experts has changed in the meantime.

2.2 Threat modeling in the literature

Academically, threat modeling primarily appears in the context of eliciting security requirements, interacting closely with both risk analysis and secure software design [11, 25]. Threat modeling has also been extended for privacy [5, 26]. Several surveys have appeared on the topic, both pragmatic [4, 13] and systematic [22, 27]. We do not attempt to replicate these, but build on their findings.

Xiong and Lagerström find that *"threat modeling is a diverse field lacking common ground. The definitions are numerous, and used in many different and perhaps also incompatible ways"* [27]. The lack of a dedicated community and publication venue reflects this. Threat modeling publications are currently scattered among general software engineering, requirements engineering, and security and privacy conferences and journals. Despite the potential of threat modeling, there is a lack of published experience reports and industrial case studies. A few industry-based descriptions of threat modeling approaches [18, 21] and experience reports [6, 14, 19, 20] on STRIDE exist. In addition, some case studies can be found in the domains of hardware [2], cloud infrastructure [10], and even security standards [9]. While these few examples illustrate the broad applicability in diverse domains, they do not provide sufficient insight into the best practices and industry adoption barriers.

Furthermore, scientific evaluations of existing threat modeling approaches are seldomly performed, although exceptions exist [12, 23]. In terms of improvements to existing approaches, an important topic gaining traction is the enrichment of the underlying models to enable automation [1, 3, 16, 17, 24].

Both systematic reviews arrive at similar observations, namely that *"[…] the existing techniques lack in quality assurance of outcomes. Furthermore, the techniques lack maturity, validation and tool support."* [22], and *"most threat modeling work remains to be done manually, and there is limited assurance of their validations"* [27].

We observe that the current body of academic work about threat modeling does not align well with the needs of practitioners. Research so far has focused mainly on the threat elicitation step. As can be seen from Figure 1, however, this step only accounts for roughly one third of the total time spent on the project, and, while essential, it is not the most time-consuming part.

We conclude our overview by summarizing our position: **Threat modeling, as an engineering discipline, is currently at a very low level of maturity, both in terms of research, tool support, and in practice.**

3 THE WAY FORWARD

The typical approach sketched in Section 2.1 remains a largely manual effort, suffering from several drawbacks. In this section, based on our collaboration and discussions with practitioners, we formulate a research roadmap for attaining higher maturity levels of threat modeling as an engineering discipline.

3.1 Criteria for industry adoption

We first propose six criteria that are considered to be essential for successful industry adoption of a threat modeling approach. These criteria must therefore be taken into account while addressing the challenges formulated in the next section.

3.1.1 Model-based. While threat modeling is already model-based, new approaches can assist by highlighting potential gaps and ambiguities during model creation to focus the discussion. Most importantly, though, any modeling support must be compatible with the often chaotic and intense threat modeling sessions in which stakeholders come to a shared understanding of the system, and facilitate this process rather than restrict it.

3.1.2 Traceable. The construction of detailed system models and resolution of different participant interpretations generates a lot of knowledge on the underlying rationale and assumptions. To ensure maintainability, this information needs to be captured and linked to the threat model; if not, future updates are hard to do without accidentally overlooking anything, forcing the modeler to redo most of the analysis.

3.1.3 Systematic. Threat modeling support should increase the trust of the participants in the correctness and completeness of the model, especially when the threat modeler has less expertise in certain technologies or architectural patterns used in the system. The quality of the outcome should ideally depend as little as possible on the competence of the expert. Automation and tool support can play a powerful role here as long as they remain under control of the expert (e.g., by making suggestions), as not all relevant factors and knowledge can be formalized.

3.1.4 Business integration. A threat modeling process must integrate with larger corporate management processes. Enterprises typically rely on information security management systems (ISMS) to link IT risks to business risks and to manage them over time. To make any true impact, a threat modeling approach should facilitate reporting the identified risks to the stakeholders and integrate in larger-scale corporate governance and risk management processes.

3.1.5 Context-aware. The system that is being threat modeled is part of a larger enterprise architecture. Threat modeling needs to allow an individual model to easily be integrated into that larger context. Specifically, suggested mitigations should make use of already-existing architectural building blocks where possible, while avoiding conflicts with agreed-upon strategies, patterns, and corporate policies. Attention to change management is also important, both for updating threat models in the face of updates to the system and newly-discovered weaknesses in existing building blocks. This context-awareness is critical for making threat modeling a standard security best practice, rather than a one-off exercise.

3.1.6 Scalable. A threat modeling approach must be scalable in terms of the resources that need to be invested in function of the size and complexity of the system. That is, the approach should scale down to a light-weight process for threat modeling a single small system, developed by a small team and threat modeled by an enthusiastic 'security champion' (who is not necessarily a security expert). On the other hand, the approach should scale up to large companies where complex systems, composed of tens to hundreds of internal projects need to be threat modeled.

3.2 Research roadmap

Inspired by maturity models such as COBIT and CMMI, we propose 5 maturity levels for threat modeling, and the research challenges that need to be addressed to evolve from one level to another. We have argued that threat modeling is currently at **Level 1: Initial, ad-hoc**, i.e., some initiatives exist, but these do not follow a rigorous, standardized process.

For moving to **Level 2: Repeatable but intuitive**, we believe that it is essential to first develop a better understanding of threat modeling.

Challenge 1. *(Level 1 → 2) Develop a reference model for threat modeling, which makes it possible to share threat modeling artifacts in a standardized manner for reuse, education, or benchmarking.*

The concept of 'threat modeling' is currently not well-defined, and different actors have a different understanding about what it entails (cfr. the findings of the academic surveys [22, 27]). Research initiatives have not yet produced an appealing, let alone standard, reference model or exemplar that can be used to capture what threat modeling entails. Therefore, it is not always clear to businesses what to aim for with a threat modeling project. For example, some believe that addressing the 'top 10 threats' is a valid substitute for a systematic threat modeling activity. A common reference model and baseline will help to clarify the value of systematic threat modeling. Furthermore, it serves as the basis for a method to exchange threat modeling information in a standardized fashion, and it can underpin tools that support the threat modeling process. Finally, addressing this research challenge can serve as a trigger for the creation of a threat modeling community and venues dedicated to this topic.

Even with this common foundation, the execution of a threat modeling project still relies heavily on the availability of expertise. In order to move to **Level 3: Defined process**, it is necessary that the process to create a threat model becomes well-supported, such that the impact of and variation due to the individuals executing it becomes smaller. Two major hurdles for making the transition to this maturity level are (1) better support during the interactive modeling sessions, and (2) reuse of knowledge across projects, experts, and organizational boundaries.

Challenge 2. *(Level 2 → 3) Develop modeling support for tracking the iterative creation of the model and the assumptions that were made, rather than only the final model.*

Such a modeling approach should thus explicitly acknowledge, rather than ignore, the chaotic nature of the modeling sessions it is used in. During these sessions, the model will undergo many changes in quick succession. The approach should make it possible to easily record the decisions when they are made, and at the same time avoid imposing constraints on model consistency and completeness (which can often not be resolved during this highly interactive session). Important assumptions that are made need to be recorded and followed up upon (i.e., to verify whether they really hold). We thus envisage a 'model-as-you-go' approach, where the operations on the model (and their rationale) are recorded during the modeling session in a flexible yet meaningful domain-specific language, and the model itself is derived as a by-product from that. Nevertheless, the modeling approach should also maximally support the analysis phase, where the model is consolidated, inconsistencies are resolved, and details are added, preferably supported by automation, while retaining the collected information.

With respect to the second hurdle, improving the reuse of knowledge, we observe that many knowledge bases for threat modeling

already exist, for example in the form of threat trees, attack patterns, or top 10 lists. However, they are not referred to systematically by (expert) practitioners during threat elicitation. They are considered most useful for beginning threat modelers to get familiar with the concepts of security (or privacy), while experts will rely on their own expertise. An interesting challenge for academia is thus to make knowledge bases more effective in practice.

Challenge 3. *(Level 2 → 3) Design knowledge bases that integrate well with a structured threat modeling process, bringing the relevant focus and expertise with minimal effort.*

Practitioners will need to be able to find the most suitable knowledge base, assess its quality, and be guided in how to use it. An important part of this challenge is to empirically quantify the merits of a knowledge base, for example in terms of cost savings or guarantees on the quality of results. Furthermore, experts need to be persuaded into updating the knowledge bases to maximally reuse the results of previous threat modeling exercises.

Once a well-defined process is in place, transitioning to **Level 4: Managed and measurable** requires suitable metrics to measure how well threat modeling is functioning in practice.

Challenge 4. *(Level 3 → 4) Define validated metrics to assess the quality of a threat modeling process and its outcome.*

This challenge refers back to the fourth question of Shostack [15]: did we do a good enough job? The question not only pertains to practitioners, but also to researchers: what metrics can be used to scientifically demonstrate the success of a threat modeling effort? The DevOps industry movement is also heavily metrics-driven, setting up a monitoring infrastructure to assess whether certain quality levels are reached. For threat modeling, an important question would be to assess, at run-time, to which extent the threat model still corresponds to reality. It is yet unclear which metrics would be suitable for this purpose, though.

Finally, in the move to **Level 5: Optimized**, a feedback process needs to be put in place that provides continuous quality assurance based on the metrics of level four.

Challenge 5. *(Level 4 → 5) Develop intelligent automation for self-adaptive and dynamic threat modeling.*

At this fifth level, the role of automation is significantly expanded. Automation no longer just supports the threat modeling process, but becomes part of the process itself, leading to self-adaptive, dynamic threat modeling. That is, run-time introspection of a system feeds information into the underlying threat model, for example on the detection of attempted attacks or the amount of sensitive information stored in a particular data store. This triggers a re-evaluation of the applicable threats and their relative priority, after which automated adaptations to the system can be made, such as changing a security parameter or deploying an additional countermeasure.

The sketched research roadmap is ambitious, and it will take time to translate it into practical contributions. However, the reward for doing so is great, as it will revolutionize threat modeling from its infancy into a mature and practical engineering discipline.

ACKNOWLEDGMENTS

This research is partially funded by the Research Fund KU Leuven and the Flemish Research Programme Cybersecurity.

REFERENCES

[1] Thibaud Antignac, Riccardo Scandariato, and Gerardo Schneider. 2016. A Privacy-Aware Conceptual Model for Handling Personal Data. In *ISoLA 2016, Part I*. Springer, 942–957.
[2] ARM Ltd. 2018. Platform Security Architecture. https://developer.arm.com/architectures/security-architectures/platform-security-architecture
[3] Bernhard J. Berger, Karsten Sohr, and Rainer Koschke. 2016. Automatically Extracting Threats from Extended Data Flow Diagrams. *ESSoS 2016 (LNCS)* 9639 (2016), 56–71.
[4] Deborah J. Bodeau, Catherine D. McCollum, and David B. Fox. 2018. Cyber Threat Modeling: Survey, Assessment, and Representative Framework. The MITRE Corporation, case number 18-1174.
[5] Mina Deng, Kim Wuyts, Riccardo Scandariato, Bart Preneel, and Wouter Joosen. 2011. A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. *Requirements Engineering* 16, 1 (2011).
[6] Danny Dhillon. 2011. Developer-Driven Threat Modeling — Lessons Learned in the Trenches. *IEEE Security & Privacy* 9, 4 (2011), 41–47.
[7] Tony Gorschek, Per Garre, Stig Larsson, and Claes Wohlin. 2006. A model for technology transfer in practice. *IEEE Software* 23, 6 (2006), 88–95.
[8] Loren Kohnfelder and Praerit Garg. 1999. The threats to our products. https://www.microsoft.com/security/blog/2009/08/27/the-threats-to-our-products/
[9] Tom Lodderstedt, Mark McGloin, and Phil Hunt. 2013. *OAuth 2.0 Threat Model and Security Considerations*. RFC 6819. RFC Editor. http://www.rfc-editor.org/rfc/rfc6819.txt
[10] Microsoft Corporation. 2018. *Performing threat modeling for the Azure IoT reference architecture*. https://docs.microsoft.com/en-us/azure/iot-fundamentals/iot-security-architecture#performing-threat-modeling-for-the-azure-iot-reference-architecture
[11] Suvda Myagmar, Adam J. Lee, and William Yurcik. 2005. Threat Modeling as a Basis for Security Requirements. In *Symposium on requirements engineering for information security (SREIS)*. 1–8.
[12] Riccardo Scandariato, Kim Wuyts, and Wouter Joosen. 2015. A descriptive study of Microsoft's threat modeling technique. *Requirements Engineering* 20, 2 (2015), 163–180.
[13] Nataliya Shevchenko, Timothy A Chick, Paige O'Riordan, Thomas Patrick Scanlon, and Carol Woody. 2018. Threat Modeling: a Summary of Available Methods. https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=524448
[14] Adam Shostack. 2008. Experiences threat modeling at Microsoft. In *Workshop on Modeling Security (MODSEC08)*.
[15] Adam Shostack. 2014. *Threat Modeling: Designing for Security*. 590 pages.
[16] Laurens Sion, Dimitri Van Landuyt, Koen Yskout, and Wouter Joosen. 2018. Sparta: Security & privacy architecture through risk-driven threat assessment. In *International Conference on Software Architecture Companion (ICSA-C)*. IEEE, 89–92.
[17] Laurens Sion, Kim Wuyts, Koen Yskout, Dimitri Van Landuyt, and Wouter Joosen. 2018. Interaction-based privacy threat elicitation. In *European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE, 79–86.
[18] J Steven. 2010. Threat Modeling - Perhaps It's Time. *Security and Privacy* 8, 3 (2010), 83–86.
[19] Rock Stevens, Daniel Votipka, Elissa M. Redmiles, Colin Ahern, Patrick Sweeney, and Michelle L. Mazurek. 2018. The Battle for New York: A Case Study of Applied Digital Threat Modeling at the Enterprise Level. In *27th USENIX Security Symposium (USENIX Security 18)*. USENIX Association, 621–637.
[20] Rock Stevens, Daniel Votipka, Elissa M Redmiles, Michelle Mazurek, and Colin Ahern. 2019. Applied Digital Threat Modeling: It Works! *IEEE Security & Privacy* PP, August (2019).
[21] Peter Torr. 2005. Demystifying the threat modeling process. *IEEE Security & Privacy Magazine* 3 (2005), 66–70.
[22] K. Tuma, G. Calikli, and R. Scandariato. 2018. Threat analysis of software systems: A systematic literature review. *Journal of Systems and Software* 144 (Oct 2018), 275–294.
[23] Katja Tuma and Riccardo Scandariato. 2018. Two Architectural Threat Analysis Techniques Compared. In *12th European Conference on Software Architecture (ECSA 2018)*, Vol. 11048. Springer, 347–363.
[24] Katja Tuma, Riccardo Scandariato, Mathias Widman, and Christian Sandberg. 2018. Towards Security Threats that Matter. In *3rd Workshop On The Security Of Industrial Control Systems & Of Cyber-Physical Systems (CyberICPS'17)*. Springer, 47–62.
[25] Sven Turpe. 2017. The Trouble with Security Requirements. In *25th International Requirements Engineering Conference (RE'17)*. IEEE, 122–133.
[26] Kim Wuyts. 2015. *Privacy Threats in Software Architectures*. Ph.D. Dissertation. KU Leuven.
[27] Wenjun Xiong and Robert Lagerström. 2019. Threat Modeling – A Systematic Literature Review. *Computers & Security* 84 (Jul 2019), 53–69.