Computers & Security 136 (2024) 103543

Contents lists available at ScienceDirect

Computers & Security

journal homepage: www.elsevier.com/locate/cose

Threat modeling of industrial control systems: A systematic literature review

Shaymaa Mamdouh Khalil[a,*], Hayretdin Bahsi[a,b], Tarmo Korõtko[c]

[a] School of Information Technologies, Tallinn University of Technology, Estonia
[b] School of Informatics, Computing, and Cyber Systems, Northern Arizona University, United States
[c] FinEst Centre for Smart Cities, Tallinn University of Technology, Estonia

ARTICLE INFO

Keywords:
Threat modeling (TM)
Cyber-physical system (CPS)
Industrial control system (ICS)
Systematic literature review (SLR)
Security-by-design
Risk assessment
Security analysis

ABSTRACT

Threat modeling is the process of identifying and mitigating potential threats to a system. It was originally developed to enhance software security during the design phase but has since been adapted for Industrial Control Systems (ICSs). ICSs are complex and interconnected systems that control critical infrastructure, such as power plants, water treatment facilities, and manufacturing plants. As such, they are major targets for cyberattacks, which may lead to human casualties, severe national security impacts, and financial instability. This systematic literature review explores the existing threat modeling methodologies for ICSs and emphasizes the importance of employing methodical frameworks that cover safety, security, and privacy aspects with clear procedural guidelines. The review reveals that ICSs threat modeling often lacks validation to ensure that the used methodologies are effective in identifying and mitigating threats. This study emphasizes the need to develop and apply better validation metrics in case studies. The main goal of this review is to help cyber security researchers and practitioners in selecting a suitable threat modeling approach that facilitates the creation of ICSs with an acceptable level of security.

1. Introduction

Industrial Control Systems (ICSs) control and monitor industrial processes in critical sectors such as energy (e.g., nuclear plant, smart grid, and oil and gas), chemical, transportation, pharmaceutical, food and beverage, and discrete manufacturing (e.g., automotive, aerospace, and durable goods) (Stouffer et al., 2015a). They are complex systems of systems that use proprietary software, network protocols, specialized standards, and merge Information Technology (IT) with Operation Technology (OT). In the past, ICSs networks used to be isolated, and cyber security was not considered in the early stages of system development. However, a major challenge is currently facing ICSs, due to the fast expansion in the usage of IIoT as a part of Industry 4.0.

ICSs networks are no longer such isolated networks. They are mostly connected to the internet due to functionality and system maintenance requirements. Unlike IT, where the confidentiality of sensitive information is mostly the priority, safety, and operation continuity are fundamentals for OT, which may create conflicts between system requirements and cyber-security. For example, encrypted network protocols might cause latency in the OT network, which could severely impact operational safety. Also, encryption might require replacing old network components with new ones that support such a feature. However, the life cycle of an ICS mostly exceeds 10 years (Stouffer et al., 2015b), and major system component replacement is not financially foreseen. Such paradigm shifts between IT and OT security are major challenges in ICSs security.

To successfully carry out a high-impact industrial control system attack, an attacker must first acquire system access, then exploit a vulnerability. Yet, the likelihood of success of such an attack is reduced if the vulnerability is newly discovered (Allodi and Etalle, 2017). Targeted attacks against ICSs are more likely to succeed, even with new vulnerabilities, as they are often carried out by well-resourced attackers (e.g., nation-state actors, terrorist organizations, and cyber-criminal groups). The 2016 and 2022 Ukrainian power grid attacks, which used the Industroyer malware, showed a sophisticated example of targeted malware that was specially designed to cause power grid damage by abusing ICSs protocols and commands (Cherepanov, 2017; Research, 2022).

[*] Corresponding author.
E-mail address: Shaymaa.Khalil@Taltech.ee (S.M. Khalil).

https://doi.org/10.1016/j.cose.2023.103543
Received 18 May 2023; Received in revised form 28 September 2023; Accepted 16 October 2023
Available online 20 October 2023

The increasing number of cyberattacks on industrial control systems (ICSs), especially those in critical infrastructure,[1] underscores the need to develop secure-by-design systems that have an acceptable level of security. Threat modeling (TM) is a process of identifying and analyzing potential threats to a system. It is an important step in the secure-by-design process because it helps to identify security requirements early in the development life cycle. This can help to prevent security vulnerabilities from being introduced into the system. However, addressing security in the later stages of development can result in significant engineering expenses or unaddressed critical vulnerabilities. Such potential consequences are more evident in ICSs due to their complex supply chain cycles and less tolerance to major system updates.

In addition to being a task in the development life cycle, threat modeling studies constitute a significant human interaction medium for different development parties at the early stages (e.g., software and hardware developers, system architects, and security experts). The relevant parties can establish common ground in overall security considerations, discuss specific security requirements, and solve potential conflicts between system and security functions in this task.

In the digital world, threat modeling was first introduced for securing software during the design phase, as a part of the secure software development life cycle (SDLC). The use of threat modeling techniques was then extended to cover ICSs as well. Threat modeling is currently included in some ICSs standards. For example, the ISA/IEC 62443 standard[2] discusses threat modeling in the secure product development life cycle requirements (ISA 62443-4-1). However, standards do not provide specific procedures on how to perform threat modeling for ICSs.

Compared to software-based systems, the components of ICSs interact and change the physical space, which makes predicting the physical consequences of cyber attacks more challenging in the threat modeling of such systems. Physical and cyber attack vectors may be interleaved immensely in an attack campaign, making it difficult to dissect the system assumptions and agree on a common ground for the security requirements.

This paper aims to provide an overview of the state of the art of ICSs threat modeling and identify the related research gaps using a Systematic Literature Review (SLR). The research aims to answer the following Research Questions (RQs):

RQ1: What are the proposed threat modeling methods for ICSs in the literature?

RQ1.1: How could we categorize the proposed ICSs threat modeling methods?

RQ1.2: What are the ICSs threat modeling phases that are covered in the literature?

RQ2: What are the identified research gaps in the proposed ICSs threat modeling methods?

This study provides a comprehensive overview of threat modeling for industrial control systems (ICSs). It begins with an extensive background section that introduces readers to the different concepts and terminology related to threat modeling. The study then analyzes a selection of papers on threat modeling for ICSs, based on a proposed taxonomy of ICSs threat modeling and identified threat modeling stages. The study also discusses the state of the art in threat modeling for ICSs, including research gaps. This study is therefore of interest to academic researchers and practitioners who are researching or applying threat modeling to ICSs. To our knowledge, no other study has provided such a comprehensive analysis of threat modeling methods within the scope of ICSs.

The rest of this paper is organized as follows: Section 2 provides background about TM and information that eases the understanding of the topic, especially for readers with limited knowledge about the domain. The background section includes TM definitions and discusses the difference between TM, risk assessment, and attack modeling. Section 2 also introduces STRIDE, the application of TM within the development life cycle, as well as the privacy, safety, and cyber-security focus within TM. The background section concludes with TM tools, gamification, and challenges. Section 3 discusses the related work. Section 4 introduces the research methodology, including the paper selection process, exclusion and inclusion criteria. The categorization criteria, Section 5, are described by an ICSs TM taxonomy and the TM stages. Results of the SLR can be found in Section 6, which includes general information about the selected papers and analyzes them based on the proposed ICSs TM taxonomy and TM stages. The research gaps and future research directions are discussed in Section 7, while Section 8 discusses the threats to the validity of this study. The SLR is concluded by Section 9.

2. Background

This section provides an overview of the concepts of threat modeling, which can help readers get familiar with the topic.

2.1. Threat modeling definitions

The literature shows that the *"definitions of threat modeling are numerous, and used in many different and perhaps also incompatible ways"* (Xiong and Lagerström, 2019). One of the threat modeling definitions promoted by Xiong and Lagerström (2019) is: *"Threat modeling is a process that can be used to analyze potential attacks or threats, and can also be supported by threat libraries or attack taxonomies"*. Nweke and Wolthusen (2020) define TM as *"a systematic approach for characterizing potential threats to a system. It ensures completeness by including the prioritization of threats and mitigation based on probabilities, business impacts, and cost of countermeasures"*.

Threat modeling is a practical process with definitions also coming from the industry and non-academic research. As a part of the Security Development Life cycle (SDL),[3] Microsoft defines threat modeling as *"a practice that allows development teams to consider, document, and (importantly) discuss the security implications of designs in the context of their planned operational environment and in a structured fashion"*. The threat modeling manifesto, created in 2020 by a group of experts from both academic and corporate fields,[4] defines threat modeling as *"analyzing representations of a system to highlight concerns about security and privacy characteristics"*. However, we find that the last definition is more relevant to software TM, as in the context of ICSs, threat modeling could also highlight concerns about the safety characteristics of the system besides its security and privacy. The manifesto also identifies four questions that should be asked during the threat modeling process: (1) What are we working on? (2) What can go wrong? (3) What are we going to do about it? (4) Did we do a good enough job?

2.2. Threat modeling and risk assessment

The term *"risk assessment"* is often used alongside *"threat modeling"* in the literature while referring to threat modeling methodologies. The integration of risk assessment techniques into the threat modeling process helps in better threat prioritization and obtaining more tangible results, as proposed by Maheshwari and Prasanna (2016). However, we noticed that there is no clear distinction between the two terminologies, as they seem to refer to similar concepts in some specific cases, such as asset-centric threat models (Section 5.1.2).

To understand the differences between the two terminologies, we have to distinguish between a *"threat"* and a *"risk"*. Based on NIST guide

[1] https://www.microsoft.com/en-us/security/business/microsoft-digital-defense-report-2022.

[2] https://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series-of-standards.

[3] https://www.microsoft.com/en-us/securityengineering/sdl.

[4] https://www.threatmodelingmanifesto.org/.

to industrial control systems security (Stouffer et al., 2015b), a threat is defined as *"any circumstance or event with the potential to adversely impact agency operations (including mission, functions, image, or reputation), agency assets, or individuals through an information system via unauthorized access, destruction, disclosure, modification of information, and/or denial of service"*. The same document defines the risk as *"the level of impact on agency operations (including mission, functions, image, or reputation), agency assets, or individuals resulting from the operation of an information system, given the potential impact of a threat and the likelihood of that threat occurring"*.

Stouffer et al. (2015b) also defined the risk assessment as *"the process of identifying risks to operations, assets, or individuals by determining the probability of occurrence, the resulting impact, and additional security controls that would mitigate this impact"*. Risk assessment is a part of the risk management process, which includes three phases: (i) the conduct of a risk assessment; (ii) the implementation of a risk mitigation strategy; and (iii) the employment of techniques and procedures for the continuous monitoring of the security state of the information system.

Risk assessment provides a clear emphasis on the impact analysis and prioritization of the risk findings, whereas threat models may not necessarily include such constructs. For instance, STRIDE focuses on finding the threats without using an impact terminology and ranking approach, thus, it requires an additional framework (e.g., DREAD) to complement that gap. However, lightweight threat models embed the risk assessment stage into their threat elicitation process to reduce the workload of experts. This can be done by focusing on critical information flows (Tuma et al., 2017), or eliminating threats with lower likelihood and impact (Wuyts et al., 2020). Some research has also proposed incorporating the risk notion directly into threat models (Sion et al., 2018).

The NIST guide for risk assessment (Force, 2012) specifies that risk assessment should be conducted throughout the system development life cycle, starting from the development phase. The guide also states that risk assessment includes identifying threat sources (e.g., adversarial, trusted insider), determining threat events at the level of tactics, techniques, and procedures, and attack graph techniques. We can conclude that threat modeling can also be considered as a form of risk assessment, especially in situations where the risk is calculated. This conclusion is consistent with the NIST definition of threat modeling as *"a form of risk assessment that models aspects of the attack and defense sides of a logical entity, such as a piece of data, an application, a host, a system, or an environment"* (Force, 2017).

2.3. Threat modeling versus attack modeling

The term, *"attack modeling"*, is often used in the literature to refer to the process of identifying security risks from the attacker's perspective (Sequeiros et al., 2020). This does not necessarily distinguish it from the attack-centric threat modeling (Section 5.1.2). However, a nuanced point would be that threat modeling effort always considers the implications of attacks to the defense actions, whereas attack modeling may solely focus on attacker behavior, not necessarily the defense perspective.

Threat models typically include a representation of the target system, such as a data flow diagram, and utilize it as a systematization construct for threat elicitation. Attack models, on the other hand, may not include such a representation. For instance, the intrusion kill chain (Hutchins et al., 2011) describes the stages of an attack, and MITRE ATT&CK framework (Alexander et al., 2020) presents the tactics and techniques that can be utilized in each stage of an attack. Both of these models focus on the attacker's actions, without explicitly outlining the target system. The Diamond Model takes a more detailed approach, by examining the capabilities and infrastructure of the attackers (Caltagirone et al., 2013).

The distinction between attack trees, and attack graphs (Lallie et al., 2020), can help us understand the difference between threat modeling and attack modeling. Although attack trees and attack graphs seem close to each other, as they use some graph notations, their practical usage is quite varied in terms of the main purpose.

Attack graphs are a more defensive approach to modeling attacks. They incorporate system knowledge, such as network access control rules and vulnerability lists, to identify possible attack paths to a victim system. Attack trees, on the other hand, are a more offensive approach to modeling attacks. They describe attack actions in a free-form text, without considering the system model. Although the experts can embed some knowledge about the target system and vulnerabilities to the nodes, this is totally up to those experts and done in an unsystematic way. Thus, an attack graph could be considered to be more closely aligned with threat modeling. Attack trees constitute the main method in various attack modeling studies and are also utilized as a complementary construct to known threat modeling approaches (e.g., STRIDE), as shown by Ahn et al. (2021).

2.4. STRIDE

The most mature threat modeling method, STRIDE, is an industry-developed threat modeling method. It was first introduced by Microsoft, as part of the Microsoft security development life cycle, *"to identify various types of threats the product is susceptible to during the design phase"*,[5] and has been used by Microsoft and other known organizations for decades.

STRIDE is a threat enumeration technique that helps to identify potential threats to a system. It is a mnemonic for **S**poofing, **T**ampering, **R**epudiation, **I**nformation disclosure, **D**enial of service, and **E**levation of privilege. STRIDE analyzes the system components against 6 main security properties (*authenticity*, *integrity*, *non-repudiation*, *confidentiality*, *availability*, and *authorization*). This method is complemented by a system modeling approach, called a Data Flow Diagram (DFD). A DFD is a graphical representation of a system, and it is composed of four basic elements: processes, data stores, data flows, and external entities. DFD can be used to understand the system's architecture and what information is exchanged between the different components of a system.

STRIDE has two variants: STRIDE-per-Element and STRIDE-per-Interaction. STIRDE-per-Element focuses on identifying threats to each element of the DFD. STRIDE-per-Interaction enumerates threats based on the origin, destination, and interaction of data flows (Adam Shostack, 2013). Several studies have proposed combining STRIDE with other known security models. For example, a study by Ahn et al. (2021) proposes combining STRIDE with attack trees, which helps in better understanding the threats to a system.

2.5. TM application phase of the SDL

Threat modeling was first proposed to be applied in the early stages of the Software Development Life Cycle (SDLC), as one of the constructs of applying the security-by-design principles. As mentioned in Section 2.4, STRIDE was proposed as part of Microsoft SDL, which should occur in parallel to the SDLC. Microsoft SDL[6] starts with a pre-requirement of training, followed by five phases: requirements, design, implementation, verification, and release phases. Microsoft SDL proposes performing TM at the design phase. However, some research has proposed the use of TM in other phases as well.

For example, Marksteiner et al. (2019) propose the use of threat modeling in automating the testing process for the system. The use of threat modeling techniques can also be extended to other phases in a system life cycle, such as the verification, release, and operation phases. For instance, assessing the system security, planning penetration testing

[5] https://www.microsoft.com/security/blog/2009/08/27/the-threats-to-our-products/.

[6] https://www.microsoft.com/en-us/download/details.aspx?id=12379.

(Salzillo et al., 2020; Rak et al., 2020), or prioritizing the system patching could benefit from TM. PatrIoT is a penetration testing methodology based on TM (Süren et al., 2023). TM is also promoted as a continuous process that should be considered in every design modification, not a one-time effort.

2.6. Privacy threat modeling

The privacy concept was introduced in an academic paper published in 1890. The paper defines privacy as *"the right to be let alone"* (Goldsmith et al., 1890). Due to its value, privacy has also been identified as one of the human rights acts.[7] There are many regulations related to data protection and privacy, including the ones related to specific sectors such as the Health Insurance Portability and Accountability Act (HIPAA),[8] and the Payment Card Industry Data Security Standard (PCI-DSS).[9] The General Data Protection Regulation (GDPR)[10] was put into effect in 2018, imposing obligations to protect the privacy of EU citizens.

As some systems, including ICSs, may store personal data, there is a significant need for threat modeling approaches that cover privacy aspects. LINDDUN is an acronym for the terms **L**inkability, **I**dentifiability, **N**on-repudiation, **D**etectability, information **D**isclosure, content **U**nawareness, policy and consent **N**on-compliance (Deng et al., 2011). The method shares a similar approach to STRIDE and was introduced as a framework to model privacy-specific threats for software-based systems. This framework categorizes privacy properties into *hard* and *soft* privacy. Hard privacy aims to conduct data minimization to reduce the need to trust the data controller, whereas soft privacy addresses data security countermeasures assuming that the data subject trusts the data controller.

Restricted Misuse Case Modeling (RMCM) is another threat modeling method that addresses the security and privacy requirements and was introduced in an industrial healthcare software case study (Mai et al., 2018). RMCM adopts a method named "RUCM" (Restricted Use Case Modeling) and extends it to meet security and privacy requirements.

2.7. Safety versus cyber-security in ICSs

In the past, only accidental component failures (hazards) were considered in the design of ICSs. However, the increasing use of communication technologies and open protocols, such as TCP/IP, have created more vulnerable systems to attacks launched in cyberspace. Cyber-attacks can cause incidents that may induce physical consequences, including human losses and property damages. Therefore, the interdependencies between the security and safety aspects should be considered during the threat analysis of ICSs (Kriaa et al., 2015).

The main difference between safety and security is that safety focuses on hazardous actions that occur unintentionally, whereas security threats are caused by a threat actor (sometimes called a threat agent) with malicious intentions. Threat modeling might consider both safety and security aspects in the system analysis. For instance, Suo et al. (2018) provide a STRIDE-based threat modeling method that was applied to an automated-driving system, and analyzes both safety and cyber security aspects.

It is worth noting that the European Parliament recently proposed a machinery regulation[11] that ensures the safety of machinery and robots. The proposed regulation states a difference between cybersecurity (a broad term related to networks) and cyber safety (related to machines).

2.8. Known threat modeling tools

There exist free and commercial threat modeling tools that help in automating the threat elicitation process. Free TM tools include Microsoft Threat Modeling Tool, OWASP Threat Dragon,[12] and Cairis.[13] While commercial TM tools include IriusRisk,[14] Tutamen Threat Model Automator,[15] SecuriCAD,[16] and Threatmodeler.[17] Commercial tools mostly support a community option that allows a single threat modeling trial.

The mentioned tools were designed mainly for software threat modeling, while some of them have expanded their scope to cover IoT and Cyber-Physical Systems (CPSs) as well. For instance, Microsoft Threat Modeling Tool provides the option to create customized threat modeling templates, which could be used to add different ICSs templates. The tool also supports a *"Medical Device Model"* template[18] that includes the representations of some cyber-physical devices related to the medical sector, similar to sensors and ionizing radiation source devices.

2.9. Threat modeling gamification

TM is an interdisciplinary process, that might be complex for persons with limited cyber security background. Tarandach and Coles (2020) list some threat modeling card games that could be used as an educational tool for teaching and practicing threat modeling. Some of the tools are based on STRIDE, similar to Elevation of Privilege[19] and Elevation of Privacy.[20] Other games do not follow a known threat modeling methodology similar to OWASP Cornucopia[21] and Security and Privacy Threat Discovery Cards.[22] The LINDDUN Team has created a card game called LINDDUN GO,[23] which is a lightweight version of LINDDUN that could be used by beginners.

2.10. Threat modeling in practice

Interviews with security experts who practice CPSs threat modeling in real life have shown that there is a need to enhance the relevant threat modeling methods (Jamil et al., 2022). According to the study, practitioners use a combination of threat modeling methods, approaches, and standards for CPSs threat modeling. However, this suggests that there is a lack of practical and systematic threat modeling methods for CPSs that suit the practitioners' needs. Also, this indicates a gap between threat modeling practices and academic research, as there are many models offered in the literature, while practitioners might not be fully aware of them.

Bernsmed and Jaatun (2019) discuss threat modeling challenges and best practices based on semi-structured interviews with experts that conduct threat modeling in agile software development for four different organizations. The interviews show that threat modeling is a time-consuming process, it is difficult to identify the relevant threats, software developers do not have the motivation to assign time for security, and the experts are not clear about when to stop the modeling process.

Soares Cruzes et al. (2018) identified 21 challenges faced by an organization while applying STRIDE within five agile development projects,

[7] https://www.echr.coe.int/documents/guide_art_8_eng.pdf.
[8] https://www.hhs.gov/hipaa/index.html.
[9] https://www.pcisecuritystandards.org/pci_security/.
[10] https://gdpr.eu/tag/gdpr/.
[11] https://ec.europa.eu/commission/presscorner/detail/en/ip_22_7741.
[12] https://owasp.org/www-project-threat-dragon/.
[13] https://cairis.org/.
[14] https://www.iriusrisk.com/threat-modeling-platform.
[15] https://www.tutamantic.com/page/features.
[16] https://foreseeti.com/securicad-enterprise/.
[17] https://threatmodeler.com/.
[18] https://github.com/microsoft/threat-modeling-templates.
[19] https://www.microsoft.com/en-us/download/details.aspx?id=20303.
[20] https://github.com/F-Secure/elevation-of-privacy.
[21] https://owasp.org/www-project-cornucopia/.
[22] https://securitycards.cs.washington.edu.
[23] https://www.linddun.org/go.

and also mapped the challenges found during the study to what was previously found in the literature. These challenges include challenges related to the threat modeling process, similar to the difficulty of drawing data flow diagrams, requiring security experts to run the threat modeling meetings, deciding when the analysis process is enough to stop, and engaging everyone in the discussions. The research team has also observed a decrease in motivation for security activities at the late threat modeling stages, similar to the threat identification stage. The research reasons this behavior by mentioning that in the initial asset identification stage meetings, the team was motivated for learning. However, while they advanced in the project to reach the threat identification stage, the meetings were oriented to document the threats, so they were not very engaging.

The threat modeling process might include the use of a threat library or attack taxonomy to enhance the efficiency of the process. While it is argued that threat libraries limit the possibility of discovering new threats, as the findings might be limited to special architectures and cannot be applied to different system architectures (Uzunov and Fernandez, 2014). On the other side, threat taxonomies are mostly at a high abstract level and require good security knowledge to identify relevant system threats. Through this literature review discussion section, we also discuss the TM challenges noticed in the selected studies that addressed ICSs TM.

3. Related work

To our knowledge, this is the first systematic literature review on threat modeling that focus on ICSs and analyzes the studies from the control systems perspective. Other systematic literature reviews have addressed TM in a more general context. Tuma et al. (2018) SLR focuses on threat modeling of software systems and analyze 25 threat modeling methodologies. The paper's main findings are: the analysis procedures are not precisely defined, there is a lack of quality assurance of the analysis outcomes and tool support, and validation is limited. Another SLR about threat modeling is presented by Xiong and Lagerström (2019), without limitation to a specific application domain. The review spots that threat modeling definitions are various and the domain is missing common ground.

The SLR of Ling et al. (2020) concludes six sources of information to be used in power systems threat modeling as follows: expert knowledge, logs and alerts, previous research, system's state, vulnerability scoring and databases, and vulnerability scanners. This paper reviews only the information sources that can be used for TMs but does not provide an in-depth discussion about the TM methods.

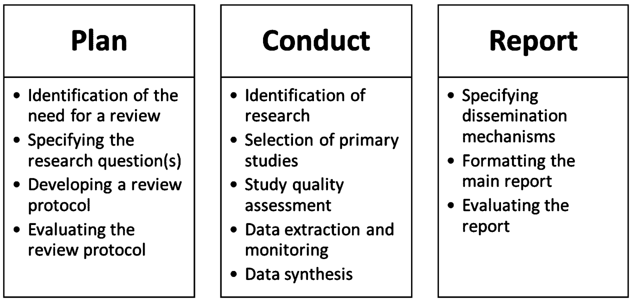
Luo et al. (2021) present an SLR of Threat Analysis and Risk Assessment (TARA) for Connected Vehicles. Kriaa et al. (2015) present a survey related to risk assessment methods that combine both safety and security aspects in ICSs. The paper does not use the *"threat modeling"* terminology while referring to many standards and risk assessment methods that focus on interdependencies between safety and security in ICSs. This survey is relatively old (i.e., published in 2015), which means recent papers regarding ICSs were not covered.

Nweke and Wolthusen (2020) review four asset-centric threat modeling approaches for software design (DREAD, Trike, OCTAVE, and PASTA). Shevchenko et al. (2018a) summarize twelve methods of threat modeling (STRIDE and its Derivations, PASTA, LINDDUN, CVSS, Attack Trees, Persona non Grata, Security Cards, hTMM, Quantitative Threat Modeling Method, Trike, VAST modeling, and OCTAVE). The study includes methods similar to CVSS and attack trees, which we do not agree on categorizing as threat modeling methods by themselves, while they might be used within threat modeling methods. Shevchenko et al. (2018b) continue the work done in Shevchenko et al. (2018a) and evaluate the twelve threat modeling methods based on five criteria: 1) Strengths and weaknesses (maturity and usage, focus/perspective, time to implement, effectiveness and mitigation strategies), 2) Adoptability (easy to use, easy to learn, and documentation and support), 3) Tailorability (integration with SDLC, compatibility with the agile development process, and scalability), 4) Applicability (coverage of safety-security interdependency, and integration of hardware and software threats), and 5) Automation (availability of tools, integration options for tools into an SDLC, portability of tools). However, the evaluation criteria for each method do not follow an approach with special metrics. For example, the evaluation of the "time/effort" aspects, which should reflect how the method is time-consuming, does not have clear criteria on how Shevchenko et al. (2018a) reached the mentioned results, the same applied to other aspects similar to "consistent results". Shevchenko et al. (2018a) recommend using the PASTA threat modeling method due to its detailed guidance through the modeling process and its flexibility. The research also proposes some additional steps while applying PASTA on a CPS, to cover more aspects related to security, safety, and privacy. However, the study did not demonstrate its effectiveness by applying it to a case study, which leaves the readers with unanswered questions about the procedures of applying PASTA to CPSs.

The previous review papers do not provide a clear and detailed overview of the proposed ICSs threat modeling methods in the literature. In addition to specific analysis within the ICSs context, this literature review provides detailed background about the field, puts special emphasis on threat modeling stages, and proposes a taxonomy to categorize ICSs threat modeling methods.

4. Research methodology

To answer the research questions discussed in Section 1, we followed the guidelines for systematic literature review proposed by Kitchenham et al. (2007), as shown in Fig. 1.



Plan	Conduct	Report
• Identification of the need for a review • Specifying the research question(s) • Developing a review protocol • Evaluating the review protocol	• Identification of research • Selection of primary studies • Study quality assessment • Data extraction and monitoring • Data synthesis	• Specifying dissemination mechanisms • Formatting the main report • Evaluating the report

Fig. 1. SLR stages proposed by Kitchenham et al. (2007).

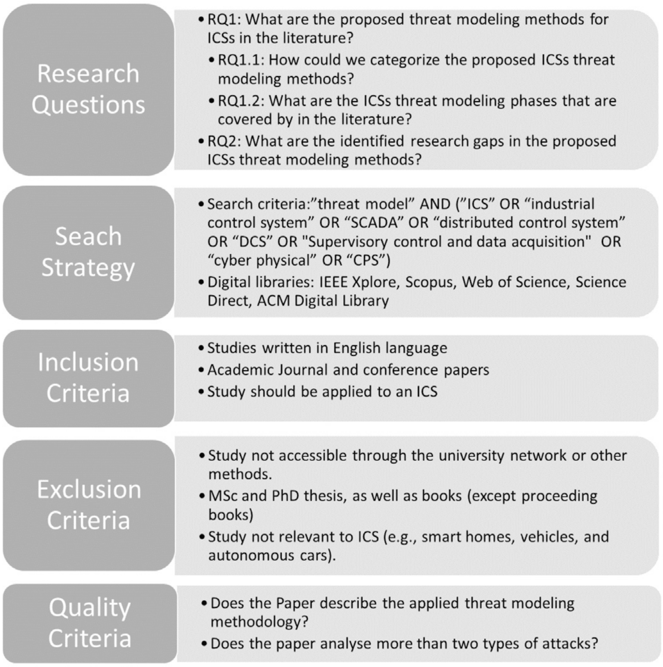
4.1. *Planning the review*

This SLR was initiated in 2021, as our team was eager to understand the state of the art of ICSs TM. We first reviewed the related studies discussed in Section 3. However, these studies did not provide us with a clear understanding of the current TM methods that are used in ICSs, while considering the unique nature of such CPSs.

As our team was involved in the secure design process of a power system, it was important for us to find a well-documented, simple, and practical ICSs TM methodology that we could adapt to our real-world project. For this reason, we created an ICSs TM taxonomy and clearly described the list of TM stages that we identified in the literature. We believe that this categorization could help the readers to better understand the strengths and weaknesses of each of the selected ICSs TM approaches. Although the categorization of threat modeling approaches based on taxonomy and applied stages is not a standard practice in systematic literature reviews, we believe that it is a valuable contribution to this study. Therefore, we have separated the categorization criteria in a separate section, Section 5.

We also decided to include an extensive and comprehensive background to this SLR, where we introduce to the reader different TM concepts, as we found a clear gap in knowledge of these concepts in other similar SLRs. This SLR was written over a period of two years, during which we performed TM in practice in parallel. Such experience greatly enhanced the research quality and helped us to better understand the identified gaps in the current literature.

We have iteratively modified and enhanced the review protocol throughout the research process to improve the completeness of the results. Due to resource constraints, the review protocol was discussed with the second author and continuously enhanced based on trial results. For example, the initial search for this SLR was conducted using the search criteria:(“threat model” AND (“ICS” OR “industrial control system” OR “SCADA” OR “supervisory control and data acquisition” OR “distributed control system” OR “DCS”)). The asterisk (*) wildcard was used to expand the keywords *model*, and *system* to include different variations of these keywords, such as *threat modeling* and *threat modelling*. This helped us to identify a broader range of papers that were relevant to our research. However, this search criteria returned a limited number of studies (as of May 2021, it returned 37 papers, of which only 11 were kept for full reading). Therefore, during the same month, we iteratively expanded the research keywords to include the *cyber physical, cyber-physical* and *CPS* keywords, which resulted in 44 papers for full reading. This confirms that adding the CPS-related keywords helped us to identify more studies that are relevant to our review. Fig. 2 summarizes the final version of the review protocol used for this SLR.


Research Questions

- RQ1: What are the proposed threat modeling methods for ICSs in the literature?
  - RQ1.1: How could we categorize the proposed ICSs threat modeling methods?
  - RQ1.2: What are the ICSs threat modeling phases that are covered by in the literature?
- RQ2: What are the identified research gaps in the proposed ICSs threat modeling methods?

Seach Strategy

- Search criteria:”threat model” AND (”ICS” OR “industrial control system” OR “SCADA” OR “distributed control system” OR “DCS” OR "Supervisory control and data acquisition" OR “cyber physical” OR “CPS”)
- Digital libraries: IEEE Xplore, Scopus, Web of Science, Science Direct, ACM Digital Library

Inclusion Criteria

- Studies written in English language
- Academic Journal and conference papers
- Study should be applied to an ICS

Exclusion Criteria

- Study not accessible through the university network or other methods.
- MSc and PhD thesis, as well as books (except proceeding books)
- Study not relevant to ICS (e.g., smart homes, vehicles, and autonomous cars).

Quality Criteria

- Does the Paper describe the applied threat modeling methodology?
- Does the paper analyse more than two types of attacks?

Fig. 2. SLR Search protocol.

4.2. Conducting the review

The first round of searches was conducted on five selected search libraries in May 2021, namely IEEE Xplore, Scopus, Web of Science, Science Direct, and ACM Digital Library. The scope of the search included the title, abstract, and keywords. This search returned 44 papers for full reading. The search was repeated in February 2022, which resulted in an additional 8 papers for full reading. A third search was conducted in June 2023, which resulted in another 32 papers for full reading. This made a total of 84 papers that were fully read through the search criteria mentioned in Fig. 2.

4.2.1. Selection process

On June 30, 2023, we conducted a third search of the five selected search libraries. This search returned a total of 288 papers, as shown in Fig. 3. After eliminating duplicate papers, we were left with 162 papers. We then compared this list with the lists from the previous search iterations and found that seven studies were missing. We manually added these studies to the list, bringing the total number of papers to 169. However, we were only able to access 162 studies.

We proceeded to read the abstracts of the 162 accessible papers. We had already read the abstracts of 105 of these papers in previous search iterations, so we mainly read 57 new abstracts. We used the inclusion-exclusion criteria to exclude papers that did not match our inclusion criteria from an abstract read. We conducted full paper readings for the papers that remained after reading the abstracts.

We were left with 106 papers for full paper reading, of which 84 came from the search criteria and 22 came from snowballing, as well as looking into other threat modeling research groups’ publications. Fig. 3 shows the results of the search criteria, as well as the number of identified papers through the different filtration stages. We selected 36 papers that met the inclusion, exclusion, and quality criteria. Of these, 27 papers were found through the systematic literature review search, and 9 papers were found through snowballing.

4.2.2. Exclusion, inclusion, and quality criteria

This SLR excluded papers that do not allow access through the network of Tallinn University of Technology and that we could not access using other methods, such as contacting the authors. We also excluded master’s and PhD theses, as well as irrelevant papers to ICSs such as those related to smart homes, vehicles, and autonomous cars as application areas for threat modeling.

To ensure that the papers included in the research propose a systematic ICSs TM methodology that the readers can apply, the quality criteria questions, mentioned in Fig. 2, had to be answered by “yes” in order to include the research in this SLR analysis. Papers discussing general information about security concerns in ICSs, without proposing a threat modeling methodology, were excluded. Additionally, papers that analyzed only one or two types of attacks were excluded because threat modeling, when applied during the design phase, should strive for completeness by considering a wide range of potential threats.

This research includes threat modeling academic papers that are written in English and focus on ICSs. The included papers analyze systems with different control components, such as Programmable Logic Controllers (PLCs), Remote Terminal Units (RTUs), and Intelligent Electronic Devices (IEDs). They also may have sensors, actuators, and other similar components that might be found in a typical ICS. The included papers discuss at least three different types of threats (e.g., False Data Injection (FDI), Man in the Middle (MITM), and Denial of Service (DOS) threats) to provide a threshold for completeness of the threat analysis. The papers included in this research are either the ones that apply or propose a threat modeling methodology, a TM Language, or a tool for automating the threat modeling process.

5. Categorization criteria

The selected 36 papers were categorized based on a proposed threat modeling taxonomy; then, each study was mapped to the threat modeling phases that were applied by it. This section introduces the categorization criteria.

5.1. ICSs threat modeling taxonomy

Based on the literature, we identified various aspects to categorize the ICSs threat modeling studies and aligned them in a descriptive taxonomy, as shown in Fig. 4. The taxonomy categorizes papers based on their *method type, approach, focus, validation method*, and *application area*. Each of these aspects is explained in the following sub-sections.

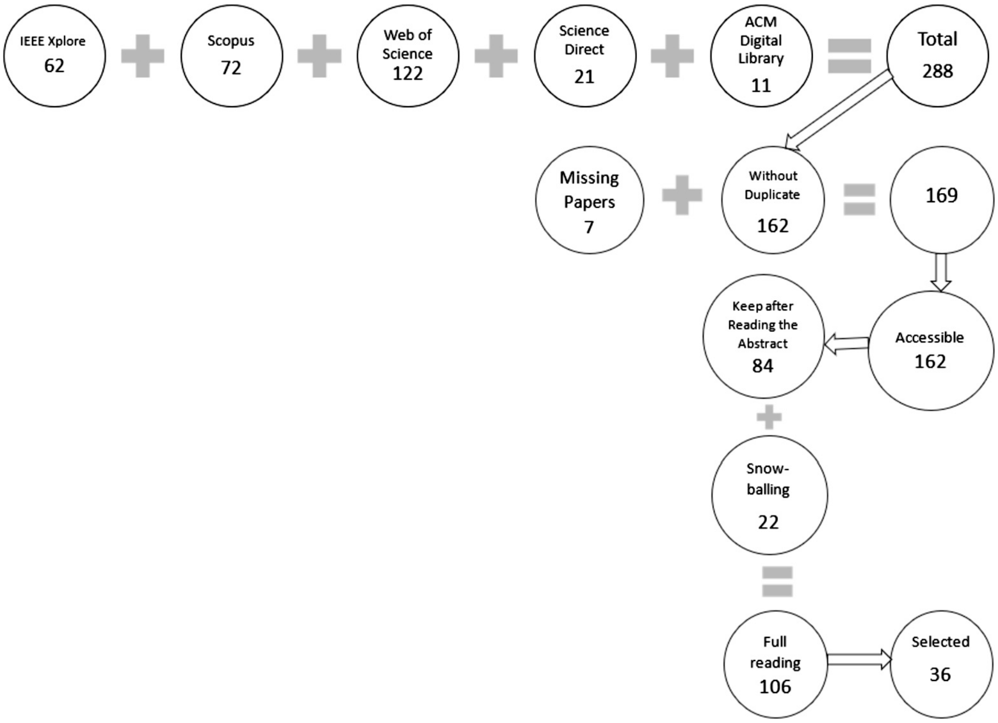


Fig. 3. Number of studies through the selection process.

5.1.1. *Method type*

As threat modeling is a broad terminology, it might be done differently, depending on the stakeholders' needs. We divided the TM methods types into five subcategories: *formal, semi-formal*, and *non-formal* methods, *modeling language*, and *tool.*

Formal method Formal threat modeling methods propose a TM approach that is fully based on mathematics. However, it is challenging to find a formal threat model that has a complete representation of CPSs threats, as considering software, hardware, and physical threats simultaneously is a challenge (Burmester et al., 2012). Such methods require specific mathematical knowledge, which makes them more suitable for academic research, while they are still relatively unpopular in practice. Formal methods could also be used to create threat modeling software that automates the TM process.

Non-formal method This category does not resort to formal representations. Non-formal TM methods mostly depend on the involvement of experts in every stage of the threat modeling process. For example, the experts create the ICSs system mapping using data flow diagrams, attack trees, or Petri-nets. In non-formal methods, the threat enumeration is done by the experts, and they decide how to prioritize the threats. They choose the prioritization criteria based on their needs and knowledge, without any advanced tool support or mathematical calculations. Such methods tend to analyze several types of threats (e.g., STRIDE analyzes 6 types of threats), and they are widely used in organizations, as they are easy to understand by different stakeholders.

Semi-formal method This SLR categorizes semi-formal TM methods as methods that combine formal and non-formal modeling constructs in a single method. In semi-formal methods, threats might be elicited by the team using a non-formal method (e.g., STRIDE), then the risk might be calculated using a formal method for calculating the risk of the identified threats. Within the context of this SLR, we considered combining non-formal methods with a Matlab simulation as a semi-formal method, as such simulation is based on a mathematical representation of the system.

Modeling language Threat modeling languages are mostly domain-specific modeling languages (DSLs) that map system components, their interactions, security constraints, requirements, or threats using a specific language, such as UMLsec or SecureUML (Johnson et al., 2018). DSLs can be used as a basis for creating TM tools.

Tool Threat modeling tools aim to automate the ICSs TM process. Examples of TM tools were discussed in Section 2.8.

5.1.2. *Approaches*

In general, threat modeling has three widely known approaches, *asset-centric, attack(er)-centric*, and *system-centric* (also named software-centric). The literature includes a fourth approach, the *data-centric* approach, which focuses on *"protecting particular types of data within systems"* (Souppaya and Scarfone, 2016). However, the proposed ICSs TM taxonomy excludes the data-centric threat modeling approach, as from its description, it does not seem to analyze the impact of threats on the physical system, which makes it more suitable for IT systems, not ICSs.

Asset-centric This approach only focuses on valuable assets during the threat analysis. As per Adam Shostack (2013), assets could be described in 3 ways: *a "thing" targeted by attackers, a "thing" that needs to be protected, or a "stepping stone" for these two*. The term asset does not only refer to system components (tangible assets), but it also includes business assets (intangible assets), such as the organization's image or reputation. Asset-centric threat modeling gives a practical perspective for the TM, as it also studies the impact of losses on the business. In this approach, critical assets are first identified, then mapped in a diagram to show interconnections between the different assets (Adam Shostack, 2013).

DREAD (Howard and LeBlanc, 2002), Trike (Saitta et al., 2005), OCTAVE (Alberts et al., 2003), and PASTA (UcedaVelez and Morana, 2015) are well-known asset-centric threat modeling methods. The review of these four threat modeling approaches, provided by Nweke and Wolthusen (2020), shows that they all contribute to risk management and adds that an asset-centric approach is sometimes referred to as a risk-based threat modeling approach. From the literature, we noticed

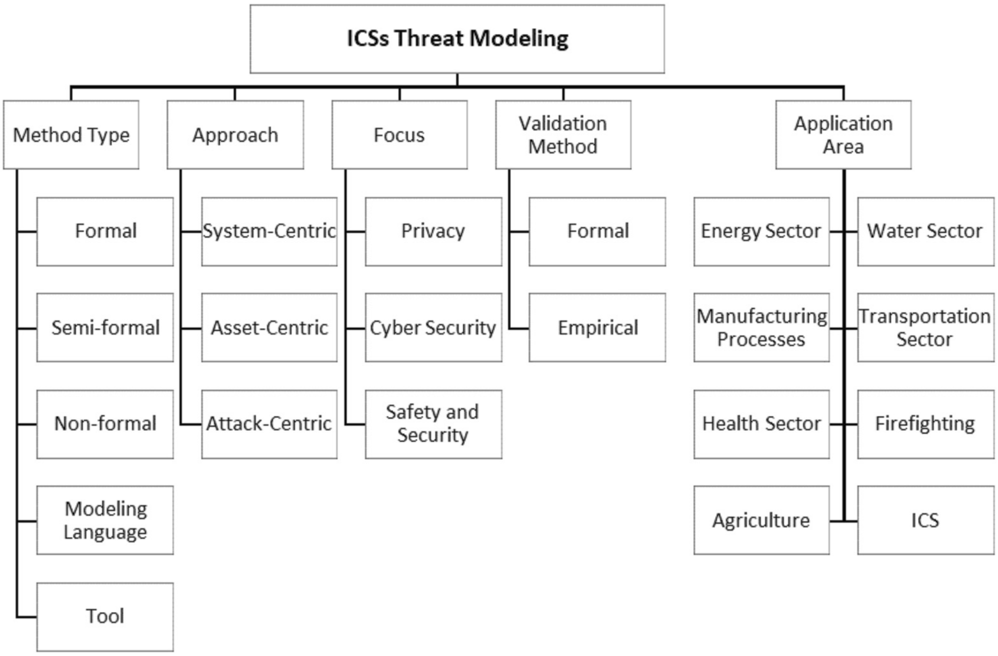


Fig. 4. Proposed ICSs threat modeling taxonomy.

that asset-centric threat modeling methods are sometimes referred to as risk assessment methods. The term *threat modeling* may not be used in papers that discuss this approach.

Attack(er)-centric This threat modeling approach focuses on the attacker's perspective. The description of this approach varies in the literature. Adam Shostack (2013) describes it as an approach that identifies the attacker types, similar to "personas" for threat agents. On the other hand, Schlegel et al. (2015) mentions that this approach also describes how the attacker would get into the system. We also noticed that Withers (2016) use both terms attack-centric and attacker-centric to describe this approach.

In general, the attack(er)–centric approach tends to identify the types of attackers that may target the system, such as state-funded groups or cyber-criminals. It also explores their objectives, resources, capabilities, and other aspects that might impact the attack results. For example, an attacker's objectives could be to steal data, disrupt operations, or damage the system. Their resources could include financial resources. Their capabilities could include the ability to exploit known vulnerabilities or perform complex zero-day attacks. However, such an approach might also use attack graphs (Withers, 2016), or attack trees (Paverd et al., 2014) to characterize threats. Zografopoulos et al. (2021a) propose a TM method that can be categorised as attack(er)-centric. The proposed threat model combines the adversary model (knowledge, access, specificity, and resources) and the attack model (frequency, reproducibility, discoverability, functional level, asset techniques, and premise). The attacker-centric approach was also referenced as security-centric by Morana and UcedaVelez (2015).

System-centric As threat modeling methodologies were first introduced to secure software through the development life cycle, this approach is also known as *software-centric* approach. However, we find the term *system-centric* approach to be more relevant to ICSs threat modeling. The idea of using a system-centric approach is to map out the whole system in diagrams and study the threats through each of the system components. This approach requires a better understanding of the functions of the entire system, which might be a challenging task for complex systems. However, when system designers are involved in the threat modeling process, the system-centric approach seems to be easy to adapt, as it does not require good knowledge of the business assets or attacker's profiles as the two other approaches (Adam Shostack, 2013).

It is interesting to mention that STRIDE is categorized as attacker-centric (Morana and UcedaVelez, 2015), and system-centric (Hajrić et al., 2020; Withers, 2016; Adam Shostack, 2013) TM approaches. However, we tend to categorize it as a system-centric threat modeling approach, as it maps the whole system and analyzes its threats. Different TM approaches might be combined to provide a "complete" list of threats (Jamil et al., 2022). For example, the system-centric approach was combined with the attacker-centric by Withers (2016).

5.1.3. Threat model focus

This SLR categorizes the focus of an ICSs threat model as privacy focused, cyber security focused, and safety and security focused.

Privacy focused TM A study is categorized as privacy focused if its primary objective is to explore privacy-related threats.

Cyber security focused TM Widely used threat modeling methods (e.g., STRIDE) analyze cyber-security aspects, similar to the CIA triad terminology (Confidentiality, Integrity, and Availability), which is derived from the classical view of information-centric loss in IT systems.

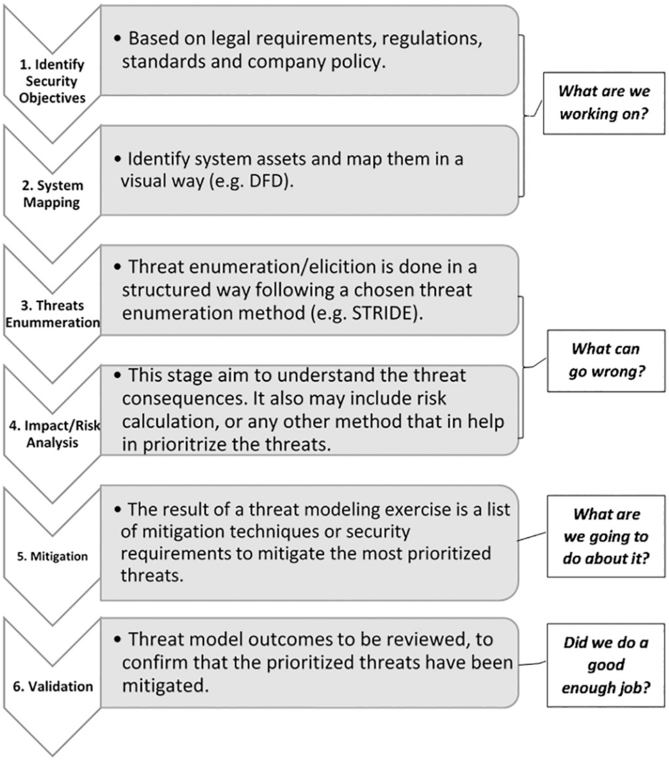
Safety and security focused TM An ICS carries the properties of a CPS, which requires a TM method that combines both safety and security aspects into its analysis. If a study has the sole focus on safety, we excluded it from our review as it is beyond the scope of this SLR.

5.1.4. Threat model validation method

Validation of a proposed threat modeling method might be done through a formal method or using an empirical validation method. Empirical validation is done based on qualitative or/and quantitative research.

5.1.5. Threat model application area

This SLR identifies the TM application areas for ICSs as the energy sector (i.e., power, oil, and gas), water, manufacturing process, transportation, and health. During the third iteration of this study, we noticed new application areas of ICSs TM, such as agriculture and



- Based on legal requirements, regulations, standards and company policy.
- Identify system assets and map them in a visual way (e.g. DFD).
- Threat enumeration/elicition is done in a structured way following a chosen threat enumeration method (e.g. STRIDE).
- This stage aim to understand the threat consequences. It also may include risk calculation, or any other method that in help in prioritrize the threats.
- The result of a threat modeling exercise is a list of mitigation techniques or security requirements to mitigate the most prioritized threats.
- Threat model outcomes to be reviewed, to confirm that the prioritized threats have been mitigated.

Fig. 5. Identified threat modeling stages.

firefighting. The term ICS is used as an application area in case the categorized paper does not indicate a specific application area.

5.2. Threat modeling stages

The literature shows that there is no unique and systematic threat modeling process to follow, as the process steps vary from one model to another. For example, Microsoft first defined threat modeling in a four-step process: *diagramming*, *threat enumeration*, *mitigation*, and *verification* (Shostack, 2008). Then they updated the threat model phases to 5 steps, after adding the *definition of security requirements*, as the first phase.[24] On the other hand, Steven (2010) proposes eight major steps for threat modeling and each step includes sub-steps: (1) diagram the system, (2) identify assets, (3) identify threats, (4) marry threats and software, (4) enumerate potential “doomsday” impact scenarios, (5) document specific misuse or abuses, (7) enumerate attack vectors, (8) output items (from 6.1 and 7.3) to validate via later application assessment activities.

Based on the literature, as well as the discussed definitions in Section 2.1, we identified six stages of a threat modeling process, as shown in Fig. 5. The first stage is *Identify Security Objectives* based on legal requirements, regulations, standards, and organizational policy. *Identify Security Goals* might also be included in this stage. Examples of identified security goals might be data confidentiality, integrity, and availability, which are aspects related to IT systems. However, as our analysis scope is ICSs, we may also identify the OT-related goals at this stage (e.g., safety, reliability, and availability). The aim of setting security objectives/goals at this stage is to facilitate the security requirement elicitation later and provide a justification for requirements selection (Yu and Mylopoulos, 1998).

In the second stage, *System Mapping*, system components are identified, and data flows are mapped. The first two stages aim to answer the question, “what are we working on?”. The question “what can go wrong” is answered using the third stage, *Threat Enumeration*, where threats are identified. In some models, threat enumeration is followed by an impact assessment and/or risk calculation, to help in threat prioritization. We find that this *Impact/Risk analysis* stage is a fundamental stage for ICSs, as it helps to understand the consequences the threats and ensure the prioritization of critical threats mitigation. For this reason, we consider that the 4th stage also answers the question “What can go wrong?”.

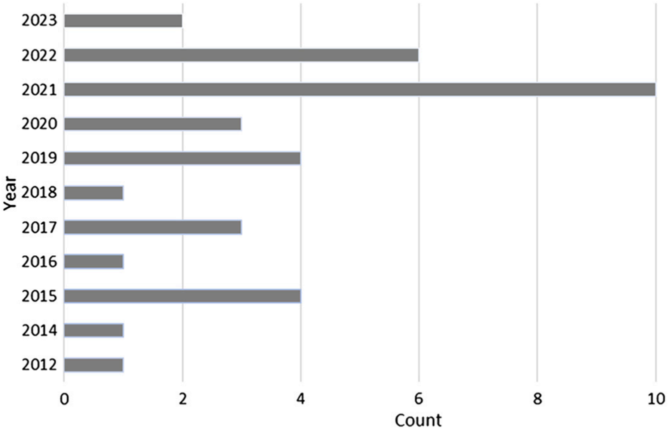


Fig. 6. The number of selected ICSs threat modeling papers per year of publication.

The fifth stage, *Mitigation*, should propose mitigation techniques for the identified threats, as this answers the question “What are we going to do about it?”. In case the threat modeling process was run in the design phase, then the outcome of this stage might be a list of security requirements or security controls.

The last stage, Validation, aims to answer the question “Did we do a good enough job?”. This stage should include validation for the threat modeling analysis, to make sure that all the addressed threats were effectively mitigated. This stage could be performed through peer review, TM tool, or any other method the team agrees on. We must mention that the described six stages seem to suit non-formal ICSs methods better, while it might be challenging to categorize non/semi-formal methods and tools using these stages.

6. Results

This section discusses the results of the analysis of the selected papers. The analysis includes general information about the papers, such as publication year and the first author’s institute location. The papers were also analyzed based on the proposed ICSs TM taxonomy, as well as the covered threat modeling phases.

6.1. General information

The analysis shown in this section is based on the selected papers that propose/apply TM for ICSs. Fig. 6 shows the number of published ICSs threat modeling papers per year of publication. The oldest selected paper is dated to 2012, which we relate to the reveal of Stuxnet in 2010, as this incident was a major motive for researchers and practitioners to prioritize ICSs security. We notice a peak in the number of ICSs TM papers during 2021, which we might refer to the growing number of attacks on ICSs, especially power grids, resulting in more research in this field. The COVID-19 pandemic may have led to a higher number of high-quality ICSs TM publications in 2021, as researchers focused on research during this time.

When looking into the location of the first author’s institutions, it was found that 18 papers (50%), out of the selected 36, were written by a first author’s institute located in Europe, 6 located in North America (17%), and 12 located in Asia (33%), as shown in Fig. 7. When

[24] https://www.microsoft.com/en-us/securityengineering/sdl/threatmodeling.

Table 1
Details of papers cited more than 100 times.

Reference.	Publication Venue	Citations
Friedberg et al. (2017)	Journal of Information Security and Applications	243
Khan et al. (2017)	2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)	198
Liu et al. (2015)	IEEE Transactions on Smart Grid	168
Zografopoulos et al. (2021a)	IEEE Access	134

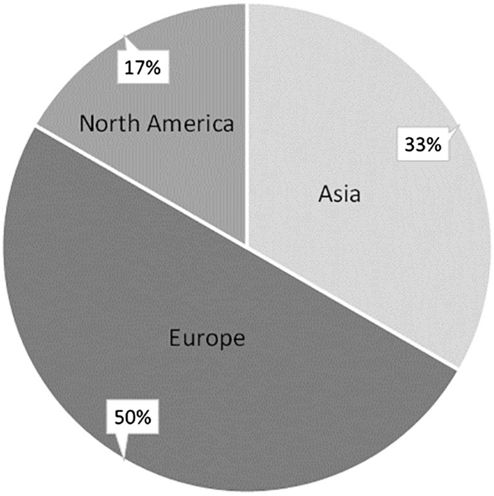


Fig. 7. The location of the first authors' institutes of the selected ICSs TM papers.

looking into the number of authors for each paper, we found that one paper is written by a single author, Fernandez (2016), while the rest of the papers are written by two or more authors. Some papers show the contribution between different institutes located in the same continent (e.g., Friedberg et al. (2017), Radoglou-Grammatikis et al. (2019)), or different continents (e.g., Ahn et al. (2021), Suleiman et al. (2015)).

We analyzed the venues where the selected papers were published. Out of the 36 selected papers, 18 papers are conference papers, 12 of which are published by IEEE conferences. The selected papers include 18 journal papers, where 9 papers are published in Elsevier journals (Journal of Systems and Software, Information Systems, Journal of Information Security and Applications, International Journal of Critical Infrastructure Protection, Internet of Things, Computers in Industry, Journal of Systems Architecture, and Computers & Security), and 4 papers are published in IEEE journals (IEEE Transactions on Smart Grid, IEEE Transactions on Dependable and Secure Computing, and IEEE Access).

As per Google Scholar on August 2023, four selected papers (three Journal papers and a conference paper) has been cited more than 100 times, as shown in Table 1. The most cited paper, Friedberg et al. (2017), proposes a TM method that integrates safety analysis (STPA) and security analysis (STPA-sec) into one framework, STPA-SafeSec, and includes the analysis of control loops in the proposed method. The STPA-SafeSec paper was cited 243 times. The second most cited paper, Khan et al. (2017), applies STRIDE on a Synchronous islanding testbed, which is related to the power industry applications, and the paper was cited 198 times. Liu et al. (2015) main focus is not threat modeling, while the study used threat modeling as a stage toward the creation of an intrusion detection system for Advanced Metering Infrastructure (AMI) and the paper was cited 168 times. Zografopoulos et al. (2021a) apply an attack-centric threat model to an energy system and was cited 134 times.

6.2. Analysis based on the ICSs TM taxonomy

Table 2 summarizes the analysis of the selected ICSs threat modeling papers based on the proposed TM taxonomy, discussed in Section 5. The analysis shows that the proposed TM methods mostly involve a system-centric approach in their threat analysis (23 papers), which focuses on the whole system's threats. We also noticed that non-formal methods tend to use a system-centric approach, as 15 out of 19 non-formal papers adopt a system-centric approach. On the other side, four out of five modeling language papers were found to use an attack-centric approach. The attack-centric approach is combined with the Asset-centric approach by Stellios et al. (2021); Lee et al. (2021); Jbair et al. (2022) and combined with the system-centric approach by Suleiman and Svetinovic (2013); Ahn et al. (2021). We also noticed that the asset-centric studies Stellios et al. (2021); Rimsha and Rimsha (2019); Zografopoulos et al. (2021b) perform risk calculation for critical assets. The third search iteration, done in June 2023, included the study by Lee et al. (2021), which uses a data-centric approach combined with asset and attack-centric approaches. The study used STRIDE for threat enumeration, while the enumerated threats were only threats related to data flow.

Of the 36 selected papers, 25 focus solely on cyber security, while a single paper, Wuyts et al. (2014), focuses solely on privacy. Two papers focus on both privacy and cyber security (Chen and Huang, 2019; Zahid et al., 2023). Six papers focus on safety and security (without considering privacy) which supports the comprehensiveness of the TM results, as ICSs safety is fundamental, and it could be impacted by cyber-security threats. For instance, STPA-SafeSec (Friedberg et al., 2017), investigates the interrelationship between safety and security vulnerabilities. The study combines safety analysis methods, STPA, and STPA-sec into a single approach to balance both safety and security constraints. We have considered studies that identify the threat consequences, such as Khan et al. (2017), and Khalil et al. (2023), to focus on both ICSs security and safety, as they consider the physical threat consequences that may impact the system safety and cause human injuries, or system damage.

Out of the 36 selected papers, only Two papers consider threats related to safety, security, and privacy, Suleiman et al. (2015) and Amro et al. (2023). The paper by Suleiman et al. (2015) elicits threats based on SQUARE and SREP methods. However, it does not clarify the procedures of the mentioned methods. The research uses a Smart Grid (SG) system as a case study and elicited 76 related threats that are categorized into 6 categories: threats to network availability, threats to data integrity, threats to data confidentiality and information privacy, threats to SG key operation, threats to inter-organizational collaborations and relationships, and threats to the resilience and safety of SG infrastructure. The study by Amro et al. (2023) uses a bottom-up risk assessment approach called *Failure Modes Effects and Criticality Analysis* (FMECA). The authors identify the failure modes of autonomous passenger ships, based on the MITRE ATT&CK framework. The paper provides extensive methodology details, which might help the reader to adapt the proposed method in TM of other application areas of ICSs.

The analysis of selected papers shows that 35 used methods are validated using empirical validation. Whereas 34 papers use qualitative methods for validation, and a single paper uses a quantitative research method. The qualitative validation methods are mostly done through use cases and case studies, except the study of Suleiman and Svetinovic (2013), who did further validation for the used ICSs TM method based

Table 2
Categorization of the selected ICSs TM papers based on the proposed ICSs TM taxonomy.

Reference	Method Type					Approach			Focus			Validation		Application Area
	Tool	Mod. Lang.	Form.	Non-Form.	Semi-Form.	Asset-Centric	Attack-Centric	System-Centric	Priv.	Safety + Sec.	Cyber Sec.	Form.	Emp.	
Wuyts et al. (2014)				Y				Y	Y				Y	Energy
Suleiman et al. (2015)				Y				Y	Y	Y			Y	Energy
Suleiman and Svetinovic (2013)				Y			Y	Y			Y		Y	Energy
Chen and Huang (2019)					Y			Y	Y		Y		Y	Energy
Ramis Ferrer et al. (2017)				Y				Y			Y		Y	Manufact.
Ahn et al. (2021)					Y		Y	Y			Y		Y	Energy
Friedberg et al. (2017)				Y				Y		Y			Y	Energy
Khan et al. (2017)				Y				Y		Y			Y	Energy
Fernandez (2016)		Y					Y			Y			Y	Transport
Haider et al. (2019)				Y				Y			Y		Y	Energy
Radoglou-Grammatikis et al. (2019)				Y				Y			Y		Y	ICS
Zografopoulos et al. (2021a)					Y		Y				Y		Y	Energy
Stellios et al. (2021)					Y	Y	Y				Y		Y	Health
Girdhar et al. (2021)				Y				Y			Y		Y	Energy
Fla et al. (2021)	Y							Y			Y		Y	Energy
Li et al. (2021)				Y				Y			Y		Y	ICS
Sun et al. (2020)		Y					Y				Y		Y	Water
Rimsha and Rimsha (2019)	Y					Y					Y			Energy
Schlegel et al. (2015)	Y							Y			Y		Y	Energy
Martins et al. (2015)	Y							Y			Y		Y	Transport
Vernotte et al. (2018)	Y						Y				Y		Y	Energy
Hacks et al. (2020)		Y					Y				Y		Y	Energy
Zografopoulos et al. (2021b)				Y		Y					Y		Y	Energy
Liu et al. (2015)				Y				Y			Y		Y	Energy
Foldvari et al. (2022)	Y							Y		Y			Y	Water
Kumar et al. (2022)				Y			Y				Y		Y	Energy
Amro et al. (2023)					Y		Y		Y	Y			Y	Transport
Lee et al. (2021)				Y		Y	Y				Y		Y	ICS
Al Asif et al. (2021)				Y				Y			Y		Y	Agriculture
Kim et al. (2022)				Y				Y			Y		Y	Energy
Shibly and De Soto (2020)				Y				Y			Y		Y	Manufact.
Zahid et al. (2023)				Y			Y		Y		Y		Y	Firefight.
Khalil et al. (2023)				Y				Y		Y			Y	Energy
Valenza et al. (2022)	Y		Y					Y			Y		Y	Energy
Jbair et al. (2022)	Y	Y				Y	Y			Y			Y	Manufact.
Rouland et al. (2021)	Y	Y	Y					Y			Y		Y	Energy

on a list of identified evaluation criteria. Using a case study, Suleiman and Svetinovic (2013) evaluate the Security Quality Requirements Engineering (SQUARE) (Mead and Stehney, 2005) based on qualitative research that analyzes the method's: flexibility, sampling, analyst permissibility, interpretiveness, data sufficiency, coherence, repeatability, completeness, usability, and credibility. The evaluation concludes that SQUARE does not guarantee the completeness of security goals identification, also it is not clear when to stop each of the SQUARE steps.

The study presented by Wuyts et al. (2014) uses a quantitative empirical method to validate the LINDDUN TM method based on four aspects: correctness, completeness, productivity, and ease of use. The experiment setup of Wuyts et al. (2014) was also previously used by Scandariato et al. (2015) to validate STRIDE as a software TM method. The LINDDUN experiment included 3 studies for validation. Two descriptive studies worked on the requirement analysis and design phases, where students discover privacy threats and design weaknesses. The third study involved privacy experts, to compare their findings to the LINDDUN method findings. The research outcomes showed that LINDDUN was considered easy to learn and the students provided reasonable levels of correctness ratios. However, the method completeness could be improved and the threat tree catalogs could explain some threats better.

As for the application areas of the proposed ICSs TM methods, we have marked papers with an "ICS" sector when they do not focus on a specific application area, instead, they focus on ICSs in general. The energy sector is the most common application area for ICSs TM methods, with 22 studies focusing on this sector. This is likely due to the critical nature of the energy sector, as many sectors depend on it, and the potential for significant damage that could be caused by a cyberattack. The Ukrainian power grid incidents were a stark reminder of the need for better security of energy systems against cyber attacks. These incidents caused widespread power outages and highlighted how cyber threats could cause large damage in such a critical sector.

This review also includes transport-related papers. Fernandez (2016) focus on a port loading system, and Martins et al. (2015) focus on a real-world wireless railway temperature monitoring system. The water sector is covered by the studies Sun et al. (2020) and Foldvari et al. (2022). Ramis Ferrer et al. (2017) is applied to a manufacturing process, and Jbair et al. (2022) is also applied to smart manufacturing applications. Al Asif et al. (2021) focus on the agriculture sector, while Zahid et al. (2023) focus on a smart firefighting system. As general application areas of ICS, Radoglou-Grammatikis et al. (2019) use a SCADA testbed in their use case, while Li et al. (2021) use a PLC-related case study.

6.3. ICSs TM methods analysis based on the applied TM stages

The selected papers were analyzed based on the covered TM stages, as shown in Table 3. We noticed only three papers, Kim et al. (2022), Zahid et al. (2023) and Suleiman and Svetinovic (2013) mention the security objectives phase, while all the other studies start the threat modeling process from the system mapping phase. We could relate this behavior to the fact that most of the papers were not applying the proposed method to a real-world system, so they were not required to identify such objectives. Also, the use of methods similar to STRIDE and LINDDUN might be considered as a substitute for this first stage, as these methods already analyze the system against aspects that could be considered as security and privacy objectives. However, we have to stress that in a real-life scenario, identifying the security objectives is

Table 3
Covered TM phases by the selected papers.

Reference	Identify Security Objectives	System Mapping	Threat Enumeration	Impact/Risk Analysis	Mitigation	Validation
Wuyts et al. (2014)		Y	Y	Out of Scope	Out of Scope	
Suleiman et al. (2015)		Y	Y		Out of Scope	
Suleiman and Svetinovic (2013)	Y	Y	Y	Y	Y	Y
Chen and Huang (2019)		Y	Y	Y		
Ramis Ferrer et al. (2017)		Y	Y	Y	Y	
Ahn et al. (2021)		Y	Y	Y	Out of Scope	
Friedberg et al. (2017)		Y	Y	Y		
Khan et al. (2017)		Y	Y	Y	Out of Scope	
Fernandez (2016)		Y	Y		Y	
Haider et al. (2019)		Y	Y	Y	Y	
Radoglou-Grammatikis et al. (2019)		Y	Y	Y		
Zografopoulos et al. (2021a)		Y	Y	Y		
Stellios et al. (2021)		Y	Y	Y	Y	
Girdhar et al. (2021)		Y	Y	Y	Y	
Fla et al. (2021)		Y	Y			
Li et al. (2021)		Y	Y			
Sun et al. (2020)		Y	Y	Y		
Rimsha and Rimsha (2019)		Y	Y	Y		
Schlegel et al. (2015)		Y	Y	Y	Y	Y
Martins et al. (2015)		Y	Y	Y	Y	
Vernotte et al. (2018)		Y	Y		Y	
Hacks et al. (2020)		Y	Y	Y	Y	
Zografopoulos et al. (2021b)		Y	Y	Y		
Liu et al. (2015)		Y	Y		Y	
Foldvari et al. (2022)		Y	Y	Y		
Kumar et al. (2022)		Y	Y			
Amro et al. (2023)		Y	Y	Y	Y	
Lee et al. (2021)		Y	Y		Y	
Al Asif et al. (2021)		Y	Y		Y	
Kim et al. (2022)	Y	Y	Y	Y	Y	
Shibly and De Soto (2020)		Y	Y	Y	Y	
Zahid et al. (2023)	Y	Y	Y		Y	
Khalil et al. (2023)		Y	Y	Y	Y	Y
Valenza et al. (2022)		Y	Y	Y		
Jbair et al. (2022)		Y	Y	Y	Y	
Rouland et al. (2021)		Y	Y		Y	

a fundamental case, as ICSs have to meet the organizational policy, as well as industry standards and regulations.

It was also found that all the selected methods go through the process of system mapping and threat enumeration. As they are the fundamental stages of threat modeling. Impact and risk assessment are important activities for prioritizing threats and choosing suitable security requirements and controls to mitigate critical threats. The analysis shows that most of the papers consider the impact/risk analysis phase. When a stage is mentioned by the paper, but it is set out of the scope of the paper, then we mention it as *out of scope* in Table 3.

We noticed that most of the studies calculate the risk using impact and likelihood. However, some papers have different definitions of how to perform the impact or risk assessment. For example, Girdhar et al. (2021) mention that STRIDE is a *"categorical risk assessment model"* and that the risk analysis is done through the use of STRIDE. The paper does not calculate the risk using impact and likelihood, as it is commonly done, instead, during the threat elicitation phase, it mentions what we may consider as the "impact" of each threat. Also, we marked that Suleiman and Svetinovic (2013) perform impact/risk analysis, while it is interesting to mention that this method performs the prioritization after eliciting the security requirements instead of prioritizing the elicited threats. Khalil et al. (2023) map losses to threat consequences and assign a qualitative priority of *critical, high, medium, and low* for each loss. Then They calculate threat priority without using weighted numbers or considering likelihood, as it depends on unavailable information, such as software and hardware vendors and versions.

Several papers mention the security mitigation phase. Suleiman and Svetinovic (2013) provide the description of 15 security requirements for AMI (e.g., The system shall prevent unauthorized access to its storage data). Ramis Ferrer et al. (2017) and Khalil et al. (2023) fulfill the security requirements according to ISA/IEC 62443 standard. On the other side, the tool proposed by Martins et al. (2015) uses NIST SP 800-82 Revision 2 (Stouffer et al., 2015b) security controls, and Zahid et al. (2023) use NIST security and privacy controls proposed by Force (2017).

Haider et al. (2019) proposed mitigation techniques for wireless attacks (e.g., Intrusion Detection System (IDS) for man-in-the-middle attacks). The TM tool proposed by Schlegel et al. (2015) is able to propose security controls for the elicited threats (e.g., propose antivirus as mitigation for malware threat). Amro et al. (2023) propose mitigation methods, similar to audit and authorization enforcement, for some MITRE ATT&CK techniques. Al Asif et al. (2021) use defense mechanisms for STRIDE categories, (e.g., Multifactor authentication for device spoofing). On the other side, Fernandez (2016) set security constraints, which we consider a positive action toward threat mitigation. Girdhar et al. (2021) and Liu et al. (2015), both papers include detection solutions, which could be considered as mitigation techniques for the elicited threats. Hacks et al. (2020) mention that mitigation techniques could be added to the modeled system on SecuriCAD, similar to antivirus and firewalls.

Suleiman and Svetinovic (2013) Validates the outcomes of the threat modeling process by including a requirements inspection phase to the SQUARE method, and it mentions that SQUARE provides a checklist-based method to perform such activity, while the details were not discussed in the research. Also, Schlegel et al. (2015) include the TM outcome validation by performing threat modeling manually and using the tool proposed by the authors, then comparing the findings in order to validate the tool outcomes. Khalil et al. (2023) used published tech-

Table 4
ICSs TM methods and techniques.

Reference	TM Method Name	Threat Enumeration Technique	System Mapping Method	Other Used Techniques
Wuyts et al. (2014)	LINDDUN	LINDDUN	DFD	Attack tree
Suleiman et al. (2015)	SG SSTM	SQUARE + SREP		
Suleiman and Svetinovic (2013)	SQUARE			Security template, use case, attack tree
Chen and Huang (2019)	I-SERM	STRIDE+p	DFD	Product Flow Diagram (PFD), Use Case Diagram (UCD), Threat Breakdown Structure + ref. scores (TBS+r), Delphi method, threat tree
Ramis Ferrer et al. (2017)		STRIDE		Qualitative risk model
Ahn et al. (2021)		STRIDE	DFD	CVE, CVSS, attack tree, CAPEC, Matlab
Friedberg et al. (2017)	STPA-SafeSec			
Khan et al. (2017)		STRIDE	DFD	
Fernandez (2016)	Misuse Pattern			Sequence diagram, patterns, activity diagram, class diagram
Haider et al. (2019)		STRIDE	DFD	DREAD
Radoglou-Grammatikis et al. (2019)	CPN-based Threat Model		CPN	Commen Weakness Enumeration (CWE), AlienVault risk assessment model
Zografopoulos et al. (2021a)				Matlab
Stellios et al. (2021)				CVSS, attack path
Girdhar et al. (2021)		STRIDE	DFD	Weighted attack defence tree
Fla et al. (2021)	Microsoft TM Tool	STRIDE	DFD	
Li et al. (2021)		STRIDE	DFD	
Sun et al. (2020)	Pimca Domain Specific Language (DSL)			
Rimsha and Rimsha (2019)	R.I.M.S.H.A. tool			
Schlegel et al. (2015)	System TM tool			
Martins et al. (2015)	An ICSs TM tool			
Vernotte et al. (2018)	SecuriCAD tool			Attack graph
Hacks et al. (2020)	PowerLang DSL			
Zografopoulos et al. (2021b)	OCTAVE Allegro			
Liu et al. (2015)			CPN	
Foldvari et al. (2022)	Impact Analysis Tool			
Kumar et al. (2022)				Attack tree
Amro et al. (2023)	FMECA	ATT&CK		
Lee et al. (2021)	PASTA	STRIDE	DFD	Attack tree
Al Asif et al. (2021)		STRIDE	DFD	
Kim et al. (2022)		STRIDE	DFD	DREAD
Shibly and De Soto (2020)	QTMM	STRIDE	DFD	Attack tree, CVSS
Zahid et al. (2023)		ATT&CK		
Khalil et al. (2023)		STRIDE	DFD	
Valenza et al. (2022)	TAMELESS tool			Attack graph
Jbair et al. (2022)	VTM&CG tool	ATT&CK		Attack tree
Rouland et al. (2021)		STRIDE		Alloy (Formal modeling language)

nical documents to cross-check the outcomes of the threat modeling process.

6.4. ICSs TM methods and techniques

As this research aims to identify the used ICSs TM methods, Table 4 summarizes the names and techniques of the used methods. The table shows that the system mapping method, DFD, is mostly used within STRIDE. Thirteen papers, out the 36 selected paper, use DFD to map the system's information flow as a step in the applications of STRIDE and LINDDUN. On the other side, Ramis Ferrer et al. (2017) propose the use of STRIDE without creating a DFD, instead, the paper uses a component table that provides for each component a description, trust level, and entry points similar to MODBUS, IP, RS232, AND HTTP PORT.

While DFD is broadly used in TM, Colored Petri-Net (CPN) might be a substitute to consider for ICSs. For this reason Table 4 highlights only the DFD and CPN as system mapping methods. The table does not mention the traditional ways of system mapping, similar to the system architecture diagrams. Architecture diagrams are mostly included in TM studies, even when DFD or CPN is applied. These diagrams are fundamental to understanding the network topology and they could be used as a starting point for the information flow mapping. In some threat modeling methods, the architecture diagrams are the only system representation, and no information flow mapping is used.

CPN is an extension of Petri-Net, which can abstractly describe physical phenomena. Petri-Net interprets the relations between the different elements and identifies the information flows. The elements of a Petri-Net are Place, Transition, Connection, and Token. CPN differentiates between the different types of information by providing a different token for each type of information flow (e.g., power flows, data flows, and command flows).

Liu et al. (2015) use CPN to map the information flows of a smart meter, and classify them into two types of information flow: data (users' consumption information and electricity prices from utility companies) and command flows (different commands received from the utility companies and other devices). The provided threat model identifies two types of attacks (attacks on data and attacks on commands). For each of the identified types of attack, they mention targetted information flows, occurring location, and possible attack techniques. The attack techniques were also categorized into two categories (physical attack and cyber attack). Radoglou-Grammatikis et al. (2019) also use CPN and classify the attacks into physical and cyber attacks.

As clarified in Section 2.4, STRIDE is a threat enumeration technique that was initially designed for software TM. For this reason Table 4 does not identify STRIDE as a TM method. Some papers propose combining STRIDE with other techniques, to better represent ICSs. For example, Ahn et al. (2021) complement the TM with attack modeling, combine STRIDE with CVE, CVSS, and attack tree, and use CAPEC to calculate a numerical attack success rate of attacks using MATLAB. The paper briefly applies the proposed methodology to a power transformer in a digital substation. While such a methodology seems to seek completeness, it leaves the reader with many questions related to the methodology. For example, the research does not provide detailed information about the MATLAB simulation, and the process of choosing CVE, CVSS, and CPEC.

Khan et al. (2017) mention that there is no standard methodology defined in the literature for applying STRIDE, and propose a methodology for its application on CPSs. Identifying a list of possible threat consequences is proposed by Khan et al. (2017) as a first step in threat analysis. Then, the study uses STRIDE-per-element followed by STRIDE-per-interaction, to complement the list of elicited threats. Kim et al. (2022) is another example of studies that combine STRIDE-per-element and STRIDE-per-interaction a in single methodology.

Li et al. (2021) is another example of studies that use STRIDE and apply it to the workflow of operators and control engineers who setup the webserver of TIA portal.[25] Li et al. (2021) use STRIDE per element, without explicitly mentioning the term, however, it did not use the famous STRIDE threat categories for mapping the DFD elements to threats. The paper also added a new STRIDE analysis element named "task scenario", which includes authentication, network and access configuration, information gathering by the operator, client authentication, and operator remote work tasks. Li et al. (2021) map the task scenarios to "usability factors" that are selected from previous literature based on CPSs needs. Examples of chosen usability factors are efficiency, safety, availability, and accuracy. The paper then categorizes the selected usability factors into 3 categories, environment-related, application-related, and user-related.

Another example of STRIDE application is presented by Haider et al. (2019), where the authors combine STRIDE with DREAD for the analysis and prioritization of wireless attacks on AMI. The study identifies 5 related threats (DOS, DDOS, FDI, De-pseudonymization, and MITM), and applies STRIDE to those threats. STRIDE is also used by Girdhar et al. (2021), in combination with a weighted attack defense tree that describes the attack objectives of the adversary.

Lee et al. (2021), added by the third iteration of this SLR, combine STRIDE for threat enumeration within the *Process of Attack Simulation and Threat Analysis* (PASTA) TM. The research justifies its choice for PASTA, as it can incorporate attacker-centric and data-centric to its asset-centric approach. PASTA applies seven threat modeling stages: define objectives, define technical scope, system/application decomposition, threat analysis, vulnerabilities and weakness analysis, attack modeling, and risk and impact analysis. The study uses an attack tree and STRIDE to analyze the data flows.

Shibly and De Soto (2020) propose a framework for Quantitative Threat Modeling Methods (QTMM) that is composed of 6 steps: define use case/problem statement, define DFD, map sTRIDE elements into DFD, identify threats, perform Quantitative risk assessment and plan risk mitigation strategies. The research combines sTRIDE with the attack tree and uses the CVSS score and risk propagation technique to calculate the risk. The paper proposes also adding risk mitigation into the attack tree.

As the STRIDE application can be automated using Microsoft Threat Modeling Tool, Fla et al. (2021) propose a smart grid template for the tool. The study created a list of threats based on the literature, existing templates, and cyber-attacks on OT. The tool template is configured to use stride-per-interaction. While the research also criticized STRIDE for not including a method for threat prioritization. STRIDE was also used with the context of formal method in Rouland et al. (2021), where it was used to elicit threats using technology-independent specification language.

The study Chen and Huang (2019) uses a threat enumeration method based on STRIDE with privacy aspect enhancements (i.e., named STRIDE+p), as a part of a new TM method named I-SERM. As per Chen and Huang (2019), STRIDE+p combines the features of STRIDE and LINDDUN, however, the methodology is not detailed enough to ease the use of the proposed method. The I-SERM method considers risk management to prioritize threats and proposes a process of seven steps to apply the model.

The use of STRIDE was justified by Girdhar et al. (2021), as STRIDE is considered the most mature threat modeling approach. The paper mentions that STRIDE evaluates the detailed system design by building DFD and identifies cyber threats against each system entity. Moreover, STRIDE is comprehensive and analyzes security properties against each system component, while ensuring system security at the component level. Also, as per Girdhar et al. (2021), STRIDE is widely adopted as a cyber security framework to secure critical infrastructure from cyber attacks in CPSs.

It is interesting to mention that some papers (e.g., Khan et al. (2017)), map analog physical components on the DFD as an external entity, while there might be a need to differentiate between an external entity and a physical component that does not have a digital connection to the system. Also, differentiating between data flows, command flows, and power flows might be important information to map on the diagrams of an ICS. As some flows might be more critical than others (e.g., control flows), which was acheived by CPN in the studies of Radoglou-Grammatikis et al. (2019) and Liu et al. (2015). On the other hand, Khalil et al. (2023) used a colored DFD, to differentiate between three types of processes: electrical power, measurement, and software-related. The paper also identifies physical processes, and analog connections in the DFD, to make it more comprehensive for the involved stakeholders.

We have also noticed that the third search iteration included research that used MITRE ATT&Ck for threat enumeration (Zahid et al. (2023), Amro et al. (2023) and Jbair et al. (2022)). Using a specific ICSs attacks database similar to MITRE ATT&CK, which is constantly updated, could enrich the threat elicitation process. The VueOne Threat Modeller and Code Generator Tool (VTM&CG) proposed by Jbair et al. (2022) also incorporate the use of attack trees.

Some papers propose different TM methods other than STRIDE. For instance, Suleiman and Svetinovic (2013) evaluate the Security Quality Requirements Engineering (SQUARE) method, which consists of nine steps: agree on definitions, identify security goals, develop artifacts to support security, requirements definitions, perform risk assessment, select requirements elicitation technique, elicit the security requirements, categorize the security requirements, prioritize the security requirements, and inspect the security requirements. While most of the TM methods propose threat prioritization, SQUARE prioritizes the security requirements instead of the threats, which require extra effort. Suleiman et al. (2015) introduce a method named Smart Grid Systems Security Threat Model (SG SSTM), which proposes a reference architecture of a smart grid and categorizes its security threats into 6 categories: network availability, data integrity, data confidentiality and information privacy, smart grid key operation, inter-organizational collaborations, and relationships, and the resilience and safety of SG infrastructure. While the paper mentions that the threats were elicited using 2 threat analysis methods: SQUARE (Suleiman and Svetinovic, 2013) and Security Requirements Engineering Process (SREP) (Wang et al., 2009) methods, it just share the outcomes of the threat elicitation process without details on the elicitation process.

TM methods mostly use a bottom-up approach, where threats are used to derive the security requirements. However, methods with a safety and security focus, similar to STPA-SAFESec, prefer to use the top-down approach. A top-down approach identifies the unacceptable system losses, and the analysis moves from general to specific (Young and Leveson, 2014). For instance, STPA-SAFESec starts with defining the control layer where the system losses, hazards, and control loops

[25] Siemens TIA portal is used by engineers and operators to configure PLCs during and after system deployment.

are identified. The method also includes refining safety and security constraints and defining hazard scenarios that help in identifying mitigation strategies.

Vernotte et al. (2018) provide a load-balancing centered smart grid reference architecture model, and perform a TM using SecuriCAD. The study defines the tool as *"a modeling framework and calculation engine that estimates the cyber security of systems-of-systems-level architectures"*. Using SecuriCAD, the paper models the system architecture, including data flows, and identifies its security characteristics (e.g., host firewall enabled). The tool then creates attack graphs for the modeled system and calculates the shortest paths and Time-To-Compromise for attacks.

Rimsha and Rimsha (2019) briefly describes the «Risk Identification and Management Security Host-based Appliance» (R.I.M.S.H.A.) tool. The tool performs TM and risk assessment based on a quantitative risk assessment method. Schlegel et al. (2015) propose a TM tool based on a data model of component, threat, impact, and security control. Martins et al. (2015) propose a tool that can identify potential threats of a CPS based on MetaGME modeling language. Foldvari et al. (2022) propose an impact analysis tool that uses the error propagation analysis while considering both CPSs safety and security. The tool explores the impact of a cyber attack on CPSs and highlights critical control paths, which help in choosing/validating countermeasures.

Some papers propose domain-specific modeling languages. PowerLang (Hacks et al., 2020) is a MAL-based domain-specific language for modeling IT and OT infrastructures in the power domain. Hacks et al. (2020) reuse two existing MAL-based DSL (corelang, sclLang), and propose (icsLang) to fill the gaps in the reused languages. Pimca, an attack-centric domain-specific threat modeling language, is proposed by Sun et al. (2020). The Pimca DSL was originally developed by the Directorate General of Armaments (DGA), French Ministry of Armed Forces.

6.5. Other analyzed aspects

As a part of this qualitative analysis, the recommended development life cycle stage for TM was analyzed in each of the selected papers. Wuyts et al. (2014) mention that "*Threat modeling can be applied at different levels of abstraction, depending on the granularity of assets considered*". As per Wuyts et al. (2014), the assets are more abstract at the requirement phase, while they become more tangible during the architectural design. While some papers (e.g., Ramis Ferrer et al. (2017); Ahn et al. (2021); Martins et al. (2015); Girdhar et al. (2021)) explicitly mention the design phase for TM application, other papers (e.g., Suleiman et al. (2015); Suleiman and Svetinovic (2013); Friedberg et al. (2017); Khan et al. (2017)) just mention that TM is applied at the early stages of the system development life cycle. The study presented by Schlegel et al. (2015) highlights the possibility of running TM at the operation phase for threat mitigation, while it mentions that running TM at the design and implementation phases is more cost-effective.

Another aspect that this SLR looked into is the required expert knowledge for each of the selected methods. Some methods explicitly mention the need for expert knowledge to run them. For example, Suleiman and Svetinovic (2013) mention that the application of the SQUARE method requires some level of domain expertise in order to identify potential threats, and propose that such action be done through a workshop. Experts are also required to assess the threats' impacts and their criticality. For example, the Delphi method, which is based on expert knowledge, is proposed for threat prioritization by Chen and Huang (2019) based on the impact and likelihood of the threats. Assessing the impacts of the threats by experts is also proposed by Ramis Ferrer et al. (2017) and Zografopoulos et al. (2021b). Khan et al. (2017) identify the threat consequences by domain experts. System losses and hazards are also identified by experts in the case of Friedberg et al. (2017). The proposed tool by Schlegel et al. (2015) targets security professionals, as sufficient security knowledge is a prerequisite. On the other hand, Vernotte et al. (2018) mention that the securiCAD tool has collected quantitative data from different sources, so there is no security expertise requirement from the tool user.

7. Research gaps and future directions

This SLR discussed the different aspects of ICSs threat modeling. The analysis shows that the ICSs TM inherits the characteristics of software TM as a general domain. The ICSs TM is currently more focused on the threat elicitation stage and does not provide clear procedures for other stages, while a clear description for all TM stages is a vital requirement for practicing threat modeling in real settings. Also, the selected threat modeling papers do not discuss the technical and organizational challenges and how to overcome them regarding the application of these methods. These notes go with the finding of Yskout et al. (2020), which also concluded that threat modeling, as an engineering discipline, is at a low level of maturity even in the software security domain where TM originated. The SLR findings are in line with other findings of Yskout et al. (2020), as this SLR identifies that there is a lack of published industrial case studies and experience reports related to ICSs TM.

TM is an interactive activity built on communication between various teams with different perspectives and goals, such as system developers, system administrators, security experts, and domain professionals. Compared to software-based systems, the development and maintenance of CPSs require more diverse expertise, ranging from experts in physical processes to system developers. This communication aspect has not been studied in the selected papers. It is unclear how to effectively construct such communication to ensure that different teams have a common understanding of the discussed threats and their implications. Interdisciplinary studies that address the intersection of technology and human perspectives are highly required in this problem domain. As a more specific suggestion, the use of gamification techniques might be considered as a potential instrument for team communication in TM processes, which could be studied in future research.

In this research, we noticed that "threat modeling" is also used for cases in which a systematic method is not followed. For example, the study of Soltan et al. (2018) has a threat modeling section, while it does not follow a systematic threat enumeration approach. It could be considered as a listing of some possible attacks based on a defined scenario, however, it just focuses on a specific type of attack without enumerating all possible attacks. Therefore, we excluded such studies from this SLR (see Section 4.2.2). We contemplate that a threat modeling method should be applicable to a wider group of systems and cover various attack types rather than being very specific to an attack type or target system.

This SLR identified two types of validation for ICSs TM. First, validating the used ICSs TM method which aims the evaluate aspects similar to the effectiveness and correctness of the method. Such validation could be done through qualitative or quantitative research methods. The second type of validation is validating the outcomes of each ICSs TM run regardless of the used methodology. The TM outcomes validation could be considered as the "quality assurance" for the TM process. It aims to ensure that the critical threats were elicited and suitable security requirements were proposed. This second type of validation might be done through a peer review or it might be automated. The provided analysis shows that there is a clear gap in both validation types, which also goes with the findings of Tuma et al. (2018).

While STRIDE seems to be the most mature threat modeling method, this SLR shows that there is no clear understanding of how to document the threats elicited using STRIDE. While many of the selected studies used STRIDE, they do not use a common language to describe the threat elicited by STRIDE. For example, Chen and Huang (2019) elicit STRIDE threats through a table, where the threat elicitation phase was merged with the risk assessment, and no detailed description is provided for each threat. On the other hand, Ramis Ferrer et al. (2017) elicit the threats with a description based on a chosen CVE related to each STRIDE threat. Also, the level of expertise for identifying threat consequences on ICSs seems to be a limitation toward the automation of ICSs TM, as identifying the criticality of an ICSs threat should be related to the impact the threat might cause on the system. This raises the question of the effectiveness of using threat libraries and knowledge bases similar to MITRE ATTA&CK.

ICSs play a key role in most of the critical infrastructures that require special consideration. The design of such systems should consider ensuring safety and security. If the target system processes information about individuals, data privacy would be another concern as a system objective. However, this study shows a need for creating mature and easy-to-adapt ICSs TM methods that cover all those aspects simultaneously.

Security threat models help derive preventive and detective controls, such as authentication, access control, and network controls. Effective reactive controls, such as incident handling, depend on the forensic readiness of the system, which requires considering forensic artifact extraction and analysis requirements early in development. Experts can handle these requirements using current models (e.g., non-repudiation in STRIDE). However, the current models do not provide a systematic instrument to address forensics readiness requirements. TM techniques can be extended to cover this important aspect of critical infrastructure, as described by Iqbal et al. (2020).

All the selected papers use TM during the design phase of an ICS. Although this finding aligns with the common perception of TM as an early-stage development instrument, we contemplate that the knowledge created in TM can be utilized in the later steps of SDLC. For instance, while conducting penetration testing at the verification stage. As cyber-physical systems have various types of interfaces, protocols, and data flows, it is very difficult to define a scope and proper course of action for penetration testing endeavors. Security experts do not perform this task systematically despite the fact that enough human resources are not available in most cases. TM can act as an instrument for identifying and prioritizing the system components/information flows that would be addressed during the penetration testing within the given resource limitations. Additionally, the utilization of TM enables knowledge transfer between the early and late-stage steps.

Threat modeling methods reviewed in this paper usually perceive the target system as static. Although some of them elaborate on the malicious actors while eliciting the threats, they do not consider the interaction between those actors and defenders. It is assumed that the defenders do not apply more proactive countermeasures such as moving target defence (i.e., changing the attack surface to misguide the attackers). It is a promising research path to incorporate game theory into threat modeling to reconcile the attack and defence actions and consider dynamic defensive actions (e.g., moving target defence) in such modeling formulation (Tan et al., 2022, 2023).

It is obvious that we may expect more ICSs components that benefit from artificial intelligence (AI) as various research studies demonstrate the applicability of AI in addressing the problem domains where cyber-physical systems are prevalent (Raza and Khosravi, 2015; Stetco et al., 2019). A huge body of research proposed adversarial attacks that threaten the AI-based models (Papernot et al., 2016). A very limited number of studies have focused on systematically identifying such attacks using threat models (Bitton et al., 2022). More importantly, providing a holistic threat model that synthesizes data-centric adversarial attacks and conventional cyber attacks and applying it to ICSs constitutes an interesting research direction.

Threat modeling is a domain in which advancement is driven by both academia and industry. Although Microsoft's SDLC practices introduced this notion to the software development community, academia has systematized it for ICSs (Khan et al., 2017; Khalil et al., 2023) and adapted it to privacy threat modeling (Wuyts et al., 2014). More cooperation between both parties is needed to demonstrate the applicability of the methods in various application areas. We admit that the proper validation of the threat modeling results is very hard to achieve as the nature of the problem does not enable to maintain ground truth data. However, still, threat libraries and knowledge bases can be enhanced cooperatively. Academia can put more effort into developing qualitative or quantitative validation methods.

8. Threats to validity

This SLR highlighted multiple research gaps in the ICSs TM domain. However, the results of this study are not free from validity threats that might impact the main findings.

The inclusion-exclusion criteria limited the number of studies included in this SLR. Setting the scope of the study to the ICSs application area excluded valuable CPSs TM methods. For example, de Souza et al. (2020) propose a safety and security-related threat model by extending STPA with STRIDE, but it was excluded from our literature review because it used an E-Voting system as an application area. Some papers were excluded from the list while they were related to ICSs, as they did not meet our criteria for ICSs. For example, Hammad et al. (2017) propose a Vulnerable-Link Adaptive Avoidance (VLAA) algorithm that uses a formal vulnerability metric for communication links in CPSs. The paper was excluded from our list, while its application domain is related to Software Defined Networks (SDN) in power systems, as it is related to communication and network applications and does not use system components related to how we identify an ICS. Also, most of the formal TM-related papers were excluded, as they tend to analyze one or two types of attacks, which did not meet our selection criteria. These facts prevented having a more exhaustive list of TM methods in our SLR.

We also noticed that this SLR list did not include some well-known threat modeling methods (e.g., TRIKE, OCTAVE), and the included OCTAVE-related paper, Zografopoulos et al. (2021b), was found through snowballing. Such missing results might be related to the keywords used in the search criteria, as these methods are mostly categorized as *risk assessment* methods, and sometimes they are also called *security assessment* methods. It was also noticed that in safety-related papers, the term *security analysis* is used instead of threat modeling (e.g., Friedberg et al. (2017)). Also, we found that TM is sometimes named *whiteboard hacking* (Yskout et al., 2020). Such findings might indicate a need for adjusting the search criteria in similar future research to include the keywords: risk assessment, security analysis, security assessment, and whiteboard hacking. However, the application of snowballing reduced the impact of limited keywords.

9. Conclusion

This research identified various ICSs threat modeling studies, suggesting that each method has its focus and covered aspects. The decision to choose a relevant ICSs TM method is based on the practitioners' needs. There are some considerably mature methods (i.e., STRIDE and LINDDUN), while most of the proposed ICSs methods still need further validation and testing. Although many methods address safety, security, and privacy aspects separately, it is time to propose a simple, well-documented, and validated framework that covers all these aspects for ICSs.

CRediT authorship contribution statement

Shaymaa Mamdouh Khalil: Conceptualization, Data curation, Investigation, Methodology, Visualization, Writing – original draft, Writing – review & editing. **Hayretdin Bahsi:** Conceptualization, Methodology, Supervision, Writing – review & editing. **Tarmo Korõtko:** Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Declaration of generative AI and AI-assisted technologies in the writing process

During the final revision of this work, the author(s) used Bard (Google chat-based AI) in order to review the grammar and language of some sentences. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Acknowledgements

This work has been supported by the Estonian Ministry of Education and Research and the European Regional Development Fund (grant 2014-2020.4.01.20-0289).

References

Adam Shostack, 2013. Threat Modeling Designing for Security, vol. 53. John Wiley & Sons. Incorporated. https://ebookcentral.proquest.com/lib/tuee/reader.action?docID=1629177. arXiv:1011.1669v3.

Ahn, B., Kim, T., Smith, S.C., Youn, Y.W., Ryu, M.H., 2021. Security Threat Modeling for Power Transformers in Cyber-Physical Environments, pp. 1–5.

Al Asif, M.R., Hasan, K.F., Islam, M.Z., Khondoker, R., 2021. Stride-based cyber security threat modeling for IoT-enabled precision agriculture systems. In: 2021 3rd International Conference on Sustainable Technologies for Industry 4.0 (STI). IEEE, pp. 1–6.

Alberts, C., Dorofee, A., Stevens, J., Woody, C., 2003. Introduction to the OCTAVE Approach. Technical Report Carnegie-Mellon Univ. Pittsburgh PA Software Engineering Inst.

Alexander, O., Belisle, M., Steele, J., 2020. MITRE ATT&CK for industrial control systems: design and philosophy. https://collaborate.mitre.org/attackics/img_auth.php/3/37/ATT%26CK_for_ICS_-_Philosophy_Paper.pdf.

Allodi, L., Etalle, S., 2017. Towards realistic threat modeling: attack commodification, irrelevant vulnerabilities, and unrealistic assumptions. In: SafeConfig 2017 - Proceedings of the 2017 Workshop on Automated Decision Making for Active Cyber Defense, Co-Located with CCS 2017, pp. 23–26. https://doi.org/10.1145/3140368.3140372. arXiv:1801.04569.

Amro, A., Gkioulos, V., Katsikas, S., 2023. Assessing cyber risk in cyber-physical systems using the ATT&CK framework. ACM Trans. Priv. Secur. 26. https://doi.org/10.1145/3571733.

Bernsmed, K., Jaatun, M.G., 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 2019. https://doi.org/10.1109/CyberSecPODS.2019.8885144.

Bitton, R., Avraham, D., Klevansky, E., Mimran, D., Brodt, O., Lehmann, H., Elovici, Y., Shabtai, A., 2022. Adversarial machine learning threat analysis in open radio access networks. arXiv preprint. arXiv:2201.06093.

Burmester, M., Magkos, E., Chrissikopoulos, V., 2012. Modeling security in cyber–physical systems. Int. J. Crit. Infrastructures Prot. 5, 118–126. https://doi.org/10.1016/j.ijcip.2012.08.002.

Caltagirone, S., Pendergast, A., Betz, C., 2013. The diamond model of intrusion analysis. Technical Report. Center for Cyber Intelligence Analysis and Threat Research Hanover Md.

Chen, Y.T., Huang, C.C., 2019. Determining information security threats for an IoT-based energy Internet by adopting software engineering and risk management approaches. Inventions, 4. https://doi.org/10.3390/inventions4030053.

Cherepanov, A., 2017. WIN32/INDUSTROYER a New Threat for Industrial Control Systems.

Deng, M., Wuyts, K., Scandariato, R., Preneel, B., Joosen, W., 2011. A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. Requir. Eng. 16, 3–32. https://doi.org/10.1007/s00766-010-0115-7.

Fernandez, E.B., 2016. Threat modeling in cyber-physical systems. In: Proceedings - 2016 IEEE 14th International Conference on Dependable, Autonomic and Secure Computing, DASC 2016, 2016 IEEE 14th International Conference on Pervasive Intelligence and Computing, PICom 2016, 2016 IEEE 2nd International Conference on Big Data, pp. 448–453. https://doi.org/10.1109/DASC-PICom-DataCom-CyberSciTec.2016.89.

Fla, L.H., Borgaonkar, R., Tondel, I.A., Gilje Jaatun, M., 2021. Tool-assisted threat modeling for smart grid cyber security. In: 2021 International Conference on Cyber Situational Awareness, Data Analytics and Assessment. CyberSA 2021. https://doi.org/10.1109/CyberSA52016.2021.9478258.

Foldvari, A., Biczok, G., Kocsis, I., Gonczy, L., Pataricza, A., 2022. Impact Assessment of IT Security Breaches in Cyber-Physical Systems: Short paper, 1–4.

Force, J.T., 2012. Guide for Conducting Risk Assessments.

Force, J.T., 2017. Security and Privacy Controls for Information Systems and Organizations. NIST Special Publication.

Friedberg, I., McLaughlin, K., Smith, P., Laverty, D., Sezer, S., 2017. STPA-SafeSec: safety and security analysis for cyber-physical systems. J. Inf. Secur. Appl. 34, 183–196. https://doi.org/10.1016/j.jisa.2016.05.008.

Girdhar, M., Hong, J., Lee, H., Song, T.-J., 2021. Hidden Markov models based anomaly correlations for the cyber-physical security of EV charging stations. IEEE Trans. Smart Grid 13 (5), 3903–3914. https://doi.org/10.1109/tsg.2021.3122106.

Goldsmith, J., Levinson, D., Harvard, S., Review, L., May, N., 1890. The Harvard law review association. Harvard Law Rev. 4, 193–220.

Hacks, S., Katsikeas, S., Ling, E., Lagerström, R., Ekstedt, M., 2020. powerLang: a probabilistic attack simulation language for the power domain. Energy Inform. 3. https://doi.org/10.1186/s42162-020-00134-4.

Haider, M.H., Saleem, S.B., Rafaqat, J., Sabahat, N., 2019. Threat modeling of wireless attacks on advanced metering infrastructure. In: MACS 2019 - 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics, Proceedings. https://doi.org/10.1109/MACS48846.2019.9024779.

Hajrić, A., Smaka, T., Baraković, S., Husić, J.B., 2020. Methods, methodologies, and tools for threat modeling with case study. Telfor J. 12. https://doi.org/10.5937/telfor2001056H.

Hammad, E., Farraj, A., Kundu, D., 2017. Communication Links Vulnerability Model for Cyber Security Mitigation. LNICST, vol. 184.

Howard, M., LeBlanc, D., 2002. Writing Secure Code. Microsoft Press.

Hutchins, E.M., Cloppert, M.J., Amin, R.M., et al., 2011. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. Lead. Issues Inform. Warf. Secur. Res. 1, 80.

Iqbal, A., Olegard, J., Ghimire, R., 2020. Digital forensic evidence-the missing link in threat modeling. In: 2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy. ICDABI 2020. https://doi.org/10.1109/ICDABI51230.2020.9325650.

Jamil, A.M., Ben Othmane, L., Valani, A., 2022. Threat Modeling of Cyber-Physical Systems in Practice. Technical Report. https://otter.ai/. arXiv:2103.04226v1.

Jbair, M., Ahmad, B., Maple, C., Harrison, R., 2022. Threat modelling for industrial cyber physical systems in the era of smart manufacturing. Comput. Ind. 137, 103611.

Johnson, P., Lagerström, R., Ekstedt, M., 2018. A Meta Language for Threat Modeling and Attack Simulations, p. 8.

Khalil, S.M., Bahsi, H., Ochieng'Dola, H., Korõtko, T., McLaughlin, K., Kotkas, V., 2023. Threat modeling of cyber-physical systems-a case study of a microgrid system. Comput. Secur. 124, 102950.

Khan, R., Mclaughlin, K., Laverty, D., Sezer, S., 2017. STRIDE-Based Threat Modeling for Cyber-Physical Systems, p. 5.

Kim, K.H., Kim, K., Kim, H.K., 2022. Stride-based threat modeling and dread evaluation for the distributed control system in the oil refinery. ETRI J. 44, 991–1003.

Kitchenham, B., Charters, S., et al., 2007. Guidelines for Performing Systematic Literature Reviews in Software Engineering.

Kriaa, S., Pietre-Cambacedes, L., Bouissou, M., Halgand, Y., 2015. A Survey of Approaches Combining Safety and Security for Industrial Control Systems.

Kumar, R., Kela, R., Singh, S., Trujillo-Rasua, R., 2022. APT attacks on industrial control systems: a tale of three incidents. Int. J. Crit. Infrastructures Prot. 37, 100521. https://doi.org/10.1016/j.ijcip.2022.100521.

Lallie, H.S., Debattista, K., Bal, J., 2020. A review of attack graph and attack tree visual syntax in cyber security. Comput. Sci. Rev. 35, 100219.

Lee, C.C., Tan, T.G., Sharma, V., Zhou, J., 2021. Quantum computing threat modelling on a generic cps setup. In: Applied Cryptography and Network Security Workshops: ACNS 2021 Satellite Workshops, Proceedings. AIBlock, AIHWS, AIoTS, CIMSS, Cloud S&P, SCI, SecMT, and SiMLA, Kamakura, Japan, June 21–24, 2021. Springer, pp. 171–190.

Li, K., Rashid, A., Roudaut, A., 2021. Vision: security-usability threat modeling for industrial control systems. In: ACM International Conference Proceeding Series, pp. 83–88. https://doi.org/10.1145/3481357.3481527.

Ling, E., Lagerström, R., Ekstedt, M., 2020. A systematic literature review of information sources for threat modeling in the power systems domain. In: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer Science and Business Media Deutschland GmbH, pp. 47–58. https://doi.org/10.1007/978-3-030-58295-1_4.

Liu, X., Zhu, Peidong Z.Y., Chen, K., 2015. A collaborative intrusion detection mechanism against false data injection attack in advanced metering infrastructure. Smart Grid Technol. 6, 2435–2443. https://doi.org/10.1017/9781108566506.011.

Luo, F., Jiang, Y., Zhang, Z., Ren, Y., Hou, S., 2021. Threat analysis and risk assessment for connected vehicles: a survey. Secur. Commun. Netw. 2021, 1–19. https://doi.org/10.1155/2021/1263820.

Maheshwari, V., Prasanna, M., 2016. Integrating risk assessment and threat modeling within SDLC process. In: 2016 International Conference on Inventive Computation Technologies, vol. 1. ICICT, pp. 1–5.

Mai, P.X., Goknil, A., Shar, L.K., Pastore, F., Briand, L.C., Shaame, S., 2018. Modeling security and privacy requirements: a use case-driven approach. Inf. Softw. Technol. 100, 165–182. https://doi.org/10.1016/j.infsof.2018.04.007.

Marksteiner, S., Ramler, R., Sochor, H., 2019. Integrating threat modeling and automated test case generation into industrialized software security testing. In: PervasiveHealth: Pervasive Computing Technologies for Healthcare. https://doi.org/10.1145/3360664.3362698. arXiv:1911.06594.

Martins, G., Bhatia, S., Koutsoukos, X., Stouffer, K., Tang, C., Candell, R., 2015. Towards a systematic threat modeling approach for cyber-physical systems. In: Proceedings - 2015 Resilience Week. RSW 2015, pp. 114–119. https://doi.org/10.1109/RWEEK.2015.7287428.

Mead, N.R., Stehney, T., 2005. Security Quality Requirements Engineering (SQUARE) Methodology. ACM SIGSOFT Software Engineering Notes, vol. 30.

Morana, M.M., UcedaVelez, T., 2015. Risk Centric Threat Modeling: Process for Attack Simulation and Threat Analysis. John Wiley & Sons. https://learning.oreilly.com/library/view/risk-centric-threat/9780470500965/c03.xhtml#c03_level1_1.

Nweke, L.O., Wolthusen, S.D., 2020. A review of asset-centric threat modelling approaches. Int. J. Adv. Comput. Sci. Appl., 1–6. https://doi.org/10.14569/ijacsa.2020.0110201.

Papernot, N., McDaniel, P., Sinha, A., Wellman, M., 2016. Towards the science of security and privacy in machine learning. arXiv preprint. arXiv:1611.03814.

Paverd, A., Martin, A., Brown, I., 2014. Smart grid security: second international workshop. In: SmartGridSec 2014, Revised Selected Papers. Munich, Germany, February 26, 2014. https://doi.org/10.1007/978-3-319-10329-7_1.

Radoglou-Grammatikis, P., Sarigiannidis, P., Giannoulakis, I., Kafetzakis, E., Panaousis, E., 2019. Attacking IEC-60870-5-104 SCADA systems. In: Proceedings - 2019 IEEE World Congress on Services, SERVICES 2019, pp. 41–46. https://doi.org/10.1109/SERVICES.2019.00022.

Rak, M., Salzillo, G., Romeo, C., 2020. Systematic IoT penetration testing: ALEXA case study. In: CEUR Workshop Proceedings, vol. 2597, pp. 190–200.

Ramis Ferrer, B., Afolaranmi, S.O., Lastra, J.L.M., 2017. Principles and risk assessment of managing distributed ontologies hosted by embedded devices for controlling industrial systems. In: Proceedings IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society 2017-Janua, pp. 3498–3505. https://doi.org/10.1109/IECON.2017.8216592.

Raza, M.Q., Khosravi, A., 2015. A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. Renew. Sustain. Energy Rev. 50, 1352–1372.

Research, E., 2022. Industroyer2: industroyer reloaded | WeLiveSecurity. https://www.welivesecurity.com/2022/04/12/industroyer2-industroyer-reloaded/.

Rimsha, A., Rimsha, K., 2019. Development of threat modeling and risk management tool in automated process control system for gas producing enterprise. In: Proceedings - 2019 21st International Conference "Complex Systems: Control and Modeling Problems". CSCMP 2019 2019-Septe, pp. 596–599.

Rouland, Q., Hamid, B., Jaskolka, J., 2021. Specification, detection, and treatment of stride threats for software components: modeling, formal methods, and tool support. J. Syst. Archit. 117, 102073.

Saitta, P., Larcom, B., Eddington, M., 2005. Trike v.1 methodology document [draft]. http://dymaxion.org/trike/Trike_v1_Methodology_Documentdraft.pdf.

Salzillo, G., Rak, M., Moretta, F., 2020. Threat modeling based penetration testing: the open energy monitor case study. In: ACM International Conference Proceeding Series. https://doi.org/10.1145/3433174.3433181.

Scandariato, R., Wuyts, K., Joosen, W., 2015. A descriptive study of Microsoft's threat modeling technique. Requir. Eng. 20, 163–180. https://doi.org/10.1007/s00766-013-0195-2.

Schlegel, R., Obermeier, S., Schneider, J., 2015. Structured system threat modeling and mitigation analysis for industrial automation systems. In: Proceeding - 2015 IEEE International Conference on Industrial Informatics. INDIN 2015, pp. 197–203. https://doi.org/10.1109/INDIN.2015.7281734.

Sequeiros, J.B., Chimuco, F.T., Samaila, M.G., Freire, M.M., Inácio, P.R., 2020. Attack and system modeling applied to IoT, cloud, and mobile ecosystems: embedding security by design. ACM Comput. Surv. 53, 1–32.

Shevchenko, N., Chick, T.A., Riordan, P.O., Scanlon, T.P., Woody, C., 2018a. Threat Modeling: a Summary of Available Methods.

Shevchenko, N., Frye, B., Woody, C., 2018b. Threat modeling for cyber-physical system-of-systems: methods evaluation. https://resources.sei.cmu.edu/asset_files/WhitePaper/2018_019_001_526372.pdf.

Shibly, M., De Soto, B.G., 2020. Threat modeling in construction: an example of a 3d concrete printing system. In: ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction. IAARC Publications, pp. 625–632.

Shostack, A., 2008. Experiences Threat Modeling at Microsoft. CEUR Workshop Proceedings, vol. 413, pp. 1–11.

Sion, L., Yskout, K., Van Landuyt, D., Joosen, W., 2018. Risk-based design security analysis. In: Proceedings of the 1st International Workshop on Security Awareness from Design to Deployment, pp. 11–18.

Soares Cruzes, D., Gilje Jaatun, M., Bernsmed, K., Tondel, I.A., 2018. Challenges and experiences with applying Microsoft threat modeling in agile development projects. In: Proceedings - 25th Australasian Software Engineering Conference. ASWEC 2018, pp. 111–120. https://doi.org/10.1109/ASWEC.2018.00023.

Soltan, S., Mittal, P., Poor, H.V., 2018. BlackIoT: IoT botnet of high wattage devices can disrupt the power grid. In: Proceedings of the 27th USENIX Security Symposium, pp. 15–32.

Souppaya, M., Scarfone, K., 2016. Guide to data-centric system threat modeling. Technical Report. National Institute of Standards and Technology.

de Souza, N.P., César, C.d.A.C., Bezerra, J.d.M., Hirata, C.M., 2020. Extending STPA with STRIDE to identify cybersecurity loss scenarios. J. Inf. Secur. Appl. 55. https://doi.org/10.1016/j.jisa.2020.102620.

Stellios, I., Kotzanikolaou, P., Grigoriadis, C., 2021. Assessing IoT enabled cyber-physical attack paths against critical systems. Comput. Secur. 107, 102316. https://doi.org/10.1016/j.cose.2021.102316.

Stetco, A., Dinmohammadi, F., Zhao, X., Robu, V., Flynn, D., Barnes, M., Keane, J., Nenadic, G., 2019. Machine learning methods for wind turbine condition monitoring: a review. Renew. Energy 133, 620–635.

Steven, J., 2010. Threat modeling-perhaps it's time. IEEE Secur. Priv. 8, 83–86.

Stouffer, K., Pillitteri, V., Lightman, S., Abrams, M., Hahn, A., 2015a. Guide to industrial control systems (ICS) security NIST special publication 800-82 revision 2. NIST special publication 800-82 rev 2, 1–157. http://industryconsulting.org/pdfFiles/NIST_Draft-SP800-82.pdf.

Stouffer, K., Pillitteri, V., Lightman, S., Abrams, M., Hahn, A., 2015b. NIST Special Publication 800-82 Revision 2 Guide to Industrial Control Systems (ICS) Security. https://doi.org/10.6028/NIST.SP.800-82r2.

Suleiman, H., Alqassem, I., Diabat, A., Arnautovic, E., Svetinovic, D., 2015. Integrated smart grid systems security threat model. Inf. Sci. 53, 147–160. https://doi.org/10.1016/j.is.2014.12.002.

Suleiman, H., Svetinovic, D., 2013. Evaluating the effectiveness of the security quality requirements engineering (SQUARE) method: a case study using smart grid advanced metering infrastructure. Requir. Eng. 18, 251–279. https://doi.org/10.1007/s00766-012-0153-4.

Sun, T., Drouot, B., Golra, F., Champeau, J., Guerin, S., Roux, L.L., Mazo, R., Teodorov, C., Aertryck, L., Hostis, B., Sun, T., Drouot, B., Golra, F., Champeau, J., Guerin, S., Domain, A., Sun, T.N., Drouot, B., Golra, F.R., 2020. A Domain-Specific Modeling Framework for Attack Surface Modeling.

Suo, D., Siegel, J.E., Sarma, S.E., 2018. Merging safety and cybersecurity analysis in product design. IET Intell. Transp. Syst. 12, 1103–1109.

Sűren, E., Heiding, Fredrik, Olegård, J., Lagerström, R., 2023. PatrIoT: practical and agile threat research for IoT. Int. J. Inf. Secur. 22, 213–233. https://doi.org/10.1007/s10207-022-00633-3.

Tan, J., Jin, H., Hu, H., Hu, R., Zhang, H., Zhang, H., 2022. WF-MTD: evolutionary decision method for moving target defense based on Wright-Fisher process. IEEE Trans. Dependable Secure Comput., 1–14.

Tan, J., Jin, H., Zhang, H., Zhang, Y., Chang, D., Liu, X., Zhang, H., 2023. A survey: when moving target defense meets game theory. Comput. Sci. Rev. 48, 100544.

Tarandach, I., Coles, J.M., 2020. Threat Modeling: A Practical Guide for Development Teams.

Tuma, K., Calikli, G., Scandariato, R., 2018. Threat analysis of software systems: a systematic literature review. J. Syst. Softw. 144, 275–294. https://doi.org/10.1016/j.jss.2018.06.073.

Tuma, K., Scandariato, R., Widman, M., Sandberg, C., 2017. Towards security threats that matter. In: Computer Security. Springer, pp. 47–62.

UcedaVelez, T., Morana, M.M., 2015. Risk Centric Threat Modeling: Process for Attack Simulation and Threat Analysis. John Wiley & Sons.

Uzunov, A.V., Fernandez, E.B., 2014. An extensible pattern-based library and taxonomy of security threats for distributed systems. Comput. Stand. Interfaces 36, 734–747. https://doi.org/10.1016/j.csi.2013.12.008.

Valenza, F., Karafili, E., Steiner, R.V., Lupu, E.C., 2022. A hybrid threat model for smart systems. IEEE Trans. Dependable Secure Comput. 20 (5), 4403–4417.

Vernotte, A., Välja, M., Korman, M., Björkman, G., Ekstedt, M., Lagerström, R., 2018. Load balancing of renewable energy: a cyber security analysis. Energy Inform. 1, 1–42. https://doi.org/10.1186/s42162-018-0010-x.

Wang, H., Jia, Z., Shen, Z., 2009. Research on security requirements engineering process. In: IE and EM 2009 - Proceedings 2009 IEEE 16th International Conference on Industrial Engineering and Engineering Management, pp. 1285–1288.

Withers, R., 2016. Software and attack centric integrated threat modeling for quantitative risk assessment. In: Proceedings of the Symposium and Bootcamp on the Science of Security.

Wuyts, K., Scandariato, R., Joosen, W., 2014. Empirical evaluation of a privacy-focused threat modeling methodology. J. Syst. Softw. 96, 122–138. https://doi.org/10.1016/j.jss.2014.05.075.

Wuyts, K., Sion, L., Joosen, W., 2020. Linddun go: a lightweight approach to privacy threat modeling. In: 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW). IEEE, pp. 302–309.

Xiong, W., Lagerström, R., 2019. Threat modeling – a systematic literature review. Comput. Secur. 84. https://doi.org/10.1016/j.cose.2019.03.010.

Young, W., Leveson, N.G., 2014. Inside Risks an Integrated Approach to Safety and Security Based on Systems Theory, pp. 31–35.

Yskout, K., Heyman, T., Van Landuyt, D., Sion, L., Wuyts, K., Joosen, W., 2020. Threat modeling: from infancy to maturity. In: Proceedings - 2020 ACM/IEEE 42nd International Conference on Software Engineering: New Ideas and Emerging Results. ICSE-NIER 2020, pp. 9–12.

Yu, E., Mylopoulos, J., 1998. Why goal-oriented requirements engineering. In: Proceedings of the 4th International Workshop on Requirements Engineering: Foundations of Software Quality, pp. 15–22.

Zahid, S., Mazhar, M.S., Abbas, S.G., Hanif, Z., Hina, S., Shah, G.A., 2023. Threat modeling in smart firefighting systems: aligning mitre att&ck matrix and nist security controls. Int. Things 22, 100766.

Zografopoulos, I., Konstantinou, C., Georgios Tsoutsos, N., Zhu, D., Broadwater, R., 2021a. Security Assessment and Impact Analysis of Cyberattacks in Integrated T&D Power Systems 21.

Zografopoulos, I., Ospina, J., Liu, X., Konstantinou, C., 2021b. Cyber-physical energy systems security: threat modeling, risk assessment, resources, metrics, and case studies. IEEE Access 9, 29775–29818. https://doi.org/10.1109/ACCESS.2021.3058403. arXiv:2101.10198.

Shaymaa Mamdouh Khalil is a PhD candidate and an early-stage researcher at the Center for Digital Forensics and Cyber Security at Tallinn University of Technology, Estonia. She holds a BSc in electronics and communication engineering from Cairo University (2006) and an MBA International Paris from the University Paris Dauphine and the University Paris 1 Pantheon Sorbonne (2013). In 2020, she received her MSc cum laude in cyber security with a digital forensics specialization from Tallinn University of Technology and the University of Tartu. Her research interests include cyber-physical systems security and digital forensics.

Hayretdin Bahsi is a research professor at the Center for Digital Forensics and Cyber Security at Tallinn University of Technology, Estonia. He has two decades of professional and academic experience in cybersecurity. He received his PhD from Sabancı University (Turkey) in 2010. He was involved in many R&D and consultancy projects about cybersecurity as a researcher, consultant, trainer, project manager, and program coordinator at the National Cyber Security Research Institute of Turkey between 2000 and 2014. His research interests include the application of machine learning to cyber security problems, digital forensics, and cyber-physical system security.

Tarmo Korõtko received the B.Sc. and M.Sc. degrees in mechatronics and the PhD degree in energy and geotechnology from the Tallinn University of Technology (TUT), Tallinn, Estonia, in 2007, 2010, and 2019, respectively. In 2018, he became a member of the Microgrids and Metrology Research Group of the Department of Electrical Power Engineering and Mechatronics at TUT, where he is currently employed as a Senior Researcher. He has published more than 20 articles on the topics of microgrid control, electric power system digitalization, local energy markets and communities, energy storage systems, and machine learning applications in electric power systems. His research interests include microgrids, local energy markets and communities, prosumers, power system digitalization, and artificial intelligence in electric power systems. He is a member of IEEE.