A Systematic Mapping Study on Security in Configurable Safety-Critical Systems Based on Product-Line Concepts

Richard May, Jyoti Gautam, Chetan Sharma, Christian Biermann and Thomas Leich
Harz University of Applied Sciences, Wernigerode, Germany
{rmay, u36911, u36942, tleich}@hs-harz.de, christian.biermann@msg.group

Keywords: Security, Configurable Systems, Software Product Lines, Safety-Critical Systems, Systematic Mapping Study.

Abstract: Safety-critical systems are becoming increasingly configurable. However, as the number of features and configurations grows, the systems' complexity also increases, making cyber attacks more likely. Nevertheless, we miss an overview of security in configurable safety-critical systems which are based on product-line engineering. Thus, we conducted a systematic mapping study in which we analyzed 44 papers (2008–2022) to discuss relevant properties and to identify 8 research opportunities. Our key finding is that security in the context of variability and safety-critical systems needs more consideration and research. We emphasize that safety-critical systems, especially those with networking capabilities, cannot be safe if they do not provide techniques to ensure security and do not consider the systems' configurability. Our study is aimed to guide both researchers and practitioners in understanding the importance of security for configurable safety-critical systems, relevant properties, and open issues.

1 INTRODUCTION

Safety-critical systems (SCS), i.e., systems whose execution can result in fatalities or extensive damages to a system, its users, or environment, are becoming increasingly predominant in various industries, e.g., manufacturing (Knight, 2002). SCS are quite software-intensive (Nešić et al., 2019), making the software engineering process highly challenging as its main goal is to provide software that functions reliably and predictably in the event of a system breakdown, failure, regular system operation (Gutgarts and Temin, 2010). As modern SCS usually rely on networking technologies, addressing security besides safety is more important than ever before. Exploiting system vulnerabilities or even software bugs can not only lead to unauthorized data access but also to fatal system errors, and thus a compromised system safety (Mubeen et al., 2020). SCS are becoming increasingly configurable (Lohmüller and Bauer, 2019), i.e., they are composed of diverse customizable components to meet requirements, e.g., stakeholder demands (Jamshidi et al., 2017). Precisely, they are based on variability concepts comprising similar but adapted variants built on the same assets but with varying features, e.g., based on product-line engineering (PLE) (Fægri and Hallsteinsen, 2006). However, as the configurability of software systems increases, they also become more complex, e.g., due to configuration options or feature interactions (Jamshidi et al., 2017). This situation also leads to a growing attack surface, making the overall engineering process even more challenging (Wolschke et al., 2019). However, there is currently no systematic review of recent research focusing exclusively on the security of PLE-based SCS, while covering a comparable body of the literature landscape. By conducting a systematic mapping study, we address this research gap to provide an understanding of the intersection of security, PLE, and SCS. This way, our goal is to explore what and to what extent security- and variability-related properties have already been covered by existing research and what topics still need more research. With our study, we contribute the following:

- A systematic overview of recent research regarding security of configurable SCS in PLE.
- A discussion of relevant properties of the investigated topics and research opportunities.
- A replication package to ensure a higher comprehensibility and replicability of the study.[1]

Our results can help researchers and practitioners in selecting research topics and emphasize the importance of security in the development of variant-rich software in safety-critical environments.

[1] https://doi.org/10.5281/zenodo.7538781

May, R., Gautam, J., Sharma, C., Biermann, C. and Leich, T.
A Systematic Mapping Study on Security in Configurable Safety-Critical Systems Based on Product-Line Concepts.
In *Proceedings of the 18th International Conference on Software Technologies (ICSOFT 2023)*, pages 217-224
ISBN: 978-989-758-665-1; ISSN: 2184-2833

2 BACKGROUND

In this section, we provide relevant background information on configurable systems, SCS, and security.

2.1 Configurable Systems

Today's systems typically offer the ability to be customized (i.e., including or excluding functionalities) according to stakeholder demands, hardware limitations, or legal regulations (Iqbal et al., 2022). In this context, system variants comprising individual features can be developed to meet relevant requirements. To ensure that all features have been successfully configured and that they are actually functional, it is necessary to verify them — usually based on 3 different strategies, i.e., *feature-, product-, or family-based* (Thüm et al., 2014). Configurable systems are usually based on techniques and tools allowing the organization, modeling, documentation, and implementation of their configuration options (i.e., features). A well-known method to manage these variability mechanics is PLE (Schaefer et al., 2012). Configurable systems are typically classified by their projection. The *problem space* refers to the abstraction of the domain, while the *solution space* addresses a system's implementation. If both spaces are connected it is called *mapping* (Apel et al., 2013a).

2.2 Safety-Critical Systems

The term *safety-critical* refers to the loss of human lives, economic losses, or environmental damage, e.g., in avionics (Knight, 2002). So, the main goal in the development of SCS is to ensure safety, i.e., implementing functions that protect the overall system, parts of the system, or their users against undesirable events that could cause harm (Rausand, 2014). One of the most common safety standards is IEC 61508 (2010). According to this standard, there are 2 failure categories, precisely hardware failures and systematic failures. Both categories can be measured by the 4-level software integrity level (SIL) which is the basic measure that needs to be considered when developing safe SCS (IEC 61508, 2010). Nevertheless, due to increasing software complexity, networking, and growing data amount, ensuring the minimum possible safety risk for SCS is becoming quite challenging. Precisely, safety is sensitive to system changes, e.g., system configurations (Debbech et al., 2019). Furthermore, ensuring safety also implies ensuring security of a system (Hatcliff et al., 2014), meaning the presence of security issues has a major impact on the safety of SCS (Ebnauf et al., 2019).

2.3 Security

Security is a software quality property, aiming to protect data against unauthorized access of persons or systems which are not allowed to (Fægri and Hallsteinsen, 2006). It is usually characterized by assets (e.g, sensitive data), threats, risks, and measures (Myllärniemi et al., 2015). A threat is defined as an unwanted but potential event that could harm a system. A risk arises when a threat can potentially be exploited, e.g., attacks, bugs, or environmental errors. Overall, there are 6 security goals for software systems, including *confidentiality*, *integrity*, *availability* (i.e., the CIA triad), *accountability*, *authenticity*, and *non-repudiation* (i.e., information security goals) (ISO/IEC 27000, 2018).

3 METHODS

To accomplish our main goal, we conducted a systematic mapping study according to Petersen et al. (2015) (cf. Figure 1).

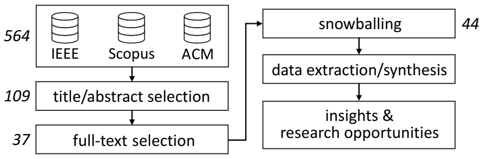

Figure 1: Methodological study overview (numbers indicate the amount of selected papers).

3.1 Study Design

Search String. First, we created the following search string for a search in IEEE XPLORE, SCOPUS, and the ACM GUIDE TO COMPUTING LITERATURE, consisting of relevant keywords of PLE and SCS:

> *("product line*" OR "SPL" OR "SPLE") AND ("product famil*" OR "system famil*" OR "software famil*" OR "config*" OR "variab*" OR "variant*" OR "feature" OR "model*") AND ("safety-critical")*

The term *security* or related terms are intentionally not included to be able to also find papers only implicitly dealing with security.

We are aware of a mapping study by Kenner et al. (2021) focusing on safety and security in the context of configurable software systems. However, they analyzed safety and security separately (i.e., articles retrieved from Scopus that are either about safety or security) without focusing exclusively on SCS. Thus, we cover a larger body of knowledge with another research focus.

Selection Criteria. Papers were only selected if they were published between 2008 and 2022 at a peer-reviewed conference or journal provide with at least 4 pages and are concerned with security of configurable SCS. We intentionally excluded work focusing on configurable SCS but not considering PLE, as well as work describing safety-oriented approaches but does not explicitly fall within the scope of SCS.

Extraction Criteria. To extract data, we defined criteria oriented towards PLE and security. We do not focus on safety but only on security with regard to our research objectives.

1. **Publication**
   - *Publication year* of the paper.
   - *Contribution type* as classification of a paper, i.e., open items, method, model, metric, or tool (Engström and Runeson, 2011).
   - *Domain* of the SCS, e.g., automotive, avionics.
   - *Perspective* indicates whether the paper's focus is on security of a system or on a security concern based on PLE (Kenner et al., 2021).
2. **Configurable SCS**
   - *Variability type* specifies whether the software, or both software and hardware are configurable (Queiroz and Braga, 2014).
   - *Projection* reflects if the SCS addresses the problem space, solution space, or a mapping between both (Apel et al., 2013a).
   - *Verification*, classifies the SCS regarding verification strategies (Thüm et al., 2014).
   - *Evolution* indicates whether system evolution aspects are considered (Apel et al., 2013a).
3. **Security**
   - *Security standard* addressed in the paper.
   - *Security goals* of the software related to the SCS, including goals of the *CIA triad*, *information security*, and further goals (e.g., reliability).
   - *Security threats and risks* which are addressed in the publication, including the description of patterns.
   - *Security measures* discussed or suggested to tackle security risks.

3.2 Study Conduct

The automated search was conducted on November 1st, 2022 and yielded in 564 publications (534 after duplication removal). Then, papers were selected by the first author according to the selection criteria (109 papers). However, we first considered all papers regarding PLE-related SCS even when they did not mention security in the title or abstract. After reading full-texts, we selected 37 papers which mention or describe security to a certain degree. To increase the number of publications, we did 1 iteration of forward and backward snowballing. So, we found 7 more papers resulting in 44 papers in total (cf. Figure 1). Next, the data extraction was carried out by the first, second, and third author. We used open-coding to label the identified data and open-card-sorting to fit recurring data into common data categories. Finally, all data was discussed by all authors to derive both well-covered topics and future research opportunities.

4 RESULTS

In this section, we present the results of the literature analysis (cf. Table 1).

4.1 Publication

Publication Years. We did not find any work that addressed security concerns until 2010. In the following 4 years (2011–2014), the number of papers is consistently low (average of 2 papers per year). From 2015–2018, the number of papers increased to 4 papers per year. After a peak of 8 papers (2019), the number of papers decreased to an average of 4 papers per year (2020–2022). Surprisingly, we identified only 2 papers in 2022.

Contribution Type. In most papers (17) a method was introduced. 12 papers described open items. In 11 cases, models as problem solutions are described. However, we did not identify an approach focusing on metrics, especially in the context of security. In 4 cases, authors introduced tools.

Domain. The publications covered various domains. The most common domains are cyber-physical systems (8), avionics (5), automotive (5), medical (4), and production (3). 13 papers deal with general configurable SCS.

Perspectives. Most publications (32) are concerned with security for configurable systems, e.g., security as quality property for secure communication (Heikkilä et al., 2016). In 6 papers, the authors realized security functions based on variability techniques, e.g., secure PLE in cloud environments (Krieter et al., 2018). Moreover, we extracted 6 approaches focusing on both perspectives.

4.2 Configurable Safety-Critical Systems

Variability Type. We identified that half of the papers (22) address only the configurability of SCS software. The other half is focused on the configurability of both software and hardware components.

Table 1: Overview of the extracted data related to security and PLE.

				Security goals																
				CIA triad			Information security							Projection			Verification			
Reference	PLE for security	Security for PLE	Security standard	Confidentiality	Integrity	Availability	Authorization	Accountability	Non-repudiation	Further goals	Threats and risks	Patterns	Measures	Problem space	Mapping	Solution space	Feature-based	Product-based	Family-based	Evolution
Trujillo et al. (2010)	○	●	○	○	●	○	○	○	○	●	●	○	●	●	○	○	○	○	○	○
Dordowsky et al. (2011)	○	●	○	○	○	○	○	○	○	○	●	○	●	○	○	●	●	○	○	○
Cichos et al. (2011)	○	●	○	○	○	○	○	○	○	○	●	○	●	○	○	●	●	○	○	○
Li and Yang (2012)	●	●	○	●	○	●	○	○	○	●	●	○	●	○	●	○	○	○	●	●
Apel et al. (2013b)	○	●	○	○	○	●	○	○	○	●	●	○	●	○	○	●	○	○	●	○
Ubayashi et al. (2013)	●	●	○	○	○	○	○	○	○	●	●	○	○	○	●	○	●	○	○	○
Andel et al. (2014)	●	○	○	●	●	●	○	○	○	○	●	○	●	○	●	○	○	●	○	○
Cleland-Huang et al. (2014)	○	●	○	○	●	●	●	○	○	●	●	○	○	●	○	○	●	○	○	●
Hatcliff et al. (2014)	○	●	○	○	●	○	○	●	○	●	●	○	○	●	○	○	●	○	○	●
Gallina and Fabre (2015)	●	○	○	●	●	●	○	○	○	●	●	○	●	○	●	○	○	○	○	●
Ayala et al. (2015)	○	●	○	○	○	○	○	○	○	○	●	○	○	○	○	●	○	○	○	●
Vogel-Heuser et al. (2015)	○	●	○	●	○	○	○	○	○	○	○	○	○	●	○	○	●	○	○	●
Arrieta et al. (2015)	○	●	○	○	○	○	●	○	○	○	○	○	○	●	○	○	○	○	○	○
Barron et al. (2016)	○	●	○	○	●	○	○	○	○	○	●	●	●	○	●	○	○	○	○	○
Etigowni et al. (2016)	●	○	○	●	●	○	●	○	○	●	●	●	●	○	●	○	○	○	○	○
Kuhrmann et al. (2016)	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	●	○	○	○	●
Heikkilä et al. (2016)	○	●	○	○	○	○	○	○	○	○	●	○	●	○	○	●	○	○	○	○
Barner et al. (2017)	○	●	○	○	○	○	○	○	○	●	●	○	○	○	○	●	●	○	○	○
Carpenter et al. (2017)	●	●	●	●	●	●	●	○	○	●	●	●	●	○	○	●	●	○	○	○
Nicolas et al. (2017)	○	●	○	○	●	○	○	○	○	○	○	○	○	○	○	●	○	●	○	○
Pessoa et al. (2017)	○	●	○	○	○	●	○	○	○	●	○	○	○	○	●	○	●	○	○	●
Gannouni et al. (2018)	○	●	○	○	○	○	○	○	○	○	○	○	○	○	●	○	●	○	○	○
Islam and Azim (2018)	○	●	○	●	●	○	●	○	○	●	○	○	○	○	○	●	●	○	○	●
Krieter et al. (2018)	●	○	○	●	●	○	○	○	○	○	●	○	●	●	○	○	○	●	○	●
Nešić and Nyberg (2018)	○	●	○	○	○	○	○	○	○	○	●	○	○	●	○	○	○	○	○	○
Bennaceur et al. (2019)	●	○	●	○	○	●	○	○	○	●	●	●	●	○	●	○	○	●	○	●
de Oliveira et al. (2019)	○	●	○	○	●	●	○	○	○	○	●	○	○	○	●	○	○	●	○	○
Lohmüller and Bauer (2019)	○	●	○	○	○	○	○	○	○	○	○	○	●	●	○	○	○	●	○	○
Ebnauf et al. (2019)	○	●	○	○	○	○	○	○	○	○	○	○	○	○	○	●	○	○	○	○
Meixner et al. (2019)	○	●	○	○	○	●	○	○	○	○	○	○	○	●	○	○	○	○	○	●
Wolschke et al. (2019)	○	●	○	○	○	○	○	○	○	○	○	○	●	●	○	○	●	○	○	●
Shaaban et al. (2019)	●	○	●	●	●	○	●	○	●	●	●	○	●	○	○	●	○	●	○	○
Chumpitaz et al. (2019)	○	●	○	○	○	●	●	○	○	●	●	○	●	●	○	○	○	○	○	●
Burow et al. (2020)	●	●	○	●	●	○	○	○	●	○	●	○	●	○	●	○	○	○	○	○
Ghamizi et al. (2020)	●	●	○	●	○	○	○	○	○	○	●	●	●	○	○	●	○	○	○	○
Bressan et al. (2020)	○	●	○	○	●	●	○	○	○	○	○	○	○	○	●	○	○	○	○	○
Freitas et al. (2020)	○	●	○	○	●	○	○	○	○	○	●	○	○	●	○	○	●	○	○	○
Fischer et al. (2021)	○	●	○	○	○	●	●	○	○	○	●	○	●	●	○	○	○	○	○	○
Castro et al. (2021)	○	●	○	○	○	○	○	○	○	○	○	○	●	○	○	●	○	○	●	●
Nešić et al. (2021)	○	●	○	●	○	○	○	○	○	○	○	○	●	○	●	○	●	○	○	○
White et al. (2021)	○	●	○	○	○	○	○	○	○	○	○	○	○	○	●	○	○	○	○	○
Bressan et al. (2021)	○	●	○	○	○	○	○	○	○	○	○	○	●	○	●	○	○	○	○	○
Zampetti et al. (2022)	●	●	○	○	○	●	○	○	○	○	●	○	●	●	○	○	○	●	○	●
Prikler and Wotawa (2022)	○	●	○	○	○	○	○	○	○	○	●	○	●	●	○	○	●	○	○	○

● Fulfilled ○ Not fulfilled

Projection. We found 15 papers in the problem space and 14 paper in the solution space. 15 publications covered a mapping.

Verification. We discovered that 26 publications described verification. Usually, the authors presented a feature-based verification (15). In 8 cases, we found information regarding product-based verification. Authors described family-based verification strategies in 3 cases.

Evolution. Software evolution is addressed in 16 papers. However, we note that evolution is usually only mentioned and not described in detail. In 28 papers, evolution was not mentioned at all.

4.3 Security

Standards. We found only 3 papers covering security standards. These include AAMI TIR57 as a security management guidance document in the medical domain to meet the requirements of ISO 14971, the general guidelines of the NIST cyber-physical systems program, and the production-related security standards IEC 62443 and IEEE 1686.

Goals. The goals of the CIA triad were only described by 3 papers. Integrity (16) is the most common security goal. However, note that integrity may also refer to the SIL. Confidentiality was described by a total of 12 papers. Only 1 paper referred to account-

ability (Hatcliff et al., 2014) and 2 papers to non-repudiation (Shaaban et al., 2019; Burow et al., 2020). Authorization was mentioned or described in 8 cases.
Threats and Risks. The publications showed diverse security threats (30) or risks (11). Most common threats are system complexity (9), trust (7), general vulnerabilities (6), bugs (3), and configurability (3). Mentioned risks include unauthorized access (4), cyber attacks (2), or manipulation (2).
Measures. The identified security measures can be organized in 3 groups: threat and risk prevention (11), system architecture (11), and mitigation techniques (10). Threat and risk prevention involves risk assessment strategies (7), security policies (2), certification (1), and security testing (1). System architecture refers to isolation techniques (4), decentralization (2), program partitioning and diversity (2), dynamic variants (1), system hiding (1), and changing library locations (1). The mitigation techniques cover a wide range and are usually described in a more general way, e.g., encryption (2).

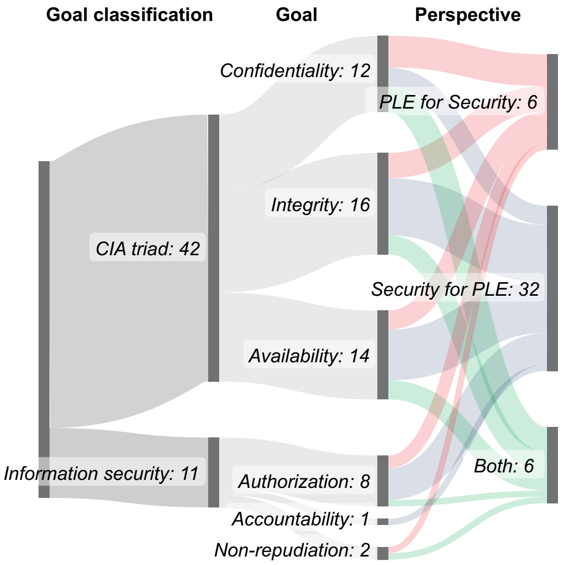

Figure 2: Distribution of security goals and perspectives (numbers indicate the amount of papers which can exceed 100% due to possibility of multiple goals per paper).

5 DISCUSSION

Next, we discuss the study results to provide 8 relevant **research opportunities (RO)**.

5.1 Configurable Safety-Critical Systems

Referring to the publication years, there is a trend towards the number of papers related to configurable SCS since 2015. Interestingly, until 2017, most papers focused on proposing methods, while since 2018, there is a trend regarding open items. This may be due to the evolving IoT technologies, which offer a variety of opportunities but also challenges related to system architectures as well as security and safety risks. Consequently, there is an increasing number of papers which are more related to the problem space or the mapping. So, we argue that **($\mathbf{RO_1}$)** *more research is needed focusing on the actual solution of problems or configurations (i.e., mappings) of configurable SCS.*

The selected configurable SCS were not verified in most cases. However, while general configurable software and storages refer more to product-based verification (Kenner et al., 2021), configurable SCS focus much more on the verification of features. This fact may refer to the safety-critical properties of SCS, meaning their great dependence on safety-related functionalities, e.g., the manipulation of assets such as sensors. Nevertheless, the proportion of papers referring only to safety-related verification is most common. In this context, we emphasize that **($\mathbf{RO_2}$)** *SCS should be verified regarding both safety and security functionalities.*

64% of the papers do not refer to evolution, e.g., feature updates. It is often not clear how such changes are handled, how they influence each other, or how they influence the security and thus safety of SCS. So, we argue that it is essential to **($\mathbf{RO_3}$)** *explore the dependencies of evolution processes, e.g., vulnerabilities caused by evolutionary changes.*

5.2 Security

Although security issues have an impact on the safety of SCS (Ebnauf et al., 2019), no reference was found to the ISO/IEC 27000 series as a major security standard. However, security requirements as well as safety requirements (e.g., SIL) should be taken into account according to standards. So, we recommend to **($\mathbf{RO_4}$)** *consider both safety and security standards.*

Not surprisingly, information on security, i.e., goals, risks, or measures, is usually missing or only given in a superficial way. However, threats are often named, albeit from a high-level perspective. For instance, while complexity or trust are mentioned as properties that may threaten the security of SCS, actions to exploit these threats are not described. So, the existence of threats and their relevance is known but **($\mathbf{RO_5}$)** *ideas of actual prevention strategies and the relationships between the IoT as a major driver, configurability, and safety seem quite under-explored.* Referring to goals, these are mainly oriented towards requirements which are most relevant for applications with IoT capabilities, i.e., the CIA triad and authorization (cf. Figure 2).

5.3 Security in Configurable Safety-Critical Systems

Research on SCS focuses rather on safety (52%) than on security (21%) or both equally (27%). Although this finding is not surprising, it highlights that the relevance of security and its configurability is underestimated. Since the number of attacks is constantly increasing and the possible consequences of a successful attack are even more fatal for SCS, the number of papers addressing safety and security equally should be significantly higher. Thus, **(RO_6)** *both the configurability of security and the equal consideration of safety and security for SCS needs more research.*

Although 30% of the papers are designed for general domains, 25% are developed for industrial applications, e.g., cyber-physical systems. Consequently, there is a huge potential for SCS-related research in this area, emphasizing the dependence of safety-related industrial applications on security. The failure of SCS can not only endanger people and the environment in this context but the economic damage, i.e., time, quality, and costs, may also be considerable. However, only 2 of these papers refer to (domain-related) security standards (e.g., IEEE 1686). Thus, it is usually not clear, how these standards are applied and which interactions or dependencies regarding potential configurations may occur. So, it is essential to **(RO_7)** *provide an understanding of security standards and implementation possibilities.*

Moreover, new requirements based on potential dependencies or configuration options may occur when considering both safety and security requirements in the SCS development process, e.g., cyber-physical systems and the application of IoT technologies. Although the interplay of safety and security is already discussed in research (Lyu et al., 2019), the consideration of variability aspects influencing the implementation of these requirements is missing. Thus, **(RO_8)** *there is a need for analyzing emerging dependencies between safety and security in the context of configurability.*

5.4 Threats to Validity

We lack comprehensive information about the accuracy of results (e.g., evaluations) as well as term understandings (e.g., integrity regarding safety or security). Some authors explain their work in great detail while others describe their approach from a high-level perspective. Another issue arises regarding the fulfillment of the extraction criteria. Specifically, the papers usually provide only a few details on security. Both issues regarding the internal validity caused several assumptions on our side and might have led to misinterpretations. We are aware that our automated search and manual filtering strategy may have resulted in the unintended exclusion of publications. This issue may pose threats to the external validity. We aimed to mitigate these threats by systematically extracting data by several researchers relying on established definitions and criteria. All results were discussed until consensus was achieved, especially in the context of data interpretation and assignments. Overall, we argue that our results are valuable and provide highly interesting insights as we covered 15 years of research relying on 3 well-established literature databases.

6 RELATED WORK

We found several papers dealing with the analysis of security in the context of configurable systems. For instance, Hammani (2014) analyzed non-functional requirements in the context of PLE. The review of Queiroz and Braga (2014) is concerned with approaches dealing with PLE and SCS (2006–2013), however, without considering security. Geraldi et al. (2020) analyzed papers which are concerned with the IoT as well as PLE (2006–2018). SCS are mentioned as domain while security is considered as a non-functional requirement.

Overall, most related work covers rather software engineering in general, is focused either on security, safety, or SCS, is not related to PLE, or is not as comprehensive as our study (i.e., number of analyzed papers or literature databases). In contrast to these studies, we provide a comprehensive review with a different focus than any related work, i.e., security of PLE-related configurable SCS. The closest work related to ours is a mapping study by Kenner et al. (2021) which is focused on security and safety of configurable software systems (2011–2020). However, they did not exclusively focus on security of SCS.

7 CONCLUSION

In this paper, we conducted a systematic mapping study on security concerns in configurable SCS (2008–2022). We identified relevant insights and presented 8 research opportunities in the context of secure, configurable SCS. Overall, the intersection of security, configurability, and SCS needs more research to create an understanding of how security is related to configurability and which dependencies exist between security and safety in this context. We emphasize that SCS cannot be safe if they do not provide techniques to ensure security which also takes into account variable features. So, when modeling or configuring SCS, security features and their strategies must be as dynamic as the system features are. Further research is strongly recommended, e.g., analyzing relevant security requirements in accordance to current security and safety standards.

REFERENCES

Andel, T. R., Whitehurst, L. N., and McDonald, J. T. (2014). Software security and randomization through program partitioning and circuit variation. In *MTD*. ACM.

Apel, S., Batory, D., Kästner, C., and Saake, G. (2013a). *Feature-oriented software product lines*. Springer.

Apel, S., Von Rhein, A., Wendler, P., Größlinger, A., and Beyer, D. (2013b). Strategies for product-line verification: Case studies and experiments. In *ICSE*. IEEE.

Arrieta, A., Sagardui, G., and Etxeberria, L. (2015). Cyber-physical systems product lines: Variability analysis and challenges. *Jornadas de Computación Empotrada*.

Ayala, I., Amor, M., Fuentes, L., and Troya, J. M. (2015). A software product line process to develop agents for the iot. *Sensors*, 15(7).

Barner, S., Diewald, A., Migge, J., Syed, A., Fohler, G., Faugere, M., and Pérez, D. G. (2017). Dreams toolchain: Model-driven engineering of mixed-criticality systems. In *MODELS*. IEEE.

Barron, S., Cho, Y. M., Hua, A., Norcross, W., Voigt, J., and Haimes, Y. (2016). Systems-based cyber security in the supply chain. In *SIEDS*. IEEE.

Bennaceur, A., Ghezzi, C., Tei, K., Kehrer, T., Weyns, D., Calinescu, R., Dustdar, S., Hu, Z., Honiden, S., and Ishikawa, F. (2019). Modelling and analysing resilient cyber-physical systems. In *SEAMS*. IEEE.

Bressan, L., de Oliveira, A. L., and Campos, F. (2020). An approach to support variant management on safety analysis using chess error models. In *EDCC*. IEEE.

Bressan, L., de Oliveira, A. L., Campos, F., and Capilla, R. (2021). A variability modeling and transformation approach for safety-critical systems. In *VaMoS*. ACM.

Burow, N., Burrow, R., Khazan, R., Shrobe, H., and Ward, B. C. (2020). Moving target defense considerations in real-time safety-and mission-critical systems. In *MTD*. ACM.

Carpenter, T., Hatcliff, J., and Vasserman, E. Y. (2017). A reference separation architecture for mixed-criticality medical and iot devices. In *SafeThings*. ACM.

Castro, T., Teixeira, L., Alves, V., Apel, S., Cordy, M., and Gheyi, R. (2021). A formal framework of software product line analyses. *TOSEM*, 30(3).

Chumpitaz, L., Furda, A., and Loke, S. (2019). Evolving variability requirements of iot systems. In *Software Engineering for Variability Intensive Systems*. Auerbach Publications.

Cichos, H., Oster, S., Lochau, M., and Schürr, A. (2011). Model-based coverage-driven test suite generation for software product lines. In *MODELS*. Springer.

Cleland-Huang, J., Gotel, O. C. Z., Huffman Hayes, J., Mäder, P., and Zisman, A. (2014). Software traceability: Trends and future directions. In *ICSE*.

de Oliveira, A. L., Braga, R., Masiero, P., Parker, D., Papadopoulos, Y., Habli, I., and Kelly, T. (2019). Variability management in safety-critical systems design and dependability analysis. *Journal of Software: Evolution and Process*, 31(8).

Debbech, S., Bon, P., and Dutilleul, S. C. (2019). Conceptual modelling of the dynamic goal-oriented safety management for safety critical systems. In *ICSOFT*. ACM.

Dordowsky, F., Bridges, R., and Tschope, H. (2011). Implementing a software product line for a complex avionics system. In *SPLC*. IEEE.

Ebnauf, M., Abdelmoez, W., Ammar, H. H., Hassan, A., and Abdelhamid, M. (2019). State-driven architecture design for safety-critical software product lines. In *ICOM*. IEEE.

Engström, E. and Runeson, P. (2011). Software product line testing–a systematic mapping study. *Information and Software Technology*, 53(1).

Etigowni, S., Tian, D., Hernandez, G., Zonouz, S., and Butler, K. (2016). Cpac: Securing critical infrastructure with cyber-physical access control. In *ACSAC*. ACM.

Fægri, T. E. and Hallsteinsen, S. (2006). A software product line reference architecture for security. In *Software Product Lines*. Springer.

Fischer, S., Ramler, R., Klammer, C., and Rabiser, R. (2021). Testing of highly configurable cyber-physical systems–a multiple case study. In *VaMoS*. ACM.

Freitas, L., Scott III, W. E., and Degenaar, P. (2020). Medicine-by-wire: Practical considerations on formal techniques for dependable medical systems. *Science of Computer Programming*, 200.

Gallina, B. and Fabre, L. (2015). Benefits of security-informed safety-oriented process line engineering. In *DASC*. IEEE.

Gannouni, W., Doumbia, M. L., and Badri, A. (2018). Systematic approach furthering confirmation measures of safety critical automotive systems. *Transactions on The Built Environment*, 174.

Geraldi, R. T., Reinehr, S., and Malucelli, A. (2020). Software product line applied to the internet of things: A systematic literature review. *Information and Software Technology*, 124.

Ghamizi, S., Cordy, M., Papadakis, M., and Traon, Y. L. (2020). Featurenet: Diversity-driven generation of deep learning models. In *ICSE*. ACM.

Gutgarts, P. B. and Temin, A. (2010). Security-critical versus safety-critical software. In *HST*. IEEE.

Hammani, F. Z. (2014). Survey of non-functional requirements modeling and verification of software product lines. In *RCIS*. IEEE.

Hatcliff, J., Wassyng, A., Kelly, T., Comar, C., and Jones, P. (2014). Certifiably safe software-dependent systems: Challenges and directions. *FOSE*.

Heikkilä, T., Dobrowiecki, T., and Dalgaard, L. (2016). Dealing with configurability in robot systems. In *MESA*. IEEE.

IEC 61508 (2010). Functional safety. Standard, IEC.

Iqbal, M. S., Krishna, R., Javidian, M. A., Ray, B., and Jamshidi, P. (2022). Unicorn: Reasoning about configurable system performance through the lens of causality. In *EuroSys*. ACM.

Islam, N. and Azim, A. (2018). Assuring the runtime behavior of self-adaptive cyber-physical systems using feature modeling. In *CSSE*. ACM.

ISO/IEC 27000 (2018). Information technology – security techniques – information security management systems. Standard, ISO.

Jamshidi, P., Velez, M., Kästner, C., Siegmund, N., and Kawthekar, P. (2017). Transfer learning for improving model predictions in highly configurable software. In *SEAMS*. ACM.

Kenner, A., May, R., Krüger, J., Saake, G., and Leich, T. (2021). Safety, security, and configurable software systems: a systematic mapping study. In *SPLC*.

Knight, J. C. (2002). Safety critical systems: Challenges and directions. In *ICSE*.

Krieter, S., Krüger, J., Weichbrodt, N., Sartakov, V. A., Kapitza, R., and Leich, T. (2018). Towards secure dynamic product lines in the cloud. In *ICSE*. IEEE.

Kuhrmann, M., Ternité, T., Friedrich, J., Rausch, A., and Broy, M. (2016). Flexible software process lines in practice: A metamodel-based approach to effectively construct and manage families of software process models. *JSS*, 121.

Li, D. and Yang, Y. (2012). Enhance value by building trustworthy software-reliant system of systems from software product lines. In *PLEASE*. IEEE.

Lohmüller, P. and Bauer, B. (2019). Software product line engineering for safety-critical systems. In *MODELSWARD*. ACM.

Lyu, X., Ding, Y., and Yang, S.-H. (2019). Safety and security risk assessment in cyber-physical systems. *Cyber-Physical Systems: Theory & Applications*, 4(3).

Meixner, K., Rabiser, R., and Biffl, S. (2019). Towards modeling variability of products, processes and resources in cyber-physical production systems engineering. In *SPLC*. ACM.

Mubeen, S., Lisova, E., and Vulgarakis Feljan, A. (2020). Timing predictability and security in safety-critical industrial cyber-physical systems: A position paper. *Applied Sciences*, 10(9).

Myllärniemi, V., Raatikainen, M., and Männistö, T. (2015). Representing and configuring security variability in software product lines. In *QoSA*. ACM.

Nešić, D. and Nyberg, M. (2018). Verifying contract-based specifications of product lines using description logic. In *DL*, volume 2211. CEUR-WS.

Nešić, D., Nyberg, M., and Gallina, B. (2019). Constructing product-line safety cases from contract-based specifications. In *SAC*. ACM.

Nešić, D., Nyberg, M., and Gallina, B. (2021). Product-line assurance cases from contract-based design. *JSS*, 176.

Nicolas, C.-F., Eizaguirre, F., Larrucea, A., Barner, S., Chauvel, F., Sagardui, G., and Perez, J. (2017). Gsn support of mixed-criticality systems certification. In *SafeComp*. Springer.

Pessoa, L., Fernandes, P., Castro, T., Alves, V., Rodrigues, G. N., and Carvalho, H. (2017). Building reliable and maintainable dynamic software product lines: An investigation in the body sensor network domain. *Information and Software Technology*, 86.

Petersen, K., Vakkalanka, S., and Kuzniarz, L. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64.

Prikler, L. M. and Wotawa, F. (2022). Challenges of testing self-adaptive systems. In *SPLC*. ACM.

Queiroz, P. G. G. and Braga, R. T. V. (2014). Development of critical embedded systems using model-driven and product lines techniques: A systematic review. In *SBCARS*. IEEE.

Rausand, M. (2014). *Reliability of Safety-Critical Systems: Theory and Applications*. Wiley.

Schaefer, I., Rabiser, R., Clarke, D., Bettini, L., Benavides, D., Botterweck, G., Pathak, A., Trujillo, S., and Villela, K. (2012). Software diversity: State of the art and perspectives. *STTT*, 14(5).

Shaaban, A. M., Gruber, T., and Schmittner, C. (2019). Ontology-based security tool for critical cyber-physical systems. In *SPLC*. ACM.

Thüm, T., Apel, S., Kästner, C., Schaefer, I., and Saake, G. (2014). A classification and survey of analysis strategies for software product lines. *CSUR*, 47(1).

Trujillo, S., Perez, A., Gonzalez, D., and Hamid, B. (2010). Towards the integration of advanced engineering paradigms into rces: Raising the issues for the safety-critical model-driven product-line case. In *SD4RCES*. ACM.

Ubayashi, N., Nakajima, S., and Hirayama, M. (2013). Context-dependent product line engineering with lightweight formal approaches. *Science of Computer Programming*, 78(12).

Vogel-Heuser, B., Fay, A., Schaefer, I., and Tichy, M. (2015). Evolution of software in automated production systems: Challenges and research directions. *JSS*, 110.

White, D., Sahlab, N., Jazdi, N., and Weyrich, M. (2021). Environment modeling for evaluating system variants in model-based systems engineering. *Procedia CIRP*, 104.

Wolschke, C., Becker, M., Schneickert, S., Adler, R., and MacGregor, J. (2019). Industrial perspective on reuse of safety artifacts in software product lines. In *SPLC*. ACM.

Zampetti, F., Tamburri, D. A., Panichella, S., Panichella, A., Canfora, G., and Penta, M. D. (2022). Continuous integration and delivery practices for cyber-physical systems: An interview-based study. *Transactions on Software Engineering and Methodology*.